Yao Yao
Professor Patrick Brown
STA442
5, November, 2021

Question 1

Impact of the fall of the Berlin Wall and the global lockdown during the COVID-19 Pandemic on CO2 Concentration

Introduction

CO2 has an inseparable relationship with our lives. To some extent, CO2 concentration would reflect the development of human's industry and economy. Also, human activities would have a huge impact on CO2 concentrations. At the same time, nowadays CO2 concentration has become a very serious problem. Hence, monitoring and controlling CO2 concentration to avoid excessive damage to the environment has become one of the most important issues in the world. Thus, it is necessary to analyze the relationship between human activities and CO2 concentration. So I analyzed daily atmospheric Carbon Dioxide concentrations from an observatory in Hawaii, made available by the Scripps CO2 Program atscrippsco2.ucsd.edu. The following is the link to the data. http://scrippsco2.ucsd.edu/assets/data/atmospheric/stations/flask_co2/daily/daily_flask_co2_mlo.csv. This data shows the daily CO2 concentration of all Hawaii from 1960-03-30 to 2021-09-28. Based on this data, I explored the impact of two major events, the fall of the Berlin Wall and the global lockdown during covid-19 on CO2 concentration.

Method

First, I plot the CO2 concentration at Mauna Loa Observatory, Hawaii over all years and in recent years. Through Figure 1, we could observe that the overall trend of CO2 concentration is increasing and it has a seasonal pattern. This characteristic would be more obvious in Figure2. In Figure2, we could find that there is a repeating trend of rising and then falling. Thus, we add cosine and sine terms in our model as seasonal trending. Besides, we could also observe that the random effect of each time point in Figure 5 is different, which demonstrates that the random slope along all the time has a different value.

Figure1: CO2 at Mauna Loa Observatory, Hawaii over all years

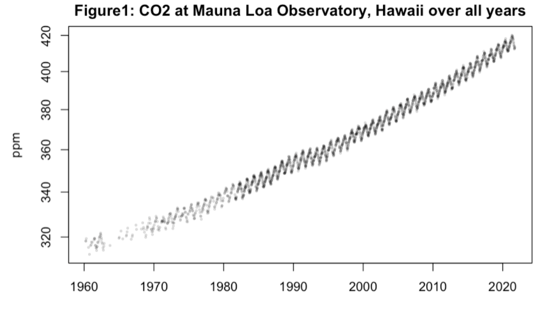


Figure2: CO2 at Mauna Loa Observatory, Hawaii in recent years

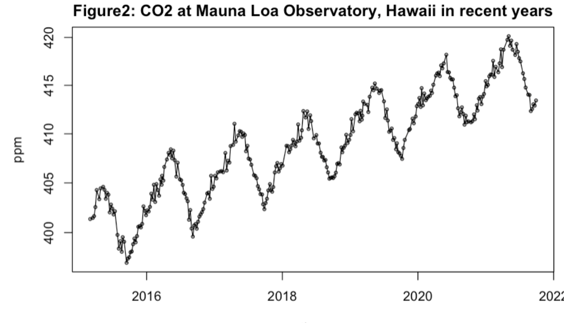
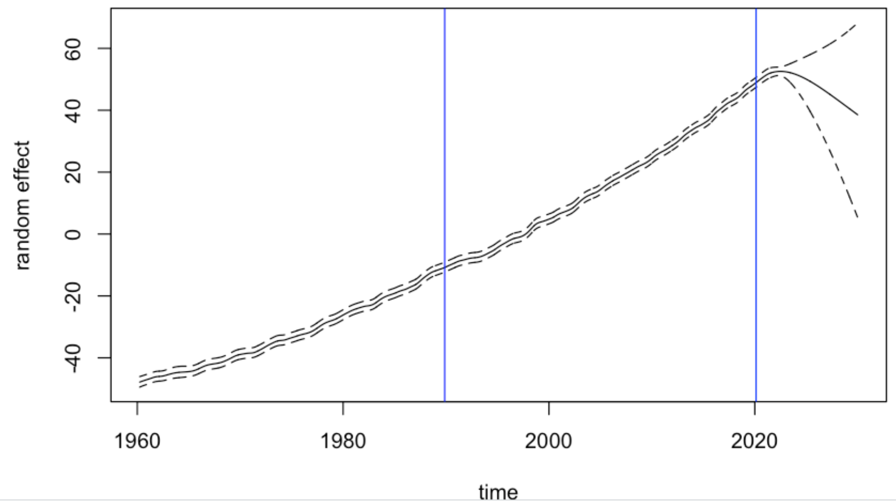


Figure5: Random effect of CO2 along time



Based on these observations, I decided to use INLA to make a GAM model. Furthermore poisson describes discrete random variables but our variable is continuous, therefore we assumed the expected CO2 concentration following the gaussian distribution is reasonable. Then I established a model as follows.

$$\mu_i = \beta_0 + X_i\beta + U_{ti}$$

where μ_i is the expected CO2 Concentration at Mauna Loa Observatory, Hawaii
X_i represents the fixed effect, which is four covariates defined by trigonometric functions for seasonal patterns for corresponding date i. Those are: sin12, cos12, sin6, cos6, where

$$cos12 = cos(2 * \pi * timeYears)$$

$$sin12 = sin(2 * \pi * timeYears)$$

$$cos6 = cos(2 * 2 * \pi * timeYears)$$

$$sin6 = sin(2 * 2 * \pi * timeYears)$$

where $timeYears$ is a variable that represents the days passed if we assume 1970-01-01 as the origin and 365.25(a year) as one unit.
β_0 represents the intercept when X_i is 0.
β represents the corresponding parameters of X_i.
U_{ti} represents RW(2). Bayesian inference is applied to the model for U_{ti}. Hence

$$U_{t+i}|U_k, k < t \sim N(-2U_t + U_{t-1}, \sigma_U^2)$$

$$U_{t+1} - 2U_t + U_{t+1} \sim N(0, \sigma_U^2)$$

that is

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

And if we applied prior, then $prob(\sigma_U > 0.001) = 0.5$
Assume Y_i, which is the expected CO2 concentration for day i. Then we have:

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

And if we applied prior, then $prob(\sigma > 1) = 0.5$

This model can also be written as

$$\mu_i = \beta_0 + \beta_1 X_{sin12} + \beta_2 X_{cos12} + \beta_3 X_{sin6} + \beta_4 X_{cos6} + U_{ti} + V_i$$

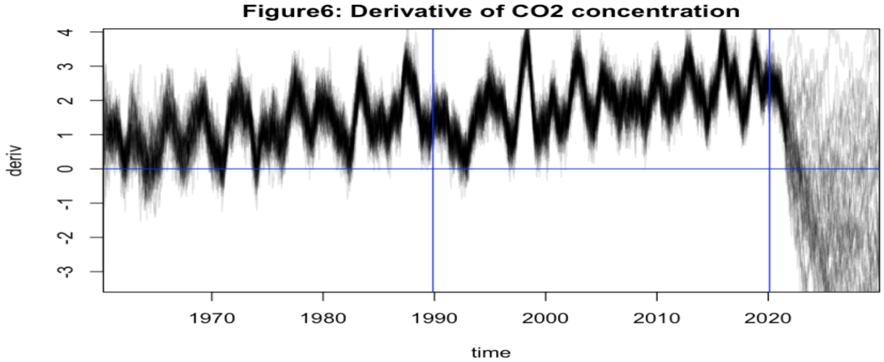
After building the model, we would analyze it by summarizing the model and plotting the distribution of the Random effect of CO2 along all the time and Derivative of CO2 concentration.

Result

Figure5 shows the Random effect of CO2 concentration all the time, which describes the random slope. We could find that the random effect at each time is different and it showed an overall upward trend, but in 1989, when the fall of Berlin Wall occured, there is no obvious change at that time point. Also this year is 2021, we are still experiencing the global lockdown during COVID-19. We could see that after 2020, the range is the confidence interval of predication. This reveals that there might be a negative impact on economic development in recent years.

Figure6 is the derivative of CO2 concentration all the time. We could observe that the derivatives in both 1989-11 and 2020-02 are positive, which indicates that their slope is positive, then CO2 concentration at 1989-11 and 2020-02 is increasing. However the second derivative of 1989-11 and 2020-02 are negative, which demonstrates that although the overall trend is an upward trend, when these two major events happened, the increasing rate was not as fast as before. Also, from Figure6 we could observe that the derivative in 1987 is around 4, while in 1989 is around 2. The derivative in 2018 is 4, while in 2020 it is around 2. This also implies that the increase of CO2 concentration is decreasing. Furthermore, from 1989 to 1993, the derivative of CO2 concentration is keep decreasing and it reaches below 0 on 1993, that might indicates that after the fall of berlin wall, the dramatics fall in industrial production in the Soviet Union and Eastern Europe had lasted about 3 years or even more, and it leads to a decrease of CO2 concentration. Besides, now we are experiencing a global lockdown during COVID-19, we predicted that the derivative would also reach below 0, this might imply after this event, CO2 concentration would also fall down.

Figure6: Derivative of CO2 concentration



In conclusion, we could determine that there is an impact on CO2 concentration after the fall of Berlin Wall and global lockdown during covid-19. And the impact of the fall of the Berlin Wall may last about 3 years or even more, and there might be a negative impact on economic development after the global lockdown during COVID-19.

Question 2

The Change in Mortality in Ontario for Heart disease, Neoplasms, Accidents and Respiratory disease after COVID-19

Introduction

Humans have always been plagued by diseases. Therefore, people keep improving their medical technology. However, the arrival of COVID-19 caused a global lockdown. The diseases people had during this period have also been changed. Hence, I made two hypotheses. During the COVID-19 global lockdown, the mortality of Malignant neoplasms (cancers) and Diseases of the heart (heart attacks) would increase because people do not have many opportunities to hospital for their healthcare. And the second hypothesis is the mortality of Accidents (unintentional injuries), and chronic lower respiratory diseases would decrease because of masks, lack of social activities and better air quality. Hence I analyzed Daily cause-specific mortality counts by province are available from Statistics Canada at https://open.canada.ca/data/en/dataset/aed00edc-26ad-414c-8aa3-82212059ef8a.

To test these two hypotheses, I plot the outcomes of the four types from 2020-01-09 to 2020-11-28. And I fitted a model to the pre-COVID data and used the model to predict the mortality of 2020-03 to 2020-11 in order to determine the change in mortality of the four types during COVID-19.

Method

First, I plot the outcomes of heart disease, neoplasms, accidents and respiratory along all the time. And from Figure1-4, we could observe that there might be a seasonal trend in all four plots. Thus, we add cosine and sine terms in our model as seasonal trending. But the overall trend after 2020-02 (when the COVID-19 occured) are different. The trend of mortality of heart disease(Figure 1) after 2020-02 is slightly increasing as well as the neoplasms (Figure2), while the trend of mortality of chronic lower respiratory(Figure3) diseases has a decreasing trend. Besides, in Figure4, the mortality of accidents has an obvious upward trend during COVID-19.

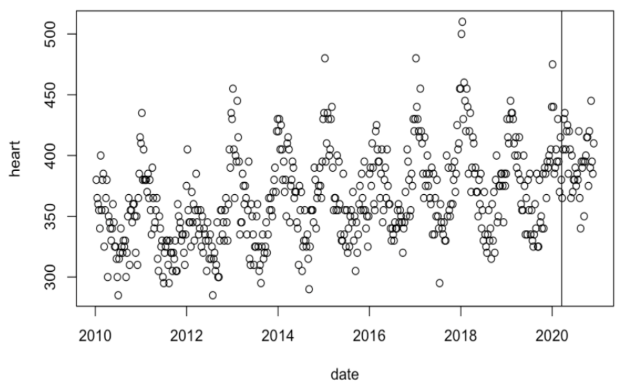


Figure 1: Outcomes of Heart Diseases

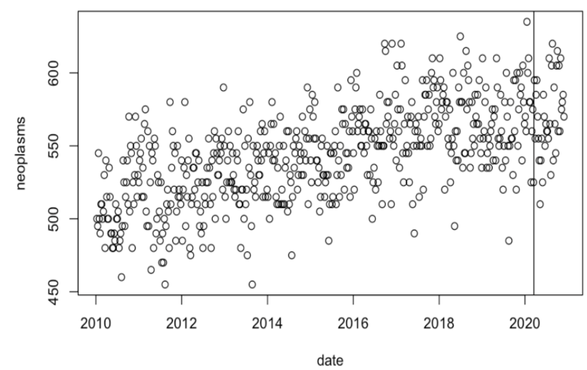


Figure 2: Outcomes of Neoplasms

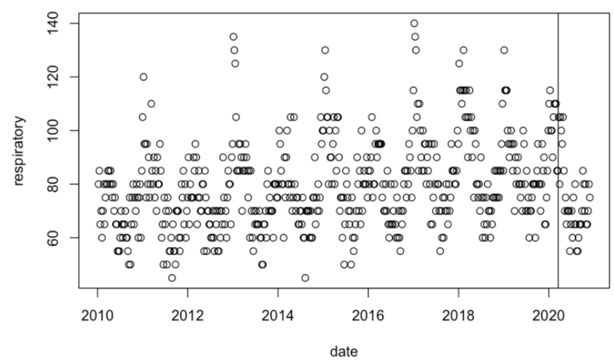


Figure 3: Outcomes of Respiratory

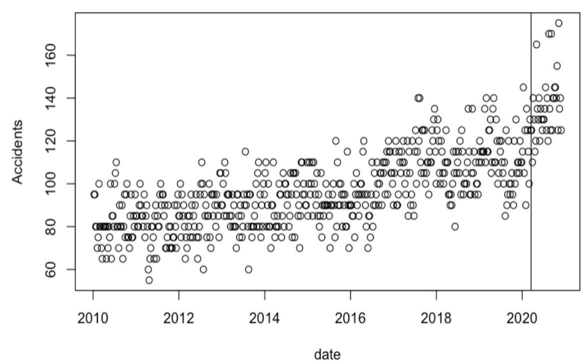


Figure 4: Outcomes of Accidents

Furthermore, we could also observe that the random effect of the four types on each time point (Figure 6, 10, 14, 18) is different, which demonstrates that the random slope along all the time has a different value.

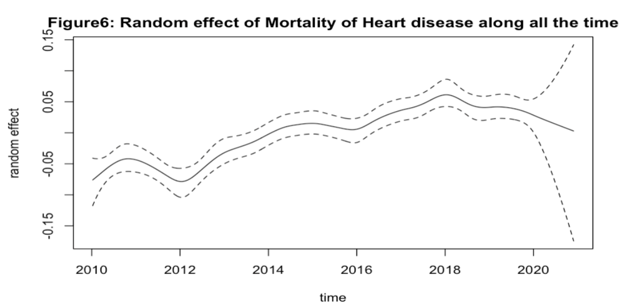


Figure 6: Random Effect of Mortality of Heart Disease

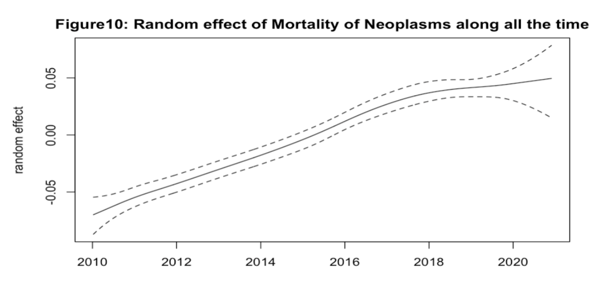


Figure 7: Random Effect of Mortality of Neoplasms

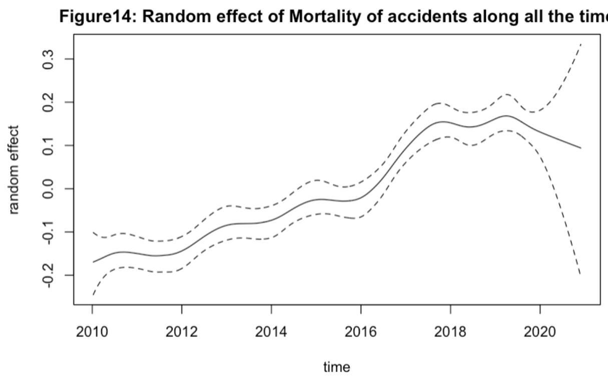


Figure 14: Random Effect of Mortality of Accidents

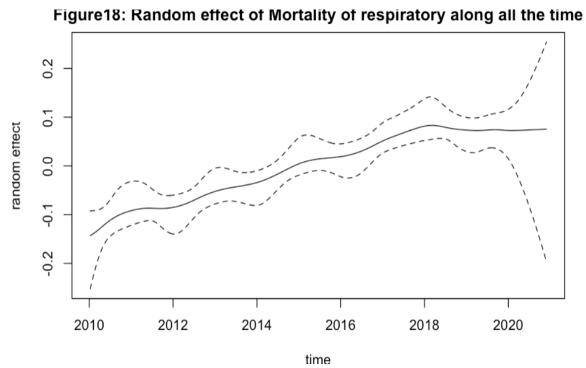


Figure 18: Random Effect of Mortality of Respiratory

Based on these observations, I decided to use INLA to make a GAM model based on the pre COVID-19 situation. Furthermore, poisson describes discrete and non-negative random variables, therefore we assume the mortality following the poisson distribution is reasonable. Then I established a model as follows.

$$\log(\mu_i) = \beta_0 + X_i\beta + U_{ti} + V_i$$

where μ_i is the expected mortality on day i.

X_i represents the fixed effect, which is four covariates defined by trigonometric functions for seasonal patterns for corresponding date i. Those are: sin12, cos12, sin6, cos6

$$cos12 = cos(2 * \pi * dateInt/365.25)$$
$$sin12 = sin(2 * \pi * dateInt/365.25)$$
$$cos6 = cos(2 * 2 * \pi * dateInt/365.25)$$
$$sin6 = sin(2 * 2 * \pi * dateInt/365.25)$$

where $dateInt$ is a variable that represents the integer of past days if we assume 1970-01-01 as the origin.

β_0 represents the intercept when all of the explanatory variables X_i are equal to 0

β represents the corresponding parameters of X_i.

U_{ti} represents dateInt, which is RW(2). Bayesian inference is applied to the model for U_{ti}. Hence

$$U_{t+i}|U_k, k < t \sim N(-2U_t + U_{t-1}, \sigma_U^2)$$
$$U_{t+1} - 2U_t + U_{t+1} \sim N(0, \sigma_U^2)$$

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

And if we applied prior, then $prob(\sigma_U > 0.1) = 0.5$

V_i represents the dateIid, which is random effects, with $V_i \sim N(0, \sigma_v^2)$

And if we applied prior, then $prob(\sigma_v > log(1.25)) = 0.5$

Assume Y_i, which is the expected mortality for day i. Then we have:

$$\log(Y_i) \sim poisson(\mu_i)$$

This model can also be written as

$$\log(\mu_i) = \beta_0 + \beta_1 X_{sin12} + \beta_2 X_{cos12} + \beta_3 X_{sin6} + \beta_4 X_{cos6} + U_{ti} + V_i$$

After building the model, we would analyze it by summarizing the model and plotting the prediction of Mortality and excess Mortality after COVID-19 of the four types.

Result

The model we used is based on the pre COVID-19 data, and used that model to predict the situation during COVID-19. In order to determine the accuracy of our model in all four types, we could look at Figure 5, 9,13,17.

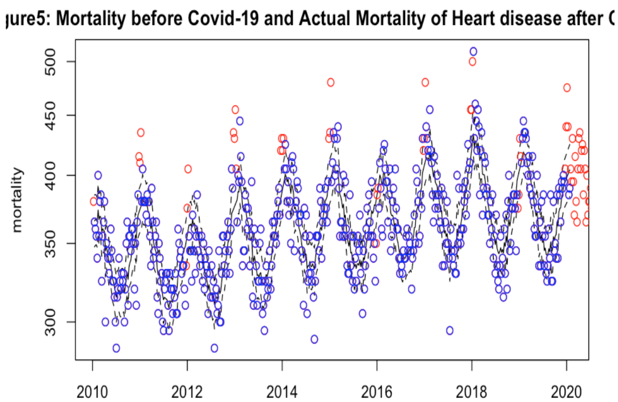


Figure 5

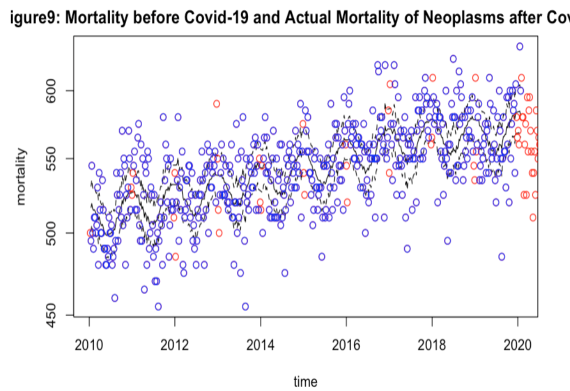


Figure 9

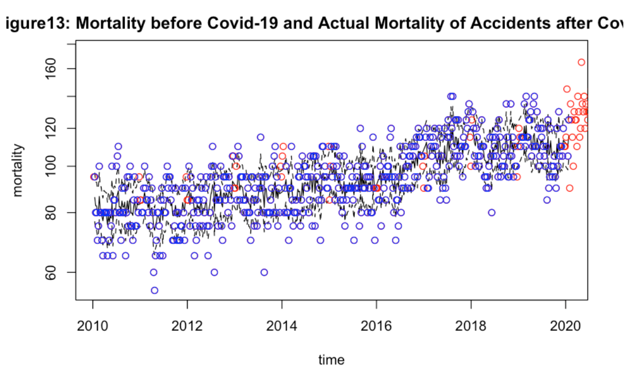


Figure 13

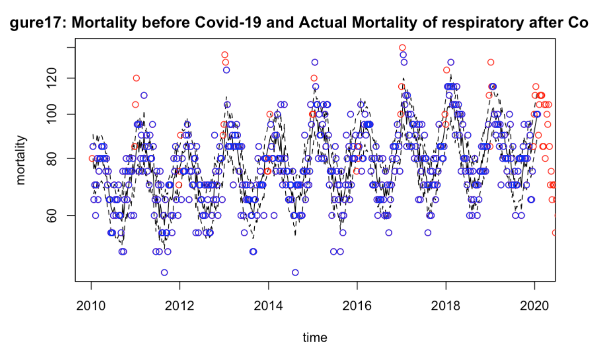


Figure 17

Figure 5 shows the distribution of the outcomes of our fitted model in heart diseases and the actual mortality of Heart disease all the time. Red points represent the actual mortality of Heart disease, while the blue points represent the mortality of Heart disease before the COVID-19. From this Figure, we could find that most of the blue points are located between the dotted lines, which is the 95% prediction interval. And also there is a repeating trend of rising and then falling in the distribution of the model(black line). Both observations lead to the fitted model about heart diseases seems to be reasonable.

Figure 9 shows the distribution of the outcomes of our fitted model in Neoplasms and the actual mortality of Neoplasms all the time. Red points represent the actual mortality of Neoplasms, while the blue points represent the mortality of Neoplasms before the COVID-19. We could observe the same result as Figure 5,that is most of the blue points are located between the dotted lines, which is the 95% prediction interval. And also there is a repeating trend of rising and then falling in the distribution of the model(black line). Both observations lead to the fitted model about Neoplasms seems to be reasonable.

By repeating the same process, Figure 13 and Figure 17 would lead to a similar result. Figure 13 shows the distribution of the outcomes of our fitted model in Accidents and Figure 17 shows about respiratory. It illustrates that our fitted model about Accident and Respiratory seems to be reasonable.

In order to determine whether the prediction is accurate, we could have a look at Figure 7, 11, 15, 19, which represent the prediction about heart disease, neoplasms, accidents and respiratory diseases respectively. Based on the pre COVID-19 data, we create an ensemble of 100 different forecasts for March-November. From Figure 8 and Figure 15, we could observe that the red points located after 2020-02 are generally higher than our prediction. That demonstrates that our prediction does not match the actual data. And there might exist an excess mortality of Heart diseases and Accidents. However, in Figure 11 and 19, most of the red points are located at our prediction distribution. Hence we could not determine whether the difference between our prediction and real data on neoplasms and respiratory diseases exists.

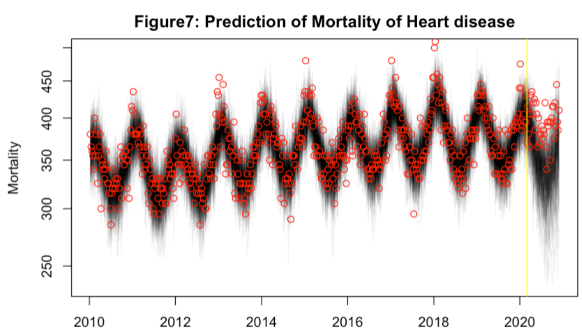


Figure7: Prediction of Mortality of Heart Disease

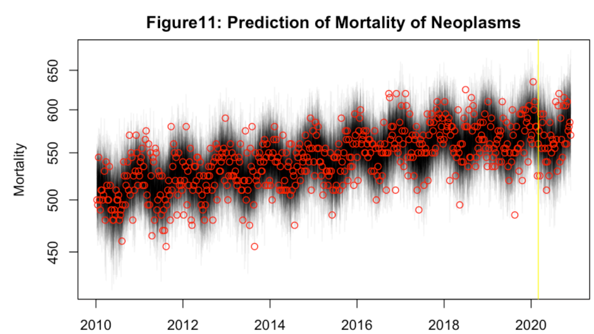


Figure11: Prediction of Mortality of Neoplasms

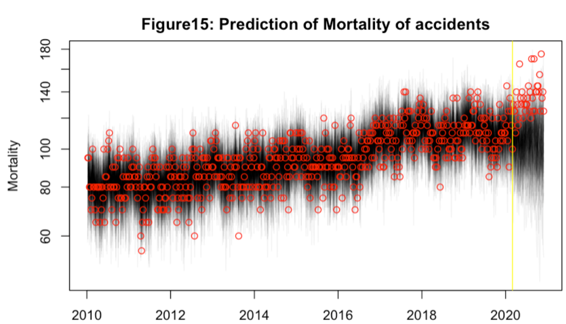


Figure15: Prediction of Mortality of Accidents

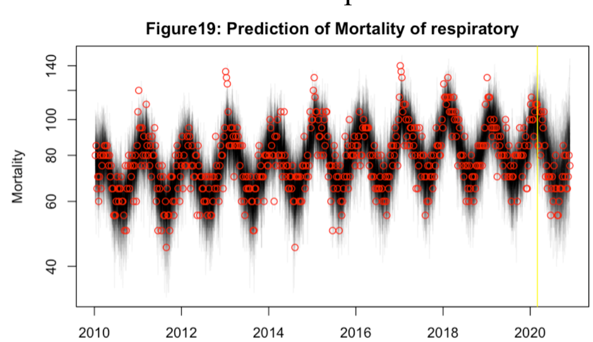


Figure19: Prediction of Mortality of Respiratory Disease

To be more specific, We could plot the excess mortality to see if the number of deaths is increasing or decreasing. Figure 8, 12, 16, 20 represents heart diseases, neoplasms, accidents and respiratory diseases respectively.

From Figure8 and Figure 16, we could see that the interval between the dotted line corresponding to most of the time points is above the baseline (0), which demonstrates that the mortality of heart diseases and Accidents is indeed increasing. This further proves that there exists excess mortality of heart diseases and Accidents, and the number of deaths is further increasing during COVID-19 period. While from Figure 20, the intervals between the dotted line corresponding to most of the time points are located below the baseline, which demonstrates that the number of deaths because of respiratory diseases is decreasing. Besides, from Figure 12, the fluctuation of the interval and baseline almost coincide, which demonstrates that the actual data is consistent with the predicted trend. There might be no excess or deficit of mortality.

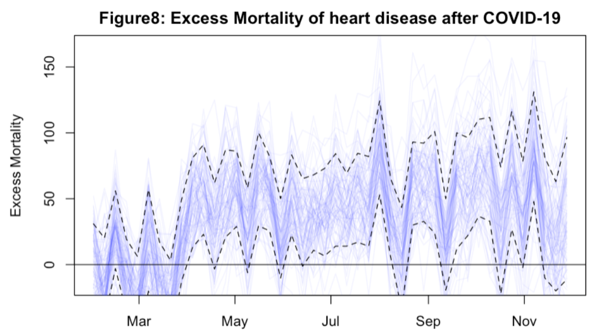


Figure 8: Excess Mortality of Heart Disease After COVID-19

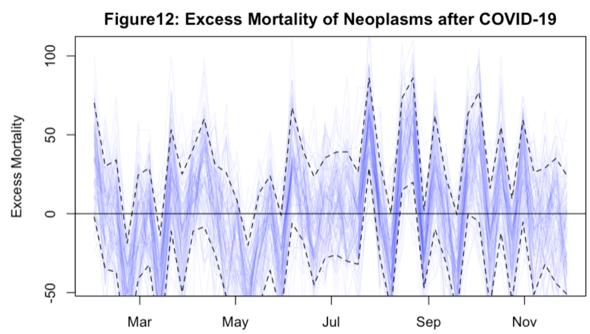


Figure 12: Excess Mortality of Neoplasms After COVID-19

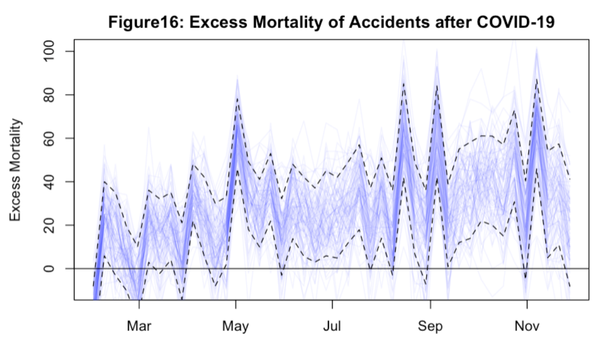


Figure 16: Excess Mortality of Accidents After COVID-19

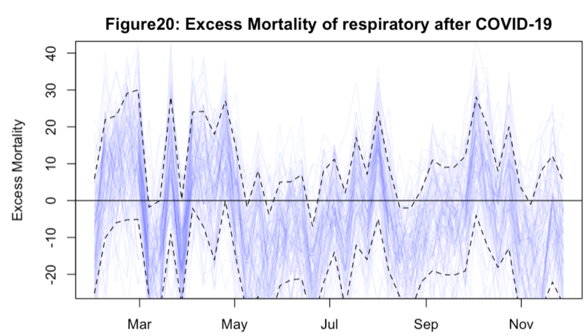


Figure 20: Excess Mortality of Respiratory Disease After COVID-19

In conclusion, during the COVID-19 global lockdown, (1) the mortality of Diseases of the heart (heart attacks) is increasing which is consistent with our hypothesis. (2) the mortality of chronic lower respiratory diseases is decreasing which is consistent with our hypothesis. However, (3)there is no excess or deficit of the mortality of Malignant neoplasms (cancers), which is inconsistent with our hypothesis. It might be because the time of treatment cycle is relatively long, so it takes a long time from the discovery to the appearance of death. Perhaps in the future, the actual mortality of neoplasms would occur. Furthermore, (4) the mortality of accidents is increasing which is inconsistent with our hypothesis. The deficit of mortality exists. That might be because the global lockdown during COVID-19 would affect human's characters and their actions that might cause unintentional injuries.

Appendix

Question1

```
```{r}
cUrl = paste0("http://scrippsco2.ucsd.edu/assets/data/atmospheric/",
"stations/flask_co2/daily/daily_flask_co2_mlo.csv")
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", "junk1", "junk2", "Nflasks", "quality",
"co2"))
co2s$date = as.Date(co2s$day)
co2s$time = strptime(paste(co2s$day, co2s$time), format = "%Y-%m-%d %H:%M",
tz = "UTC")
# remove low-quality measurements
co2s = co2s[co2s$quality == 0, ]

#CO2 at Mauna Loa Observatory, Hawaii over all years
plot(co2s$date, co2s$co2, log = "y", cex = 0.3, col = "#00000040",
xlab = "time", ylab = "ppm", main="Figure1: CO2 at Mauna Loa Observatory, Hawaii over all years")

#CO2 at Mauna Loa Observatory, Hawaii in recent years
plot(co2s[co2s$date > as.Date("2015/3/1"), c("date",
"co2")], log = "y", type = "o", xlab = "time",
ylab = "ppm", cex = 0.5, main="Figure2: CO2 at Mauna Loa Observatory, Hawaii in recent years")

#define fixed effect
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)
# disable some error checking in INLA
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())

#GAM model
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 = list(prior = "pc.prec",
param = c(1, 0.5)))), control.inla = list(strategy = "gaussian"),
control.compute = list(config = TRUE), verbose = TRUE)
```
```

```
#summary table and quantiles for random effects
qCols = c("0.5quant", "0.025quant", "0.975quant")
1/sqrt(co2res$summary.hyperpar[, qCols])

matplot(co2res$summary.random$dateWeekInt[, qCols],
type = "l", lty = 1)
# source('https://bioconductor.org/biocLite.R')
# biocLite('Biobase')
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

#Overall Prediction of CO2 concentrations
matplot(allDays, forecast, type = "l", col = "#00000010",
lty = 1, log = "y", xlab = "time", ylab = "ppm")
title(main="Figure3: Overall Prediction of CO2 concentrations")
```

```
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))

#Prediction of CO2 concentrations in details
matplot(allDays, forecast, type = "l", col = "#00000020",
lty = 1, log = "y", xlab = "time", ylab = "ppm",
xaxs = "i", xaxt = "n", xlim = range(forX), ylim = range(forecast[which.min(abs(allDays -
max(forX))), ]))
title(main="Figure4: Prediction of CO2 concentrations in details")
points(co2s$date, co2s$co2, col = "red", cex = 0.3)
axis(1, as.numeric(forX), format(forX, "%Y"))

#Random effect of CO2 along time in Hawaii
matplot(allDays, co2res$summary.random$dateWeekInt[,
qCols], type = "l", col = "black", lty = c(1, 2,
2), xlab = "time", ylab = "random effect")
title(main="Figure5: Random effect of CO2 along time")
abline(v = as.numeric(as.Date("1989-11-15")) , col = "blue")
abline(v = as.numeric(as.Date("2020-02-15")) , col = "blue")

#Derivative of CO2 Concentration along time
matplot(allDays[-1], sampleDeriv, type = "l", lty = 1,
xaxs = "i", col = "#00000020", xlab = "time", ylab = "deriv",
ylim = quantile(sampleDeriv, c(0.025, 0.995)))
```

```
#add two vertical lines with the time 1989-11-09 and 2020-02-11
abline(v = as.numeric(as.Date("1989-11-15")) , col = "blue")
abline(v = as.numeric(as.Date("2020-02-15")) , col = "blue")
abline(h = 0, col="blue")
title("Figure6: Derivative of CO2 concentration")
```

```
#Derivative of CO2 concentration in details
matplot(allDays[toPlot], sampleDeriv[toPlot, ], type = "l",
lty = 1, lwd = 2, xaxs = "i", col = "#00000020",
xlab = "time", ylab = "deriv", xaxt = "n", ylim = quantile(sampleDeriv[toPlot,
], c(0.01, 0.995)))
axis(1, as.numeric(forX), format(forX, "%Y"))
title("Figure7: Derivative of CO2 concentration in details")

```
```

Question 2:

```
```{r}
#deadFile=Pmisc::downloadIfOld("https://www150.statcan.gc.ca/n1/tbl/csv/13100810-eng.zip",path ="../data")
#(deadFileCsv =deadFile[which.max(file.info(deadFile)$size)])
x=read.csv("/Users/yaoyao/Downloads/X13100810.eng/13100810.csv")
x[1:2, ]
x$date = as.Date(as.character(x[[grep("DATE", names(x))]]))
x$province = gsub("[,].*", "", x$GEO)
# remove 2021 data, which appears incomplete
x = x[x$date < as.Date("2020/12/01") & x$province ==
"Ontario", ]

#Figure 1-4 Overall situation of Heart, neoplasms, Accidents and respiratory along the time
for (D in c("heart", "neoplasms", "respiratory", "Accidents")) {
plot(x[grep(D, x$Cause), c("date", "VALUE")], ylab = D)
  abline(v = as.Date("2020/03/17"))
}

#from 2010-01-09 to 2020-11-28, there are 568 weeks in total
dateSeq = sort(unique(x$date))
table(diff(dateSeq))

#deal with the fixed effect
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"))

#dead because of heart disease in Ontario
xSub = x[grepl("heart", x$Cause, ignore.case = TRUE) &
x$province == "Ontario", ]

#situation of heart disease before covid-19
xPreCovid = xSub[xSub$date < as.Date("2020/02/01") &
(!xSub$christmasBreak), ]
```

```
#GAM model
library("INLA")
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 ="poisson",control.compute =list(config =TRUE),
             control.predictor =list(compute =TRUE))

#red points represents the mortality of heart and blue represent the mortality of heart in pre covid-19
situation
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$VALUE),
xlab="time",ylab="mortality")
title("Figure5: Mortality before Covid-19 and Actual Mortality of Heart disease after Covid-19")
points(xSub$date, xSub$VALUE, col = "red")
points(xPreCovid$date, xPreCovid$VALUE, col = "blue")

#Random effect of Mortality along all the time
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")
title("Figure6: Random effect of Mortality of Heart disease along all the time")
```

```
#Prediction based on pre Covid-19
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(resSampleLambda))

matplot(dateSeq, resSampleCount, col = "#00000010",
type = "l", lty = 1, log = "y",
xlab="time",ylab = "Mortality")
points(xSub[, c("date", "VALUE")], col = "red")
abline(v = as.Date("2020/03/01"), col = "yellow")
```

```
title("Figure7: Prediction of Mortality of Heart disease")


is2020 = dateSeq[dateSeq >= as.Date("2020/2/1")]
sample2020 = resSampleCount[match(is2020, dateSeq),
]
count2020 = xSub[match(is2020, xSub$date), "VALUE"]
excess2020 = count2020 - sample2020
```

```
#excess mortality of heart disease
matplot(is2020, excess2020, type = "l", lty = 1, col = "#0000FF10",
        xlab="time",ylab = "Excess Mortality",
        ylim = range(-10, quantile(excess2020, c(0.1, 0.999))))
matlines(is2020, t(apply(excess2020, 1, quantile, prob = c(0.1,
0.9))), col = "black", lty = 2)
abline(h = 0)
title("Figure8: Excess Mortality of heart disease after COVID-19")
quantile(apply(excess2020, 1, sum))

```
```

```
```{r}
# neoplasms
xSub=x[grepl("neoplasms", x$Cause,ignore.case =TRUE)&
         x$province=="Ontario", ]
# death before the Covid
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 ="poisson",control.compute =list(config =TRUE),
             control.predictor =list(compute =TRUE))


#red points represents the mortality of neoplasms and blue represent the mortality of neoplasms in pre
covid-19 situation
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$VALUE),
xlab="time",ylab="mortality")
title("Figure9: Mortality before Covid-19 and Actual Mortality of Neoplasms after Covid-19")
points(xSub$date, xSub$VALUE, col = "red")
points(xPreCovid$date, xPreCovid$VALUE, col = "blue")
```

```
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")
title("Figure10: Random effect of Mortality of Neoplasms along all the time")


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(resSampleLambda))
```

```
matplot(dateSeq, resSampleCount, col = "#00000010",
type = "l", lty = 1, log = "y",
xlab="time",ylab = "Mortality")
points(xSub[, c("date", "VALUE")], col = "red")
abline(v = as.Date("2020/03/01"), col = "yellow")
title("Figure11: Prediction of Mortality of Neoplasms")


is2020=dateSeq[dateSeq>=as.Date("2020/2/1")]
sample2020=resSampleCount[match(is2020, dateSeq),]
count2020=xSub[match(is2020, xSub$date),"VALUE"]
excess2020=count2020-sample2020

#excess mortality of heart disease
matplot(is2020, excess2020, type = "l", lty = 1, col = "#0000FF10",
        xlab="time",ylab = "Excess Mortality",
        ylim = range(-10, quantile(excess2020, c(0.1, 0.999))))
matlines(is2020, t(apply(excess2020, 1, quantile, prob = c(0.1,
0.9))), col = "black", lty = 2)
abline(h = 0)
title("Figure12: Excess Mortality of Neoplasms after COVID-19")
quantile(apply(excess2020, 1, sum))

```
```

```
```{r}
# Accidents
xSub=x[grepl("accidents", x$Cause,ignore.case =TRUE)&
         x$province=="Ontario", ]
# death before the Covid
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 ="poisson",control.compute =list(config =TRUE),
             control.predictor =list(compute =TRUE))


#red points represents the mortality of accidents and blue represent the mortality of accidents in pre
covid-19 situation
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$VALUE),
xlab="time",ylab="mortality")
title("Figure13: Mortality before Covid-19 and Actual Mortality of Accidents after Covid-19")
points(xSub$date, xSub$VALUE, col = "red")
points(xPreCovid$date, xPreCovid$VALUE, col = "blue")

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")
title("Figure14: Random effect of Mortality of accidents along all the time")

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(resSampleLambda))

matplot(dateSeq, resSampleCount, col = "#00000010",
type = "l", lty = 1, log = "y",
xlab="time",ylab = "Mortality")
points(xSub[, c("date", "VALUE")], col = "red")
abline(v = as.Date("2020/03/01"), col = "yellow")
title("Figure15: Prediction of Mortality of accidents")


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",
        xlab="time",ylab = "Excess Mortality",
        ylim = range(-10, quantile(excess2020, c(0.1, 0.999))))
matlines(is2020, t(apply(excess2020, 1, quantile, prob = c(0.1,
0.9))), col = "black", lty = 2)
abline(h = 0)
title("Figure16: Excess Mortality of Accidents after COVID-19")
quantile(apply(excess2020, 1, sum))
```

```{r}
# respiratory
xSub=x[grepl("respiratory", x$Cause,ignore.case =TRUE)&
         x$province=="Ontario", ]
# death before the Covid
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 ="poisson",control.compute =list(config =TRUE),
             control.predictor =list(compute =TRUE))


#red points represents the mortality of respiratory and blue represent the mortality of respiratory in pre
covid-19 situation
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$VALUE),
xlab="time",ylab="mortality")
title("Figure17: Mortality before Covid-19 and Actual Mortality of respiratory after Covid-19")
points(xSub$date, xSub$VALUE, col = "red")
points(xPreCovid$date, xPreCovid$VALUE, col = "blue")
```

````
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")
title("Figure18: Random effect of Mortality of respiratory along all the time")

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))
dateIdSeq=paste0("dateId:",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[dateIdSeq,]})
resSampleFitted=do.call(cbind, resSampleFitted)
resSampleLambda=exp(resSampleFitted)
resSampleCount=matrix(rpois(length(resSampleLambda),resSampleLambda),
                      nrow(resSampleLambda),ncol(resSampleLambda))

matplot(dateSeq, resSampleCount, col = "#00000010",
type = "l", lty = 1, log = "y",
xlab="time",ylab = "Mortality")
points(xSub[, c("date", "VALUE")], col = "red")
````

````
abline(v = as.Date("2020/03/01"), col = "yellow")
title("Figure19: Prediction of Mortality of respiratory")

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",
        xlab="time",ylab = "Excess Mortality",
        ylim = range(-10, quantile(excess2020, c(0.1, 0.999))))
matlines(is2020, t(apply(excess2020, 1, quantile, prob = c(0.1,
0.9))), col = "black", lty = 2)
abline(h = 0)
title("Figure20: Excess Mortality of respiratory after COVID-19")
quantile(apply(excess2020, 1, sum))
```
````
````