HW3

nan Jiang

2020/11/18

CO2

Introduction

This data set contains the atmospheric Carbon Dioxide concentrations from an observatory in Haiwaii, made available by the Scrips CO2 Program at scrippsco2.ucsd.edu starting from 1960/08/18 to recently. The data set is available at http://scrippsco2.ucsd.edu/assets/data/atmospheric/ (http://scrippsco2.ucsd.edu/assets/data/atmospheric/)“,”stations/flask_co2/daily/daily_flask_co2_mlo.csv. We want to discuss the change the CO2 data impacted by the two events. The fall of the Berlin wall in November 1989 years ago, preceded a dramatic fall in industrial production in the Soviet Union and Eastern Europe. And the global lockdown during the COVID-19 pandemic starting in February 2020, shut down much of the global economy.

method

We use a INLA method to create a generalized additive model. The cos and sin function is the seasonal effect by consider yearly

$$Y_i \sim \Gamma(\lambda_i, \theta_i)$$

$$log(\lambda_i) = X_i\beta + f(t_i)$$

$$X_{i0} = 1$$

$$X_{i1} = sin(2\pi t_i/365.25)$$

$$X_{i2} = cos(2\pi t_i/365.25)$$

$$X_{i3} = sin(4\pi t_i/365.25)$$

$$X_{i4} = cos(4\pi t_i/365.25)$$

$$f(t_i, \sigma) \sim RW2(\varphi)$$

$$\sigma \sim exp\ given\ pr(\sigma > 0.1) = 0.5$$

The parameter of gamma follows a pc prior. Take 30 posterior samples of th time inla random effect, which is also the random walk.

Result

These are the confidence interval standard residuals for the gamma and time inla. Since they do not include 0. The difference is significant.

```
## Loading required package: Matrix
## Loading required package: sp
```

```
## Loading required package: parallel
```

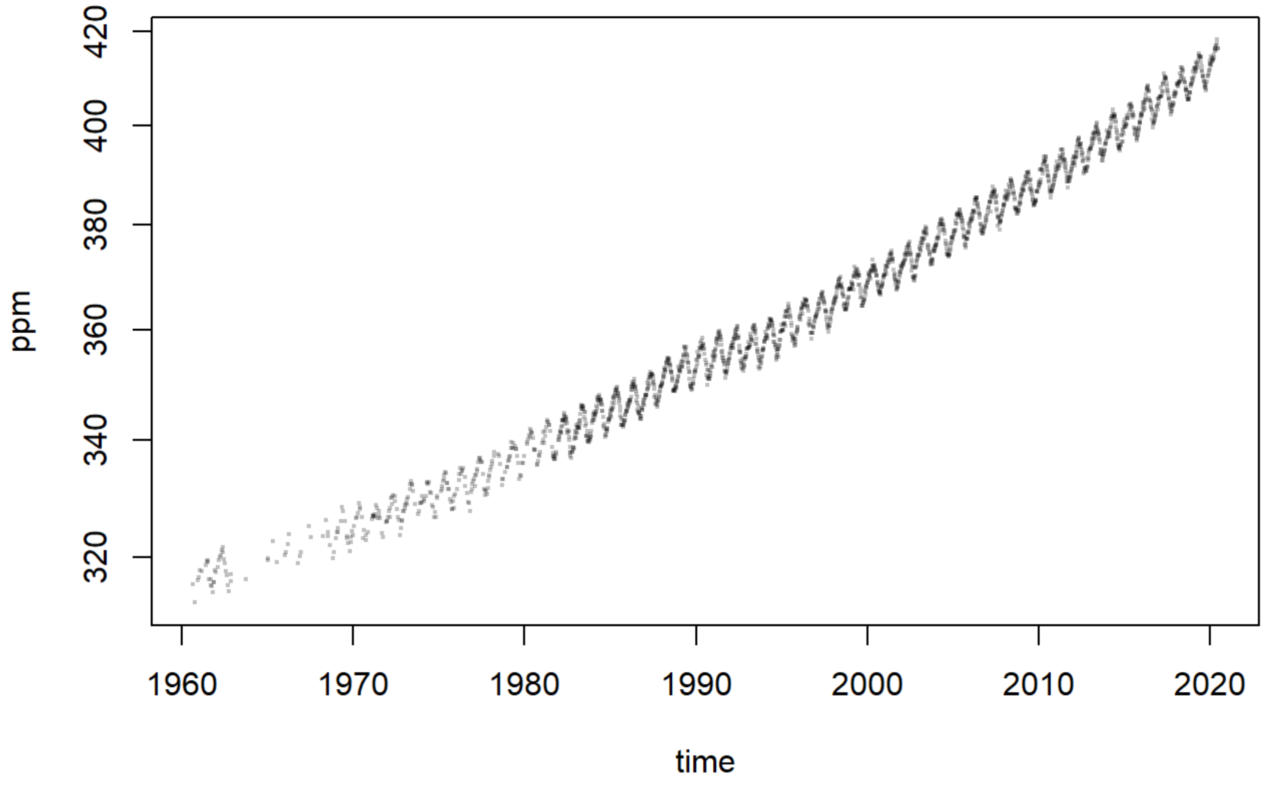
```
## Loading required package: foreach
```

```
## This is INLA_20.03.17 built 2020-11-17 18:20:08 UTC.
## See www.r-inla.org/contact-us for how to get help.
```

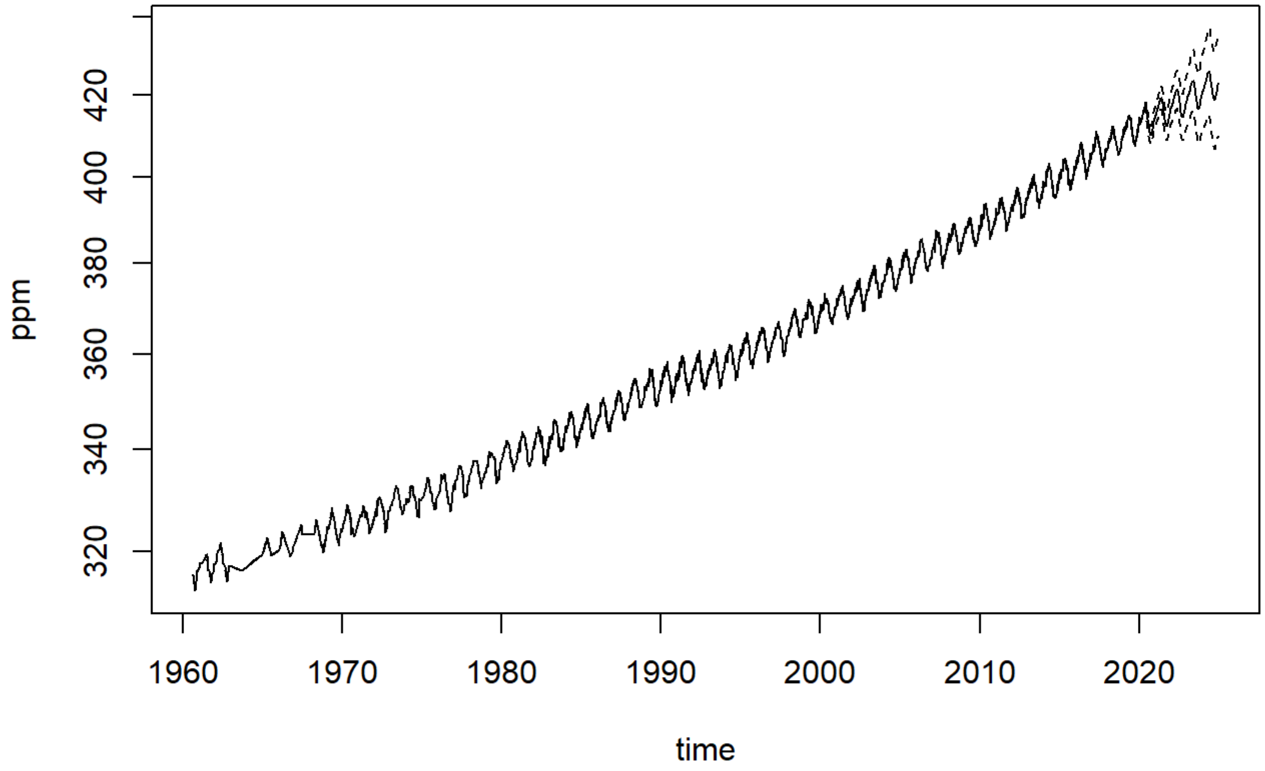
	0.5quant	**0.025quant**	**0.975quant**
	<dbl>	<dbl>	<dbl>
sd for gamma	5.807904e-05	1.146583e-05	0.0001692507
sd for timeInla	2.119757e-03	1.521756e-03	0.0028380653

2 rows

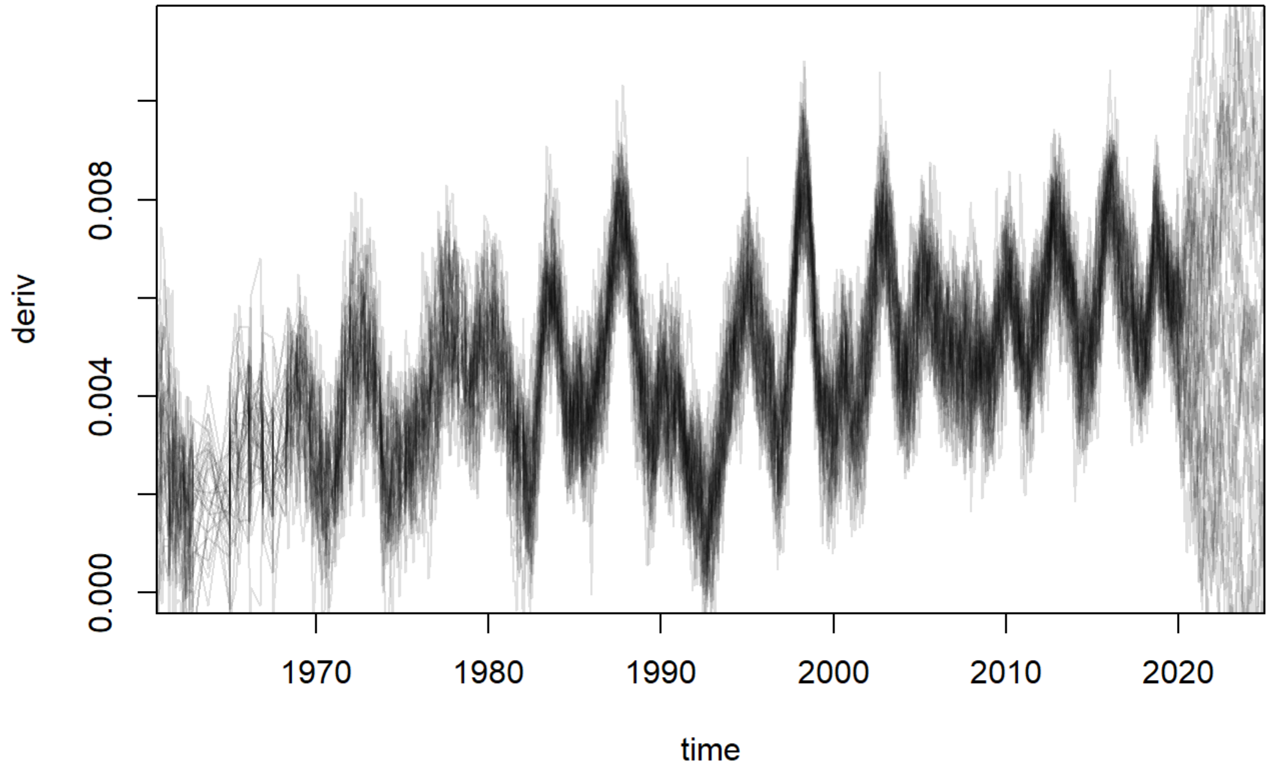
The following is the CO2 concentration graph. The CO2 has an increasing pattern with a seasonal fluctuation. We can see the increasing stops for a while around 1990, to see more clear evidence, we need to look at the first derivative graph, which is the increasing speed of the CO2 Concentration.

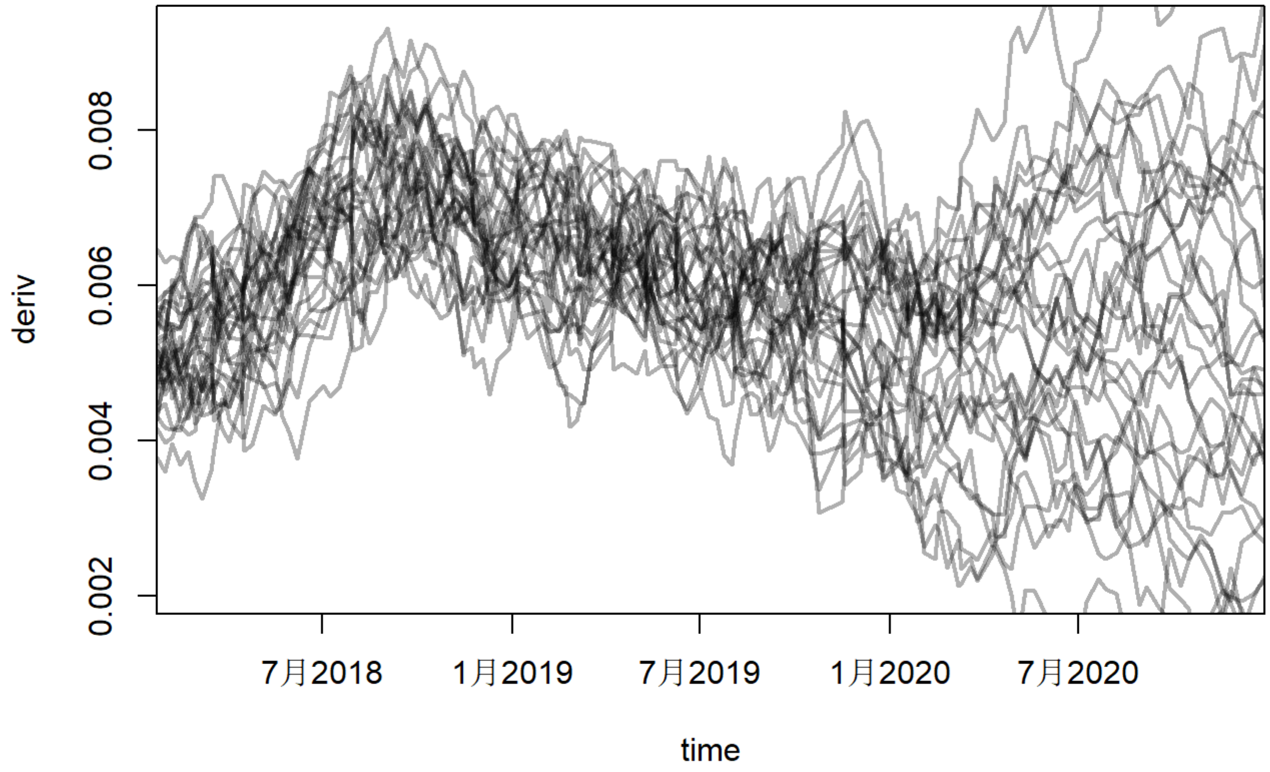


Without pridiction to 2025



with pridiction to 2025





This two graph are the first derivative graph for the year since 1960 to 2020 and then after February 2020, the data is from the prediction with the model we used.

The curve is around 1990 since the Soviet Union collapsed and the Eastern blocks their economy. Their major economy is based on heavy industry, such that the speed of the increasing of the CO2 concentration

significantly decreased. Such that from the previous graph, we can see the concentration curve is flatten around 1990, which is also the Berlin wall fall event.
The COVID-19 starting in February 2020 conducts the global lockdown. The global lockdown shuts down a lot of company which results the weaken of global economy. We want to use the previous year data to make a prediction on the effect of the CO2 concentration. There should be a decreasing trend based on the seasonal effect. The data shows a large variance on the change of the CO2 concentration. The speed could increase, decrease or keep the same in the following time periods. We do not have enough evidence to make an conclusion. Need to wait for more times to collect more data.

Conclusion

From the data in the Haiwaii, we can not make forecast of the effect of the COVID-19 event on the CO2 concentrations since it is an accident that never happened before. We need more data to make the actual prediction. For the berlin wall event and Soviet Union collapsing, we can see the decreasing of the speed of increasing of the CO2 concentration.

Death

Introduction

The data is from www.stat.gouv.qc.ca/statistiques/populationdemographie/deces-mortalite/nombre-hebdomadaire-deces_an.html. It show the daily mortality counts in Quebec. We want to find out the effective population in different time period for the COVID-19 virus. We have two hypothesis. Firstly, the first wave in March, April and May primarily affected the elderly. The second wave began in September is caused by irresponsible young people, mainly university undergraduates.

Method

We built two model to estimate the performance and make a forecast the future performance of the mortality of people under 50’s and people above 70’s. We use a INLA method to create a generalized additive model. The cos and sin function is the seasonal effect by consider yearly. For the under 50s model, Yi means the death number for the under 50s. For the over 70s model, Yi means the death number for the over 70s. But the basic model is the same.

$$Y_i \sim Possion(\lambda_i)$$

$$log(\lambda_i) = X_i\beta + f(t_i)$$

$$X_{i0} = 1$$

$$X_{i1} = sin(2\pi t_i/365.25)$$

$$X_{i2} = sin(4\pi t_i/365.25)$$

$$X_{i3} = cos(\pi t_i/365.25)$$

$$X_{i4} = cos(4\pi t_i/365.25)$$

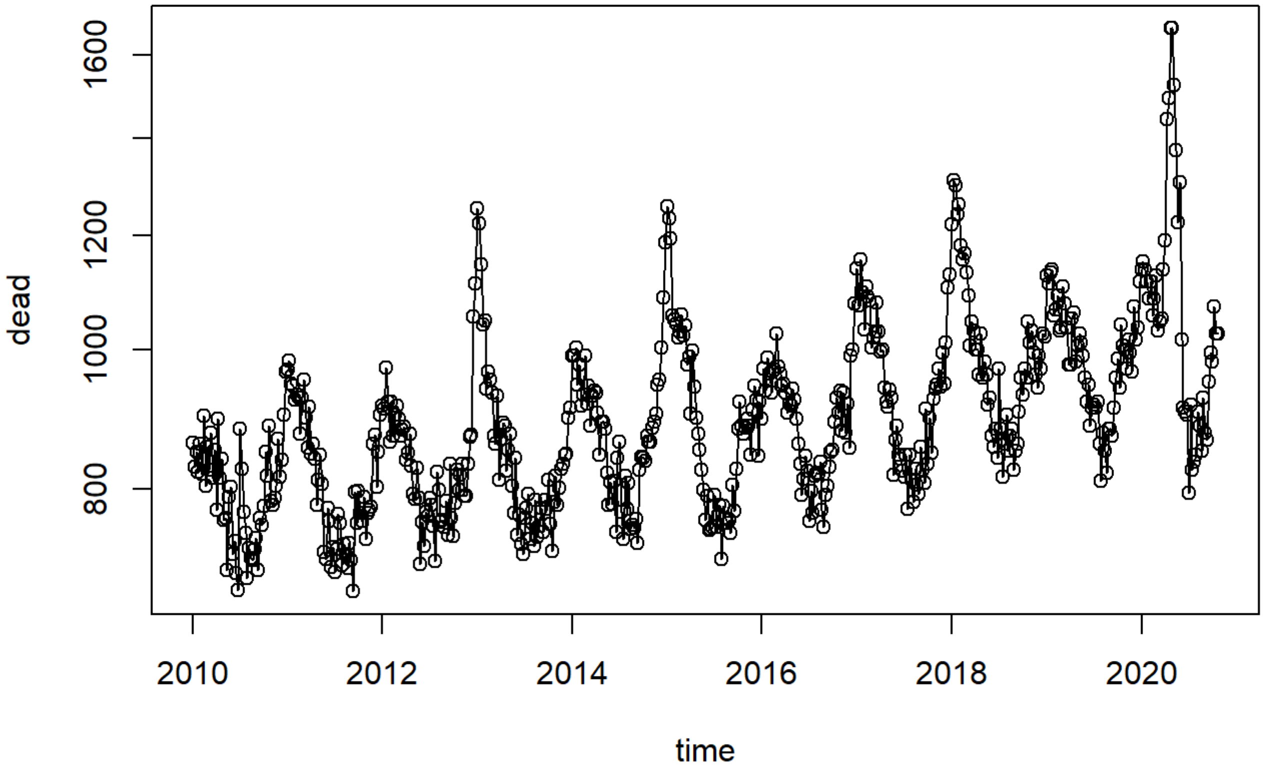
$$f(t_i, \sigma) \sim RW2(\varphi)$$

$$\sigma \sim exp\, given\, pr(\sigma > 0.1) = 0.5$$

Result

70s and over 70s

For the first hypothesis, from the mortality data, we made a graph about the death over 70 diagram. We can see there exist a seasonal trend and an increasing trend for the death number of older in Quebec. Such that we use an random effect and a seasonal factor to justify the hypothesis.

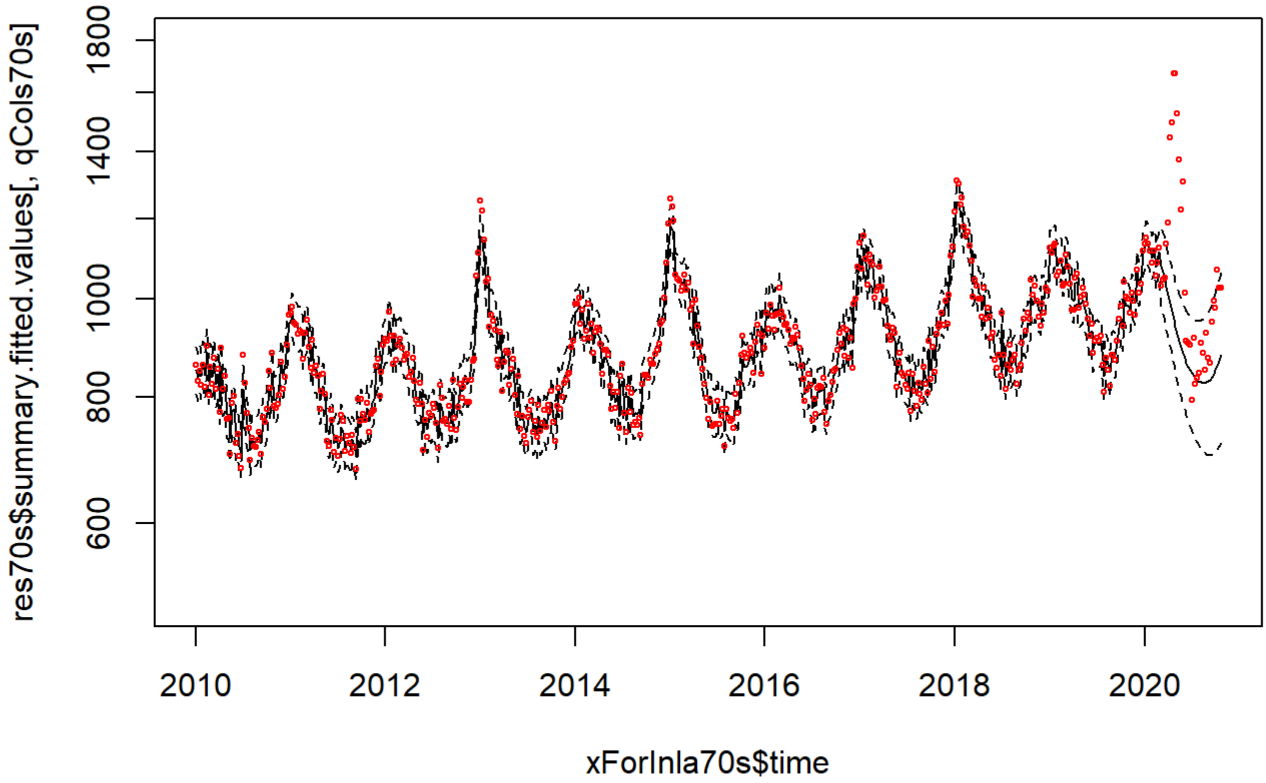


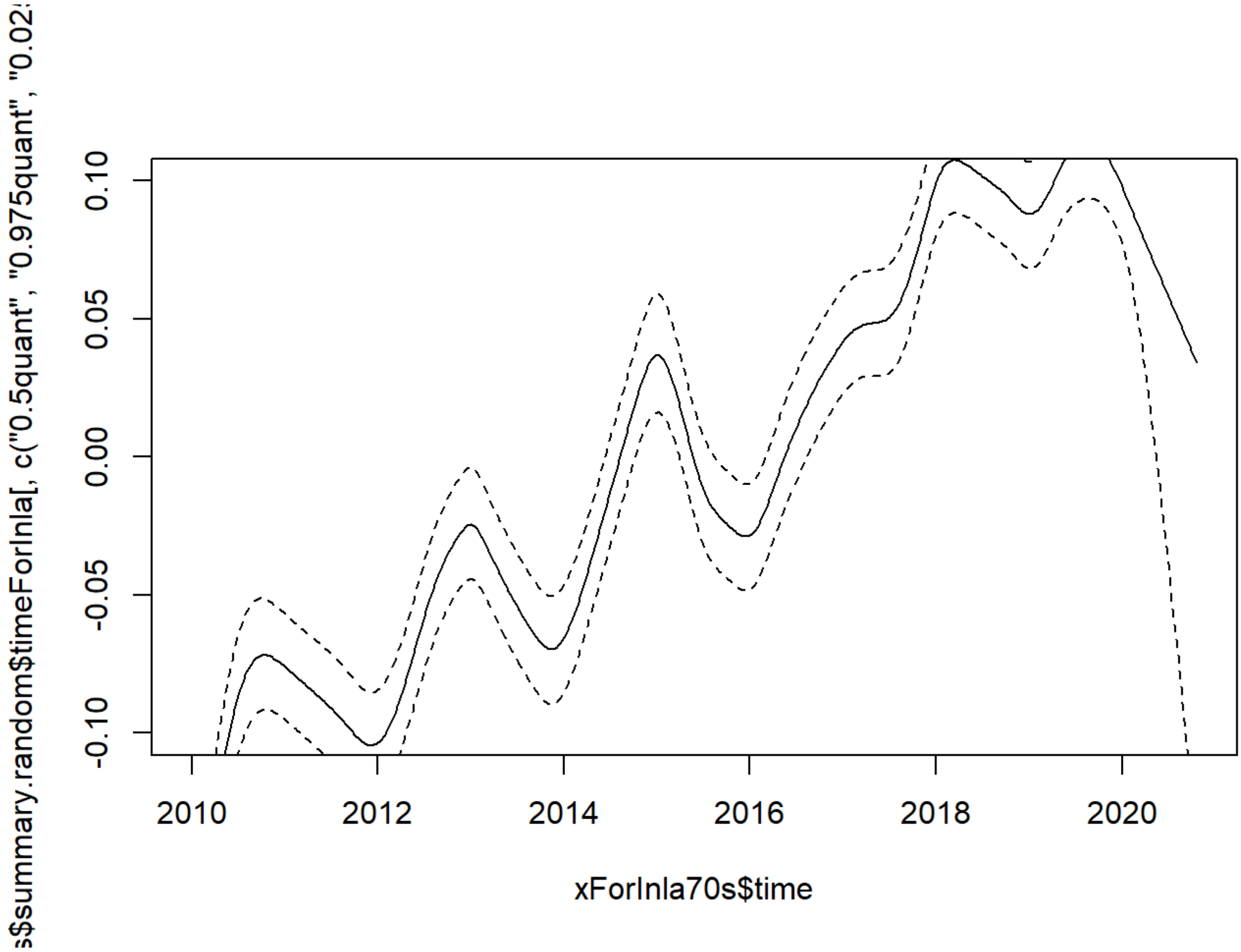
These are the parameter we estimated using the model. Since the confidence interval do not include 0. So they are all significant.

	0.5quant	**0.025quant**	**0.975quant**
	<dbl>	<dbl>	<dbl>
(Intercept)	6.78520152	6.777585166	6.79251435
sin12	0.06367218	0.056174106	0.07121570
sin6	0.01138859	0.004902609	0.01786380
cos12	0.11701055	0.109481387	0.12460245
cos6	0.01182358	0.005329909	0.01829593
SD for timeId	0.04050363	0.036336289	0.04504228
SD for timeForInla	0.16066477	0.106898939	0.23985700

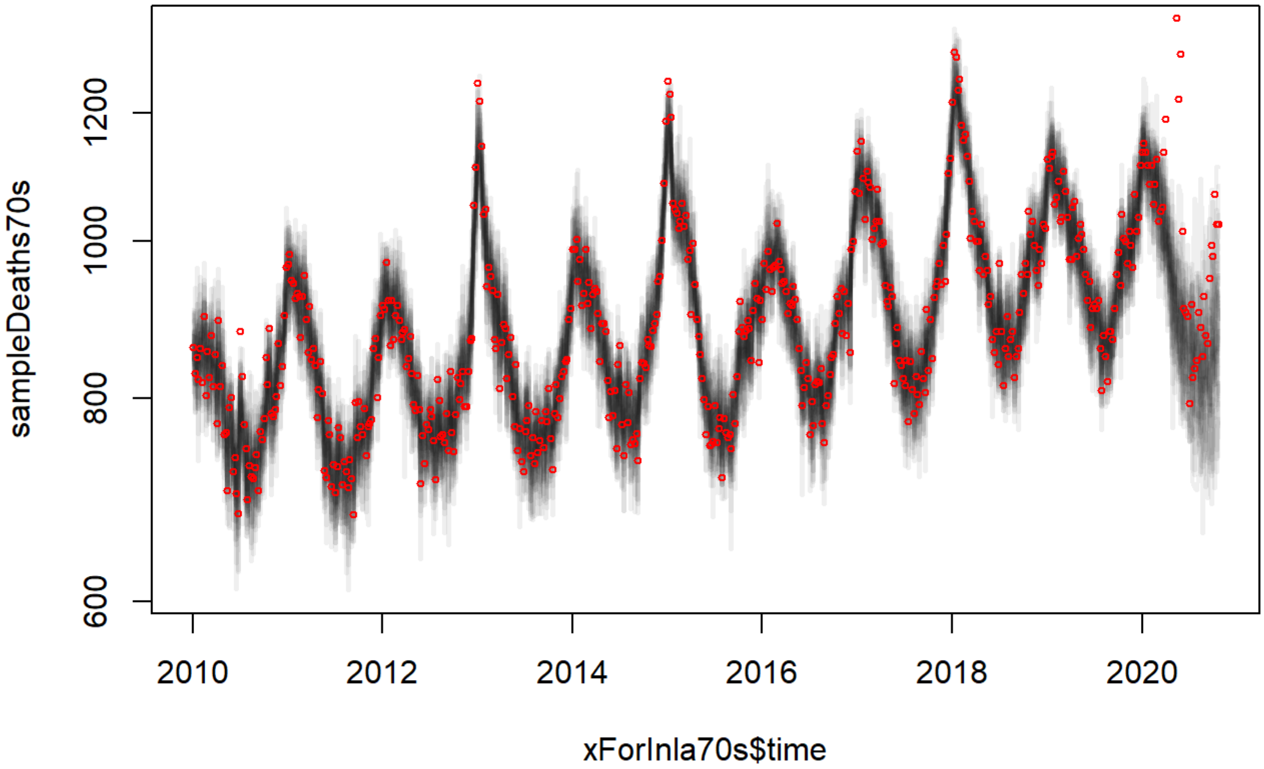
7 rows

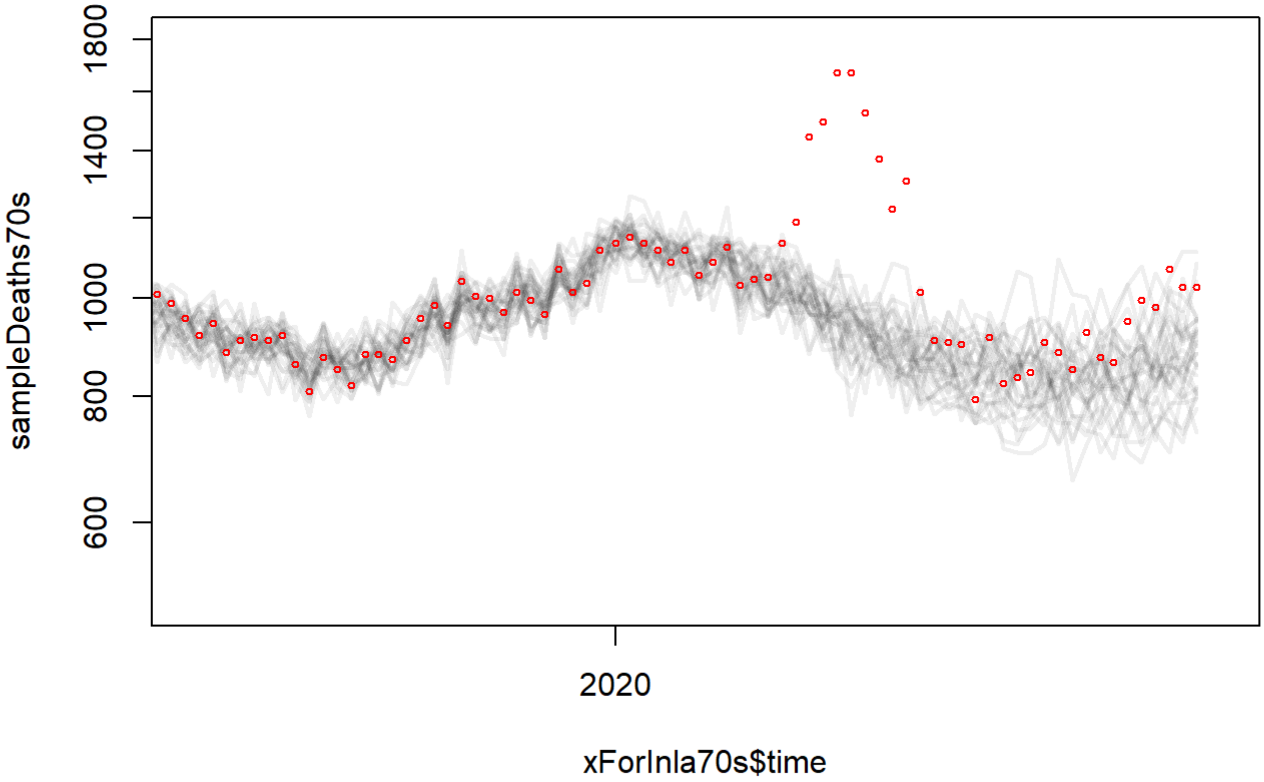
The red points means the actual death number and the black curve is what we estimated.

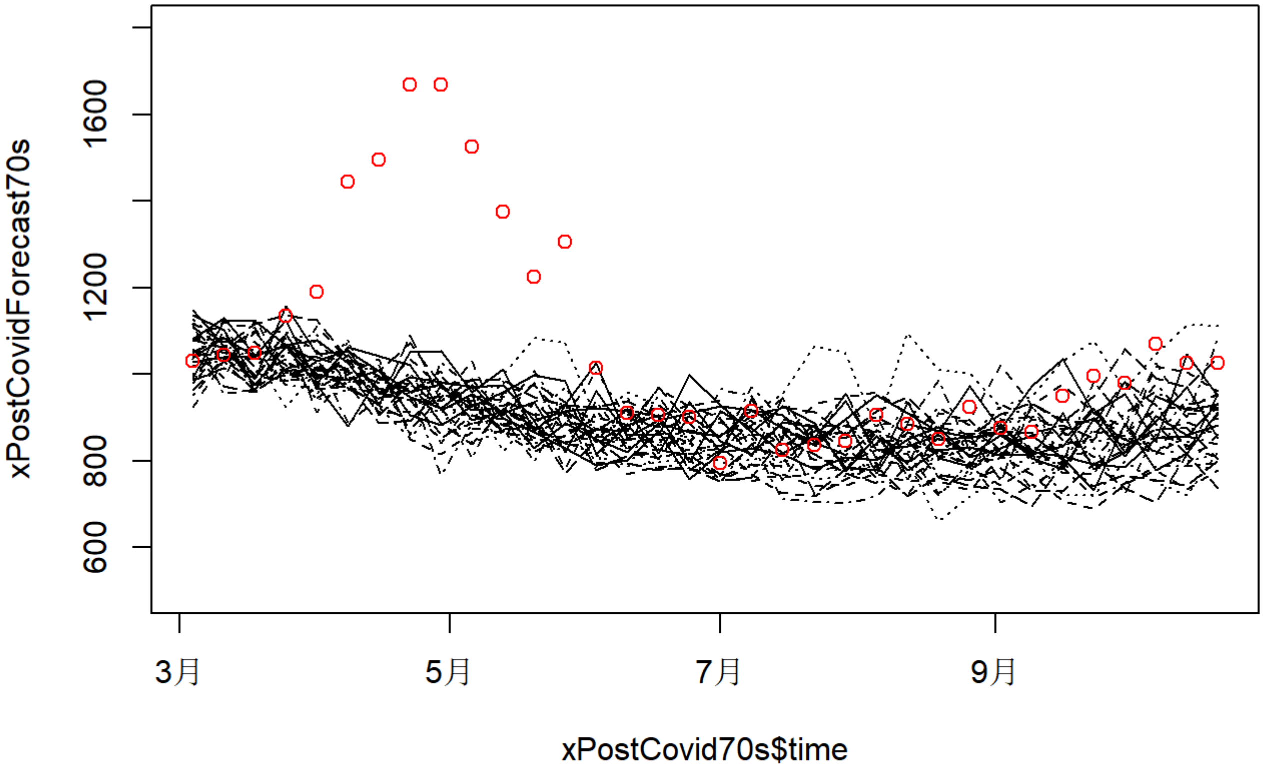




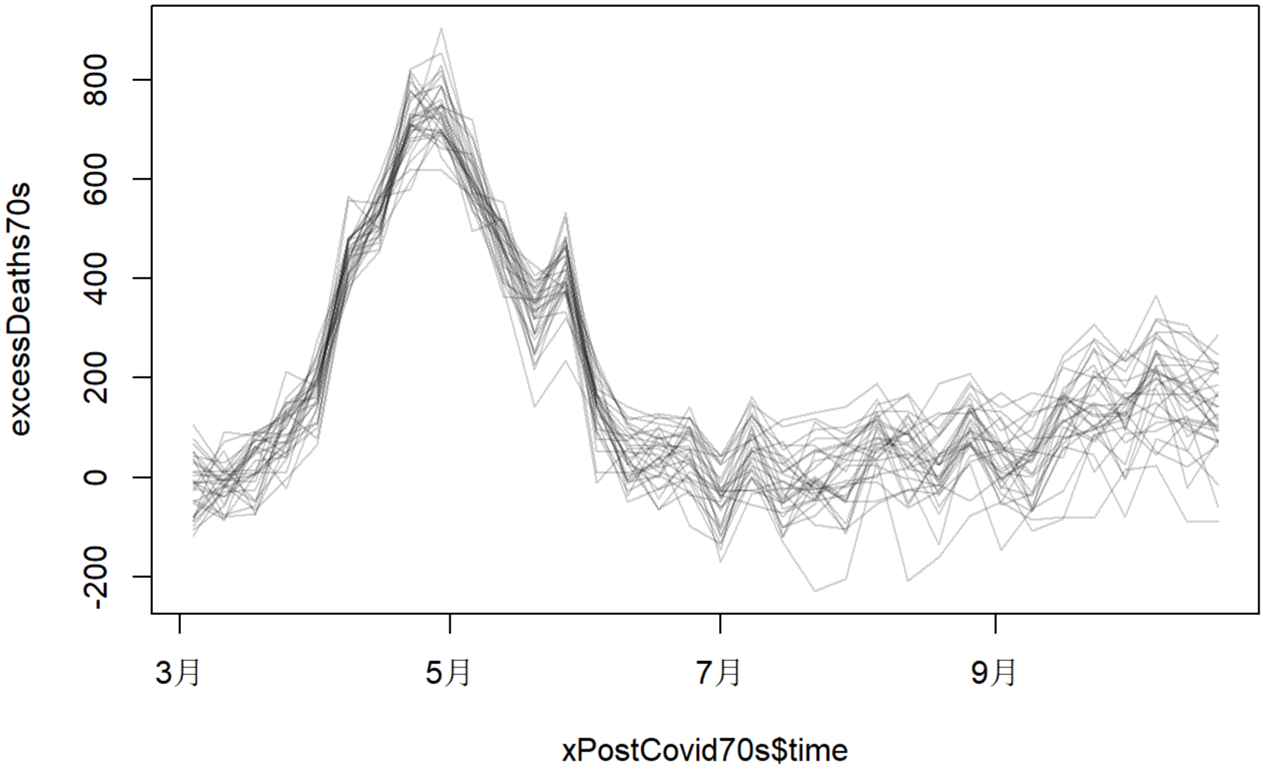
We take 30 random sample from the inla model and make the graph. Here are the comparison of sample and the actual death number.


sampleDeaths70s
600
800
1000
1200
2010
2012
2014
2016
2018
2020
xForInla70s$time


sampleDeaths70s
600
800
1000
1400
1800
2020
xForInla70s$time



From the sample quantile, we can see the estimation make a huge difference starting from March. Such that we make an excess death graph to make sure the difference between the real death number and the number we estimated.

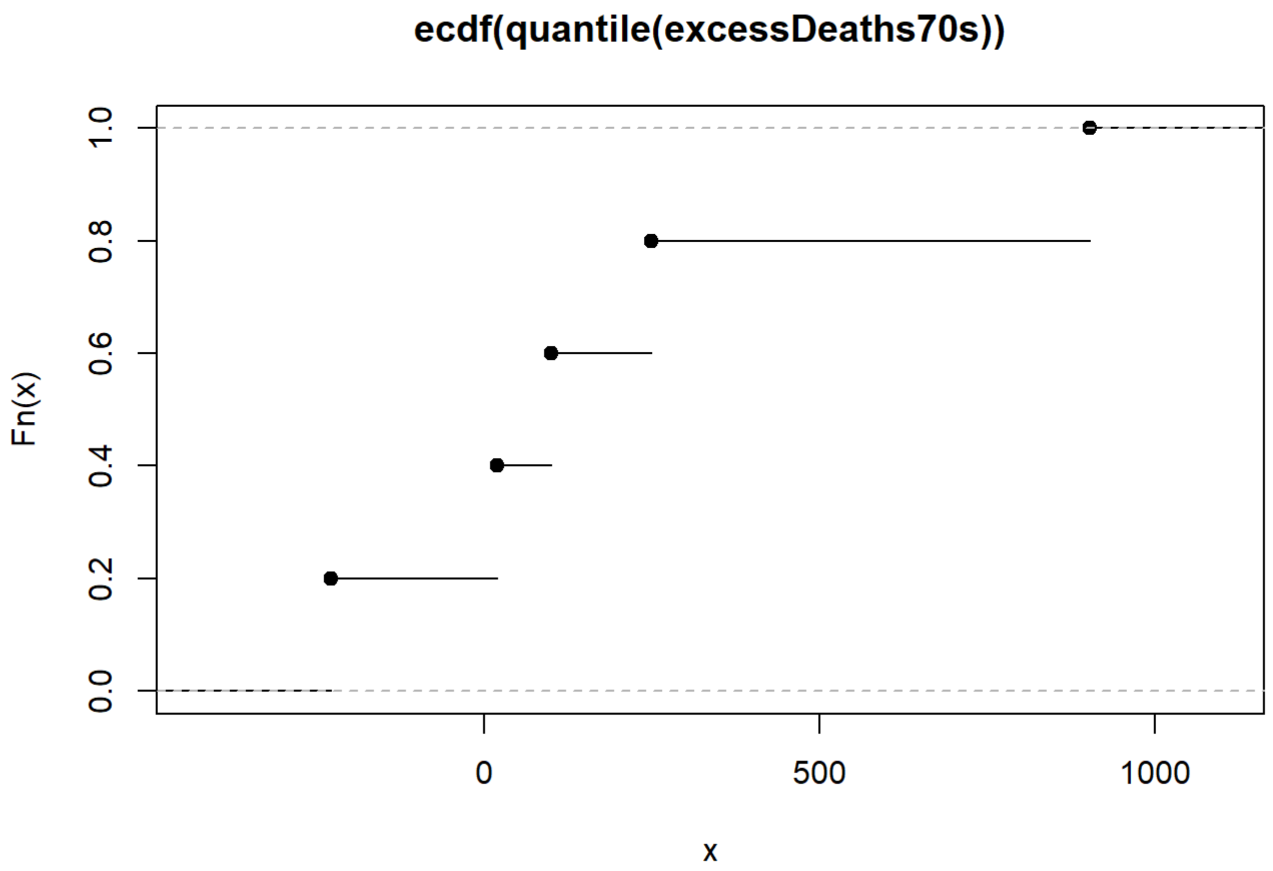


excess death other than estimation

From the graph, we can see the large number of difference exist from March to July on the population of 70s and over. The officer concerns the first wave starting among the elderly. Deaths among the elderly in the spring were well above the historical averages The hypothesis about the first wave is proofed with the data. Below is the empirical quantile distribution that is provided as an other evidence.

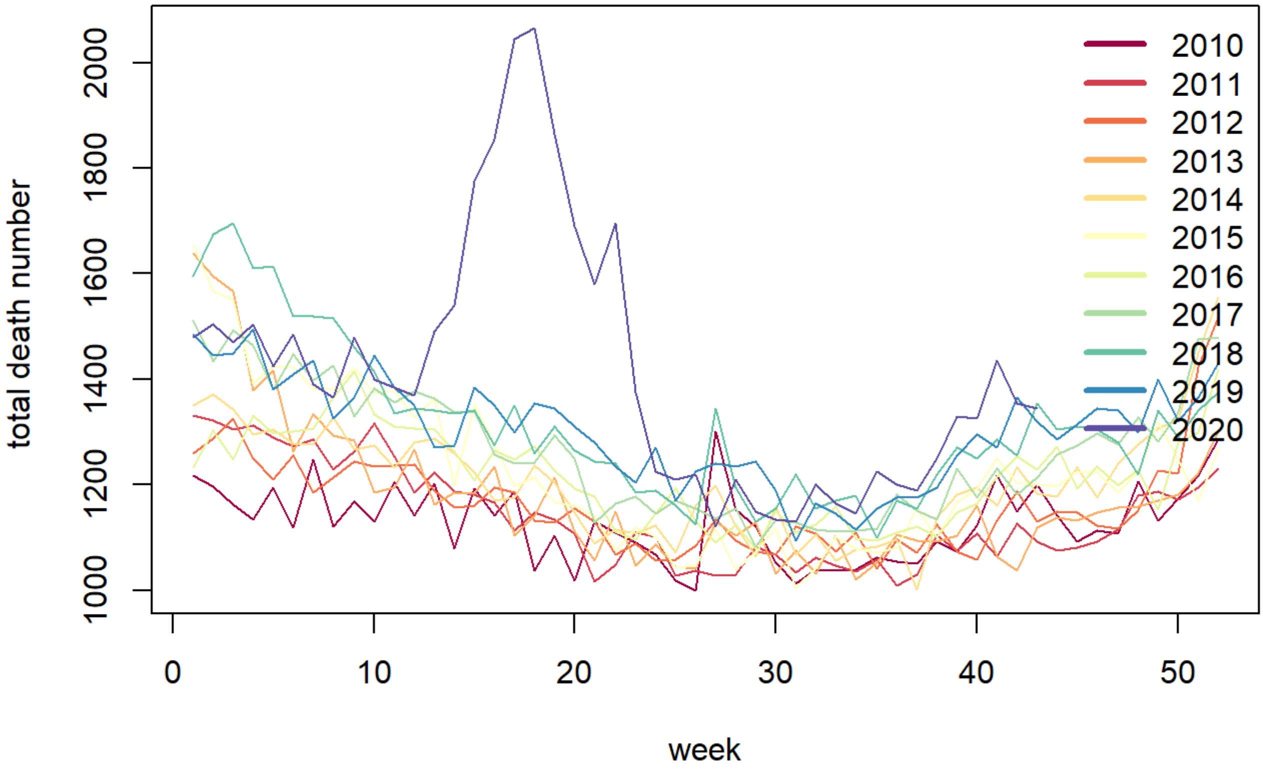
```
##   0%  25%  50%  75% 100% 
## 3702 4212 4540 4760 5087
```

```
##   0%  25%  50%  75% 100% 
##  -87   80  123  201  288
```

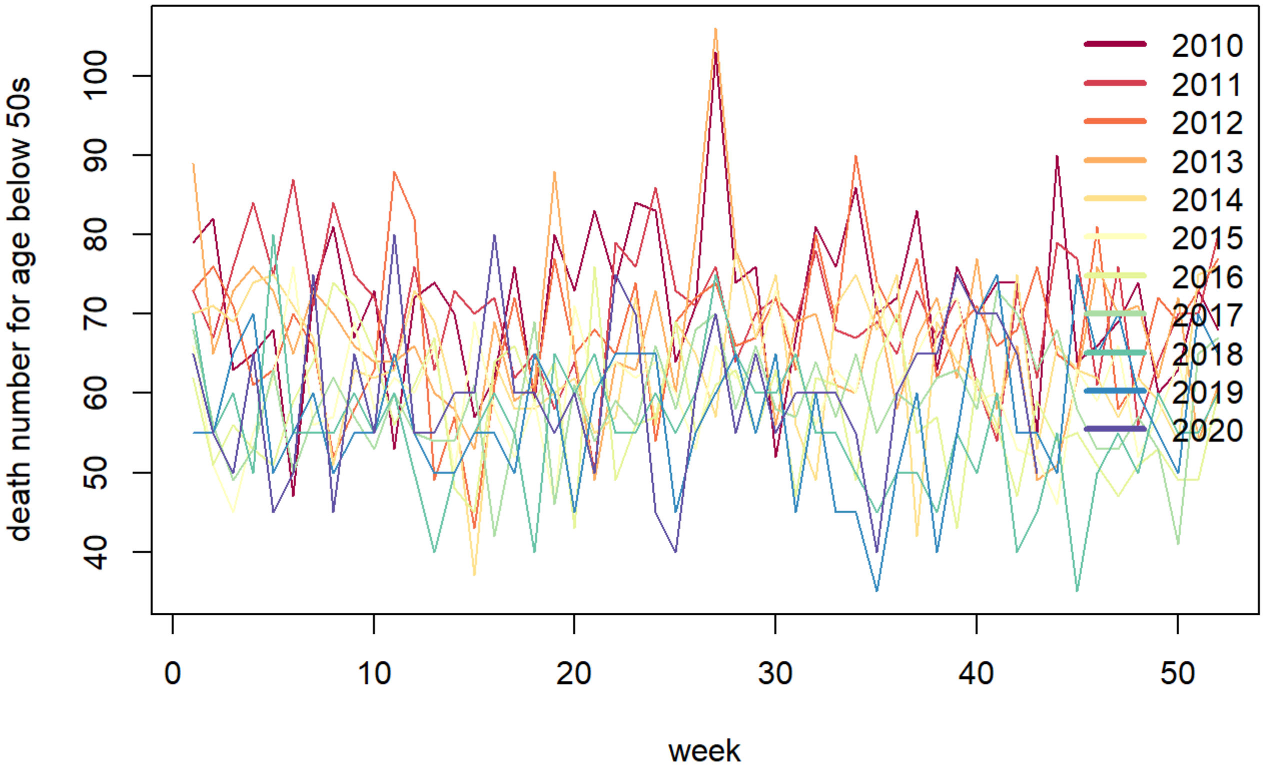


Below 50s

The hypothesis says that the second wave happens mainly around the unresponsive university undergraduates began in September. Since they tends to not concern too much about the COVID-19 virus. We made two graph of the total death number and the death number for the population below 50 years old during weeks in different year. The data appears normal in each year, we then fit a model like for the 70s to make the prediction comparison.



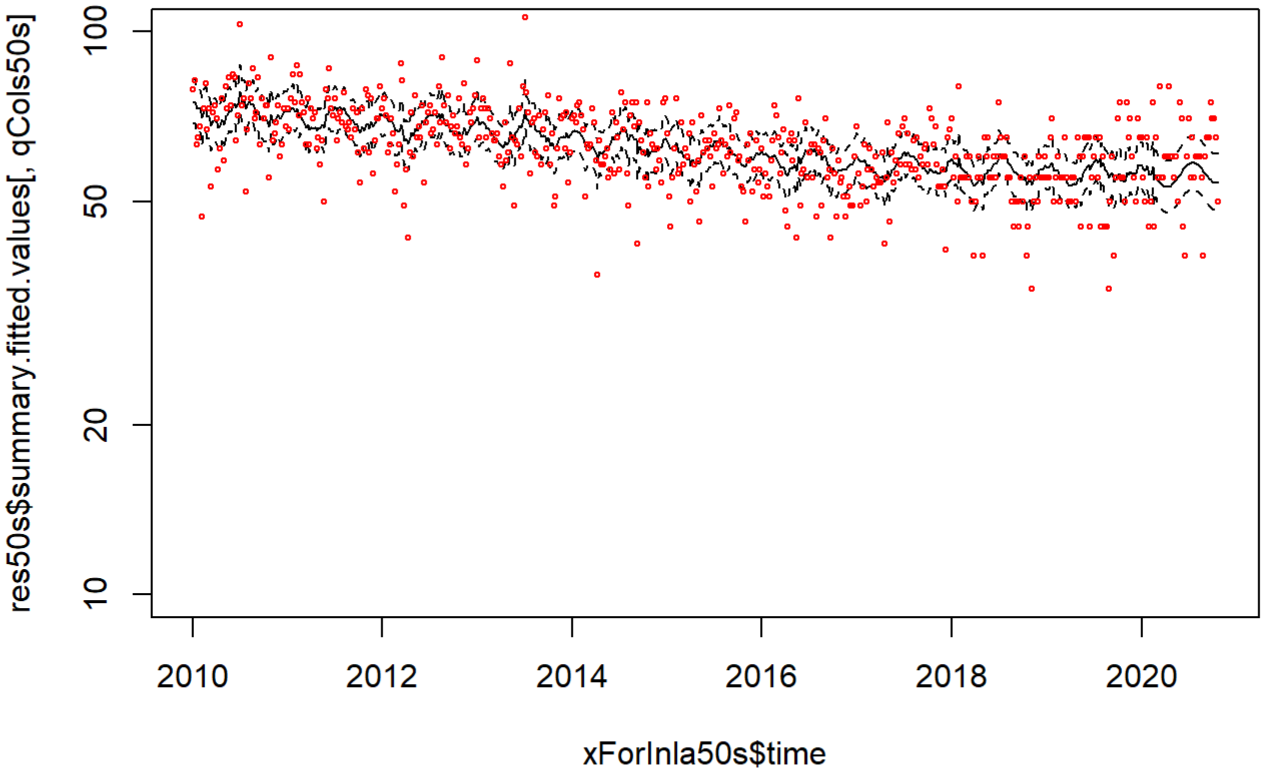
death number in quebec for every week

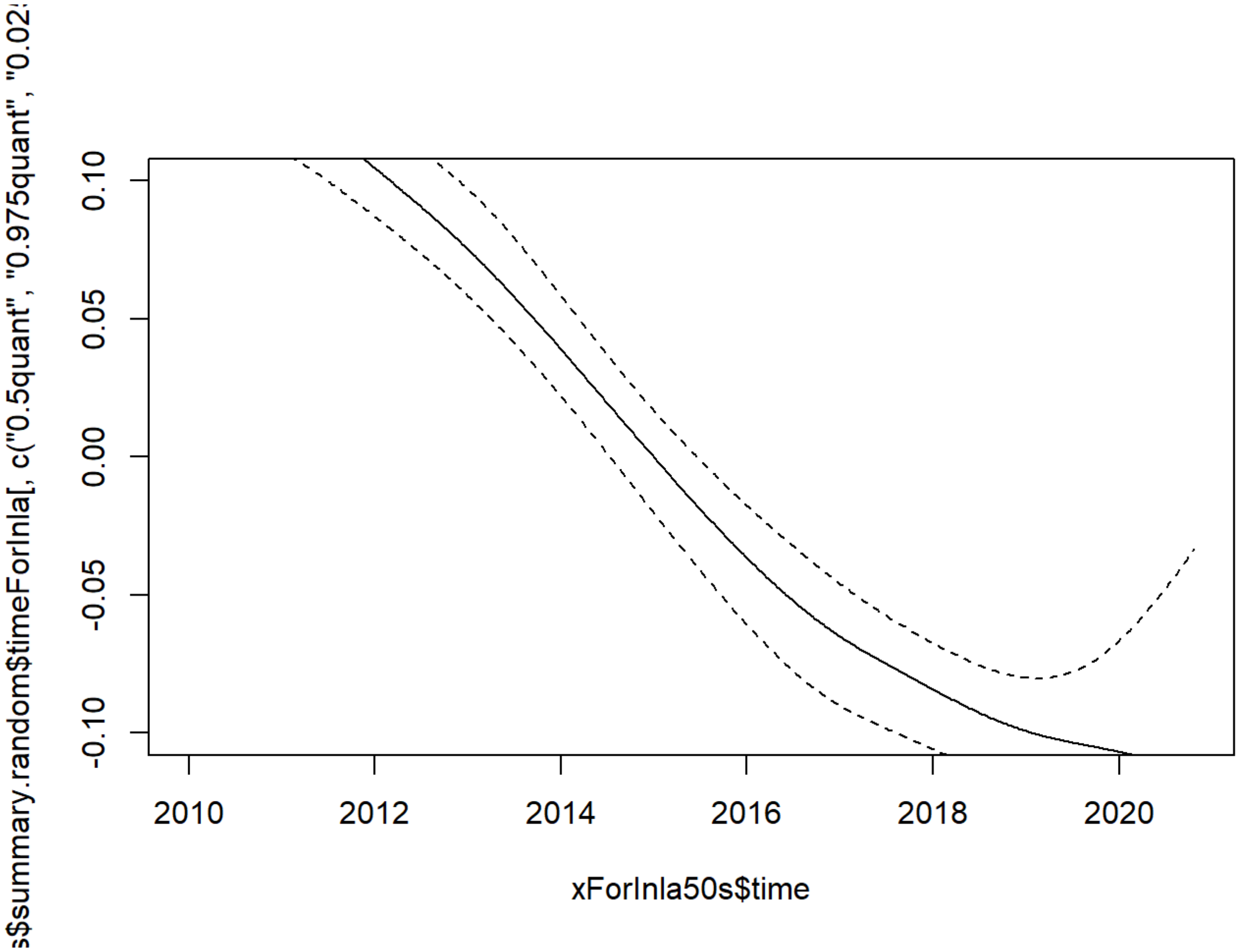


death number in quebec for every week

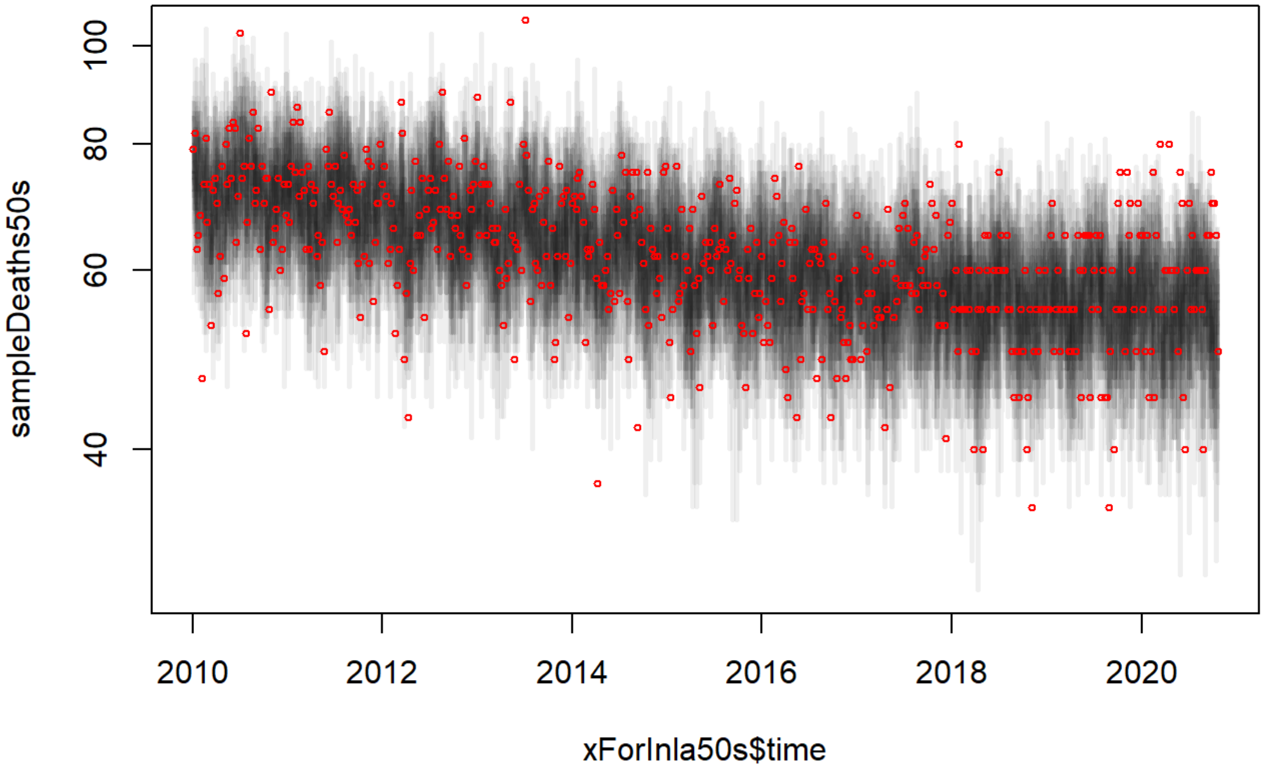
	0.5quant	0.025quant	0.975quant
	<dbl>	<dbl>	<dbl>
(Intercept)	4.12852872	4.116370381	4.140577276
sin12	-0.01141810	-0.027786429	0.004937105
sin6	0.01251369	-0.003589952	0.028602545
cos12	-0.01106991	-0.027073235	0.004929779
cos6	0.03687461	0.020705888	0.053032488
SD for timeIid	0.04311282	0.024018604	0.064336930
SD for timeForInla	0.01248947	0.003217749	0.037437798

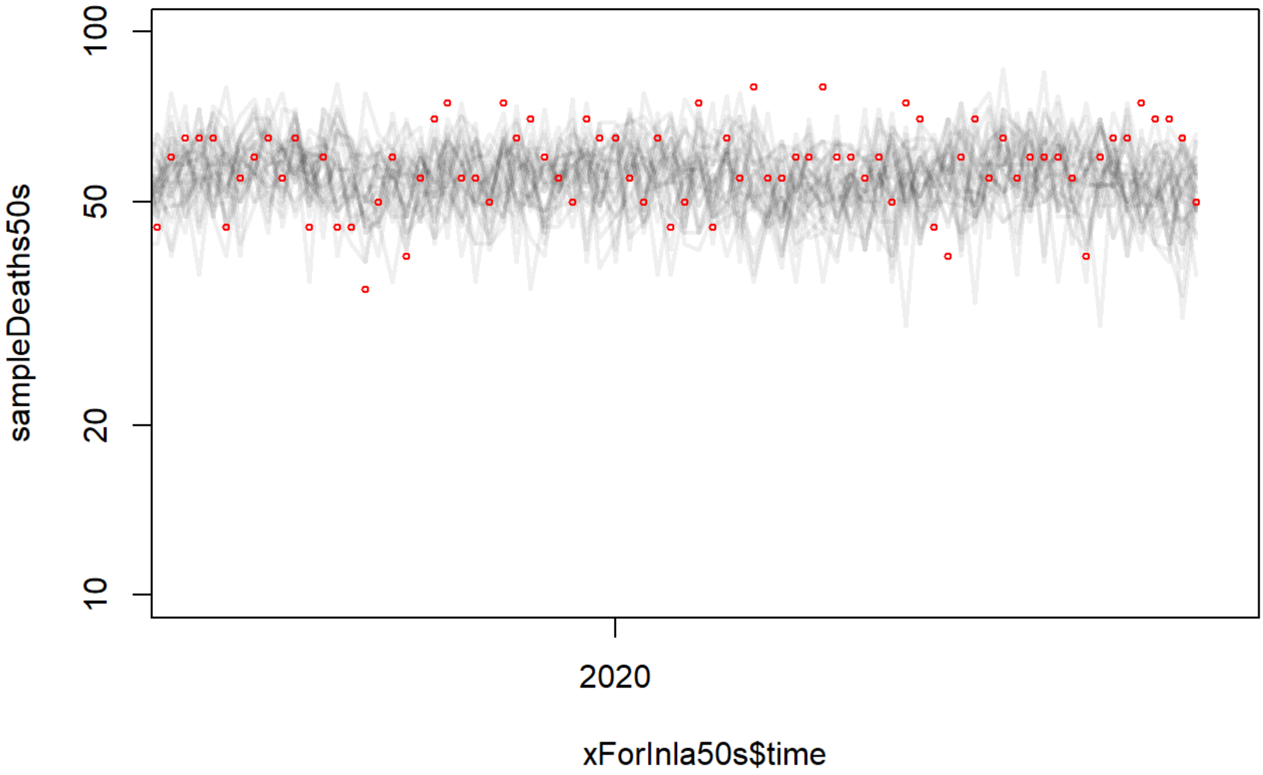
7 rows

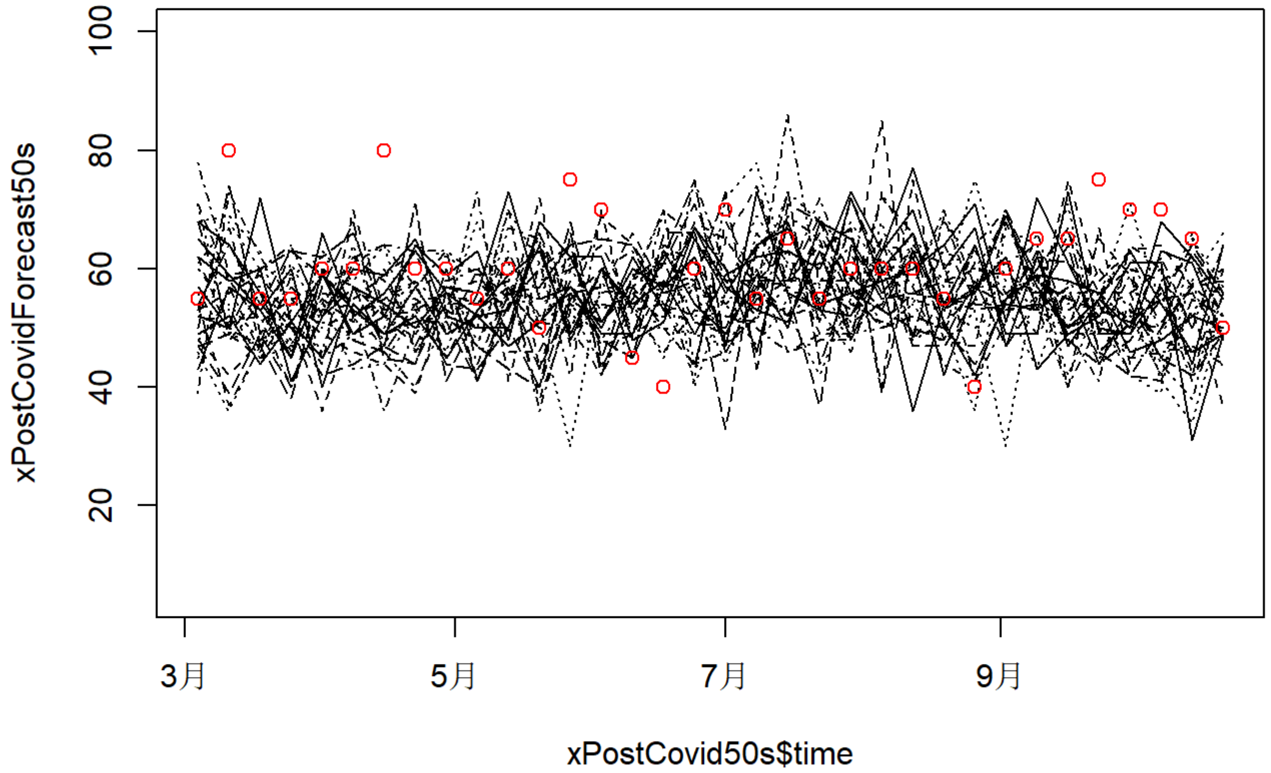




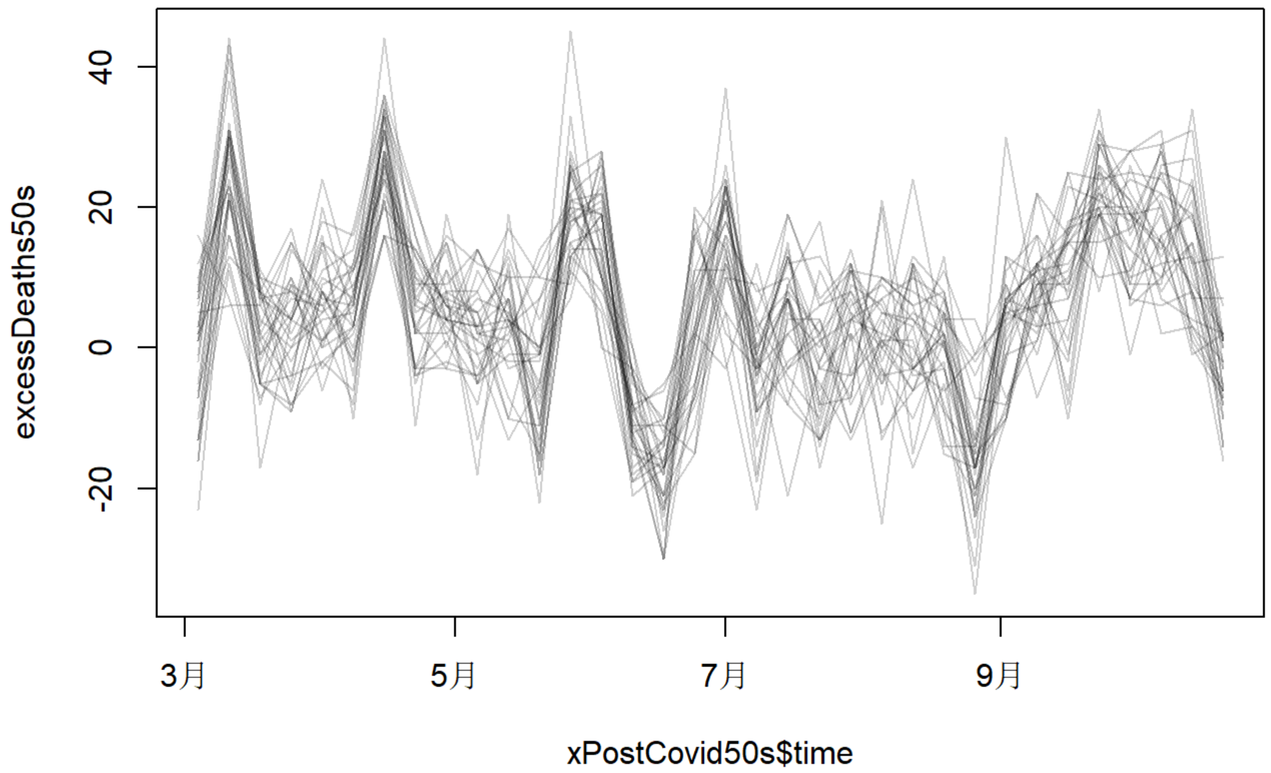
Since the seasonal effect, we use the same model for the young population. But we can see the confidence interval for sin12,sin6,cos12 contains 0, which is not significant, the other variables are significant. The two graph shows the fitted values and the confidence interval of the random effect.


100
80
60
40
sampleDeaths50s
2010
2012
2014
2016
2018
2020
xForInla50s$time


100
50
20
10
sampleDeaths50s
2020
xForInla50s$time

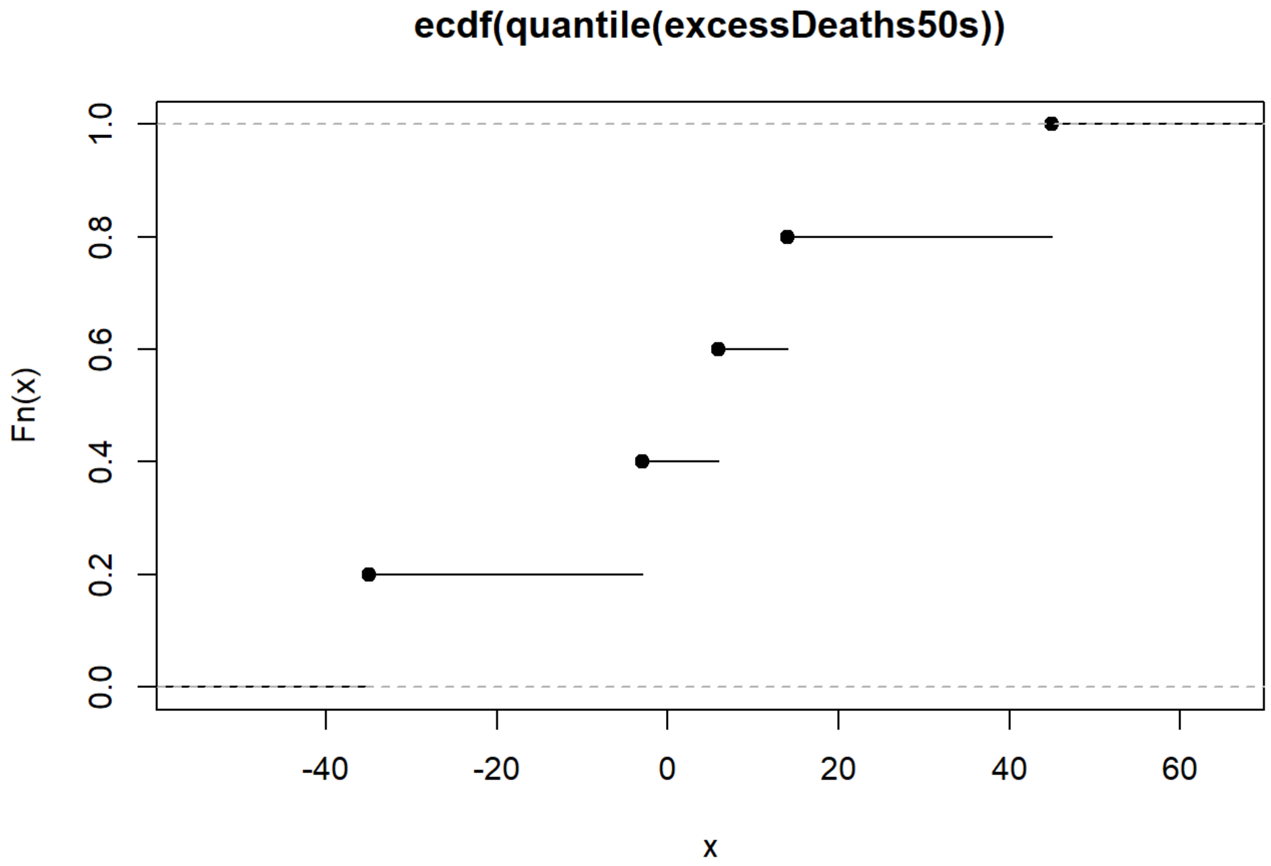


As for the elderly, we take 30 random sample from the estimation for the below 50s compared to the actual data.



```
##   0%  25%  50%  75% 100% 
##   22   67  101  136  177 
```

```
##   0%  25%  50%  75% 100% 
##  -16   -7   -3    1   13 
```



Take a look at the excess death and the quantile graph. We can see there is some unusual death increase since March. But the data shows an regular pattern compare to the elderly. The hypothesis says the second wave is mainly among the unresponsive undergraduates. The data shows a high number of excess death starting from September but not high enough to identify it as the second wave mainly population. Such that we can make an conclusion that there is an increase in death in the under 50’s. But we have not enough evidence to support the hypothesis about the cause of the second wave is primarily young people.

Conclusion

From the daily mortality counts in Quebec, we can see the unusual increase in all ages after the coming of covid-19. The first wave mainly affected the elderly but the second wave is not caused by the young people. We need more analysis to find out the main cause of the second wave. Such that the first hypothesis is true. But we have not enough evidence to support the second hypothesis.

Appendix

```
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$date = strptime(paste(co2s$day, co2s$time),format = "%Y-%m-%d %H:%M", tz = "UTC")
# remove low-quality measurements
co2s = co2s[co2s$quality == 0, ]

#plot(co2s[co2s$date > ISOdate(2015, 3, 1, tz = "UTC"), c("date", "co2")], log = "y", type = "o", xlab = "time", ylab = "ppm", cex = 0.5)
co2s$day = as.Date(co2s$date)
toAdd = data.frame(day = seq(max(co2s$day) + 3, as.Date("2025/1/1"), by = "10 days"), co2 = NA)
co2ext = rbind(co2s[, colnames(toAdd)], toAdd)
timeOrigin = as.Date("2000/1/1")
co2ext$timeInla = round(as.numeric(co2ext$day - timeOrigin)/365.25,2)
co2ext$cos12 = cos(2 * pi * co2ext$timeInla)
co2ext$sin12 = sin(2 * pi * co2ext$timeInla)
co2ext$cos6 = cos(2 * 2 * pi * co2ext$timeInla)
co2ext$sin6 = sin(2 * 2 * pi * co2ext$timeInla)
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$min.diff = NULL
assign("inla.models", mm, INLA:::inla.get.inlaEnv())
co2res = inla(co2 ~ sin12 + cos12 + sin6 + cos6 + f(timeInla, model = 'rw2', prior='pc.prec', param = c(0.1, 0.5)), data = co2ext, family='gamma', control.family = list(hyper=list(prec=list( prior='pc.prec', param=c(0.1, 0.5)))),
# add this line if your computer has trouble
control.inla = list(strategy='gaussian'),
control.predictor = list(compute=TRUE, link=1),
control.compute = list(config=TRUE), verbose=FALSE)
qCols = c('0.5quant','0.025quant','0.975quant')
Pmisc::priorPost(co2res)$summary[,qCols]
# source('https://bioconductor.org/biocLite.R')
# biocLite('Biobase')
sampleList = INLA::inla.posterior.sample(30, co2res, selection = list(timeInla = 0))
sampleMean = do.call(cbind, Biobase::subListExtract(sampleList,"latent"))
sampleDeriv = apply(sampleMean, 2, diff)/diff(co2res$summary.random$timeInla$ID)
plot(co2s$date, co2s$co2, log = "y", cex = 0.3, col = "#00000040", xlab = "time", ylab = "ppm")
matplot(co2ext$day, co2res$summary.fitted.values[, qCols], type = "l", col = "black", lty = c(1, 2, 2), log = "y", xlab = "time", ylab = "ppm")
#pridiction
Stime = timeOrigin + round(365.25 * co2res$summary.random$timeInla$ID)
#matplot(Stime, co2res$summary.random$timeInla[, qCols], type = "l", col = "black", lty = c(1, 2, 2), xlab = "time", ylab = "y")
#random effect
matplot(Stime[-1], sampleDeriv, type = "l", lty = 1, xaxs = "i", col = "#00000020", xlab = "time", ylab = "deriv", ylim = quantile(sampleDeriv, c(0.01, 0.995)))
#derivitive of prior sample
forX = as.Date(c("2018/1/1", "2021/1/1"))
forX = seq(forX[1], forX[2], by = "6 months")
toPlot = which(Stime > min(forX) & Stime < max(forX))
```

```
matplot(Stime[toPlot], sampleDeriv[toPlot, ], type = "l", lty = 1, lwd = 2, xaxs = "i", col =
"#00000050", xlab = "time", ylab = "deriv", xaxt = "n", ylim = quantile(sampleDeriv[toPlot,],
c(0.01, 0.995)))
#focusing to recently
axis(1, as.numeric(forX), format(forX, "%b%Y"))
xWide = read.table(paste0("https://www.stat.gouv.qc.ca/statistiques/",
"population-demographie/deces-mortalite/", "WeeklyDeaths_QC_2010-2020_AgeGr.csv"),
sep = ";", skip = 7, col.names = c("year", "junk","age", paste0("w", 1:53)))
xWide = xWide[grep("^[[:digit:]]+$", xWide$year), ]
x = reshape2::melt(xWide, id.vars = c("year", "age"),measure.vars = grep("^w[[:digit:]]+$", c
olnames(xWide)))
x$dead = as.numeric(gsub("[[:space:]]", "", x$value))
x$week = as.numeric(gsub("w", "", x$variable))
x$year = as.numeric(x$year)
x = x[order(x$year, x$week, x$age), ]
newYearsDay = as.Date(ISOdate(x$year, 1, 1))
x$time = newYearsDay + 7 * (x$week - 1)
x = x[!is.na(x$dead), ]
x = x[x$week < 53, ]
#plot(x[x$age == "Total", c("time", "dead")], type = "o",log = "y")
#plot(x[x$age == "0-49 years old", c("time", "dead")], type = "o",log = "y")
xWide2 = reshape2::dcast(x, week + age ~ year, value.var = "dead")
Syear = grep("[[:digit:]]", colnames(xWide2), value = TRUE)
Scol = RColorBrewer::brewer.pal(length(Syear), "Spectral")
dateCutoff = as.Date("2020/3/1")
xPreCovid = x[x$time < dateCutoff, ]
xPostCovid = x[x$time >= dateCutoff, ]
toForecast = expand.grid(age = unique(x$age), time = unique(xPostCovid$time),dead = NA)
xForInla = rbind(xPreCovid[, colnames(toForecast)],toForecast)
xForInla = xForInla[order(xForInla$time, xForInla$age),]
xForInla$timeNumeric = as.numeric(xForInla$time)
xForInla$timeForInla = (xForInla$timeNumeric - as.numeric(as.Date("2015/1/1")))/365.25
xForInla$timeIid = xForInla$timeNumeric
xForInla$sin12 = sin(2 * pi * xForInla$timeNumeric/365.25)
xForInla$sin6 = sin(2 * pi * xForInla$timeNumeric *2/365.25)
xForInla$cos12 = cos(2 * pi * xForInla$timeNumeric/365.25)
xForInla$cos6 = cos(2 * pi * xForInla$timeNumeric *2/365.25)
xForInlaTotal= xForInla[xForInla$age == 'Total', ]
xForInla50s= xForInla[xForInla$age == '0-49 years old', ]
xForInla70s = xForInla[xForInla$age == '70 years old and over', ]
library(INLA, verbose=FALSE)
plot(x[x$age == "70 years old and over", c("time", "dead")], type = "o",log = "y")
res70s = inla(dead ~ sin12 + sin6 + cos12 + cos6 +
f(timeIid, prior='pc.prec', param= c(log(1.2), 0.5)) +
f(timeForInla, model = 'rw2', prior='pc.prec', param= c(0.01, 0.5)),
data=xForInla70s,
control.predictor = list(compute=TRUE, link=1),
control.compute = list(config=TRUE),
# control.inla = list(fast=FALSE, strategy='laplace'),
family='poisson')
qCols70s = paste0(c(0.5, 0.025, 0.975), "quant")
rbind(res70s$summary.fixed[, qCols70s], Pmisc::priorPostSd(res70s)$summary[,qCols70s])
#the estimated parameter
matplot(xForInla70s$time, res70s$summary.fitted.values[,qCols70s], type = "l", ylim = c(500,
1800), lty = c(1,2, 2), col = "black", log = "y")
points(x[x$age == "70 years old and over", c("time", "dead")], cex = 0.4,col = "red")
matplot(xForInla70s$time, res70s$summary.random$timeForInla[,c("0.5quant", "0.975quant", "0.0
25quant")], type = "l",lty = c(1, 2, 2), col = "black", ylim = c(-1, 1) *0.1)
```

```
sampleList70s = INLA::inla.posterior.sample(30, res70s, selection = list(Predictor = 0))
sampleIntensity70s = exp(do.call(cbind, Biobase::subListExtract(sampleList70s,"latent")))
sampleDeaths70s = matrix(rpois(length(sampleIntensity70s),
sampleIntensity70s), nrow(sampleIntensity70s), ncol(sampleIntensity70s))
matplot(xForInla70s$time, sampleDeaths70s, col = "#00000010",lwd = 2, lty = 1, type = "l", log = "y")
points(x[x$age == "70 years old and over", c("time", "dead")], col = "red",cex = 0.5)
matplot(xForInla70s$time, sampleDeaths70s, col = "#00000010",lwd = 2, lty = 1, type = "l", log = "y", xlim = as.Date(c("2019/6/1","2020/11/1")), ylim = c(500,1800))
points(x[x$age == "70 years old and over", c("time", "dead")], col = "red",cex = 0.5)
xPostCovid70s = xPostCovid[xPostCovid$age == "70 years old and over",]
xPostCovidForecast70s = sampleDeaths70s[match(xPostCovid70s$time,xForInla70s$time), ]
excessDeaths70s = xPostCovid70s$dead - xPostCovidForecast70s
matplot(xPostCovid70s$time, xPostCovidForecast70s, type = "l",ylim = c(500, 1800), col = "black")
points(xPostCovid70s[, c("time", "dead")], col = "red")
matplot(xPostCovid70s$time, excessDeaths70s, type = "l",lty = 1, col = "#00000030")
excessDeathsSub70s = excessDeaths70s[xPostCovid70s$time >
as.Date("2020/03/01") & xPostCovid70s$time <
as.Date("2020/06/01"), ]
excessDeathsInPeriod70s = apply(excessDeathsSub70s, 2, sum)
round(quantile(excessDeathsInPeriod70s))
round(quantile(excessDeaths70s[nrow(excessDeaths70s), ]))
plot(ecdf(quantile(excessDeaths70s)))
matplot(xWide2[xWide2$age == "Total", Syear], type = "l",lty = 1, col = Scol, ylab="total death number ",xlab = "week")
legend("topright", col = Scol, legend = Syear, bty = "n",lty = 1, lwd = 3)
matplot(xWide2[xWide2$age == "0-49 years old", Syear], type = "l",lty = 1, col = Scol, ylab="death number for age below 50s",xlab = "week")
legend("topright", col = Scol, legend = Syear, bty = "n",lty = 1, lwd = 3)
res50s = inla(dead ~ sin12 + sin6 + cos12 + cos6 +
f(timeIid, prior='pc.prec', param= c(log(1.2), 0.5)) +
f(timeForInla, model = 'rw2', prior='pc.prec', param= c(0.01, 0.5)),
data=xForInla50s,
control.predictor = list(compute=TRUE, link=1),
control.compute = list(config=TRUE),
# control.inla = list(fast=FALSE, strategy='laplace'),
family='poisson')
qCols50s = paste0(c(0.5, 0.025, 0.975), "quant")
rbind(res50s$summary.fixed[, qCols50s], Pmisc::priorPostSd(res50s)$summary[,qCols50s])
matplot(xForInla50s$time, res50s$summary.fitted.values[,qCols50s], type = "l", ylim = c(10,100), lty = c(1,2, 2), col = "black", log = "y")
points(x[x$age == "0-49 years old", c("time", "dead")], cex = 0.4,col = "red")
matplot(xForInla50s$time, res50s$summary.random$timeForInla[,c("0.5quant", "0.975quant", "0.025quant")], type = "l",lty = c(1, 2, 2), col = "black", ylim = c(-1, 1) *0.1)
sampleList50s = INLA::inla.posterior.sample(30, res50s, selection = list(Predictor = 0))
sampleIntensity50s = exp(do.call(cbind, Biobase::subListExtract(sampleList50s,"latent")))
sampleDeaths50s = matrix(rpois(length(sampleIntensity50s),
sampleIntensity50s), nrow(sampleIntensity50s), ncol(sampleIntensity50s))
matplot(xForInla50s$time, sampleDeaths50s, col = "#00000010",lwd = 2, lty = 1, type = "l", log = "y")
points(x[x$age == "0-49 years old", c("time", "dead")], col = "red",cex = 0.5)
matplot(xForInla50s$time, sampleDeaths50s, col = "#00000010",lwd = 2, lty = 1, type = "l", log = "y", xlim = as.Date(c("2019/6/1","2020/11/1")), ylim = c(0.01, 0.1) * 1000)
points(x[x$age == "0-49 years old", c("time", "dead")], col = "red",cex = 0.5)
xPostCovid50s = xPostCovid[xPostCovid$age == "0-49 years old",]
xPostCovidForecast50s = sampleDeaths50s[match(xPostCovid50s$time,xForInla50s$time), ]
```

```
excessDeaths50s = xPostCovid50s$dead - xPostCovidForecast50s
matplot(xPostCovid50s$time, xPostCovidForecast50s, type = "l",ylim = c(5, 100), col = "black"
)
points(xPostCovid50s[, c("time", "dead")], col = "red")
#actual vs predicted with no covid
matplot(xPostCovid50s$time, excessDeaths50s, type = "l",lty = 1, col = "#00000030")
excessDeathsSub50s = excessDeaths50s[xPostCovid50s$time >
as.Date("2020/03/01") & xPostCovid50s$time <
as.Date("2020/06/01"), ]
excessDeathsInPeriod50s = apply(excessDeathsSub50s, 2, sum)
round(quantile(excessDeathsInPeriod50s))
round(quantile(excessDeaths50s[nrow(excessDeaths50s), ]))
plot(ecdf(quantile(excessDeaths50s)))
```