STA442 A3

Tuoyue Huang Student number:1003906712

Wednesday 13 November 2019

"Report of CO2 Data"

Introduction

In the report, we are interested in the change of atmoshperic Carbon Dioxide concentrations as a result from several global vital huaman activities related to industry and economics. The data set is collected from an observatory in Haiwaii, made available by the Scripps CO_2 Program at scrippsco2.ucsd.edu.. In the data set, we have the following covariates: collection time, number of flasks used in daily average, quality of measurement and CO_2 concentrations in ppm.

Methods

By looking at the Figure 1, the lines clearly show a regular fluctuation trend in a priod of time. Therefore, we coud use a wiggly line with restrictions to approximate the trend and get a smooth cuvre to predict the concentration level since Jan 1, 1960. A generalized additive model with a log link function from the gamma distribution family is what we need here and we could use the bayesian inference methodology with the INLA algorithm to analysis the results. The model assuption here we have:

$$Y_i \sim Gamma(\mu_i/v, v) \quad log(\mu_i) = X_i\beta + U(t_i) \quad [U_1...U_T]^T \sim RW2(0, \sigma_U^2)$$

Priors: $\sigma_U \sim Exponential(-log(0.5)/(0.005/26)) \quad v \sim Exponential(-log(0.5)/2)$

where Y_i is the CO_2 concentration at time i and $X_i\beta$ here contains 2 groups of fuctions of sin and cos, which are corresponding to yearly and half yearly fluctuations, to approximate the seasonal effect(cycles) on CO_2 concentrations. If we do not include this, then we will get large variance in our model in order to capture these fluctuations and it will be hard to forcast future levels of CO_2. U_t is a second-order random walk which has second derivatives follows normal $(0, \sigma_U^2)$, resulting a random slope. The penalized complexity prior we set on σ_U means the P($\sigma_U > 0.005/26$) = 0.5. The reason is we guess the slope should change by 0.5% through biweekly data over a year. The other prior is set according to the guessing of P(v > 0.2)= 0.5.

Before fitting the model, we we will exclude low quality measurements to get accurate results. To study the changes of CO_2 due to these events, we could use the random effect plot of time randomwalk2 and the derivative plot. In both plots, the dotted lines represents 95% credible interval, while the solid line is the posterior median.

Results

Figure 1(a): Sequence plot of all data

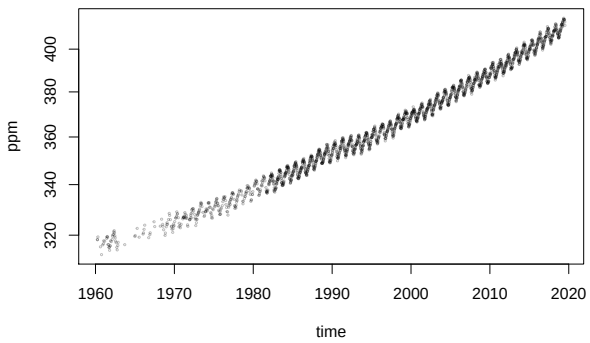


Figure 1(b): Sequence plot of recent data

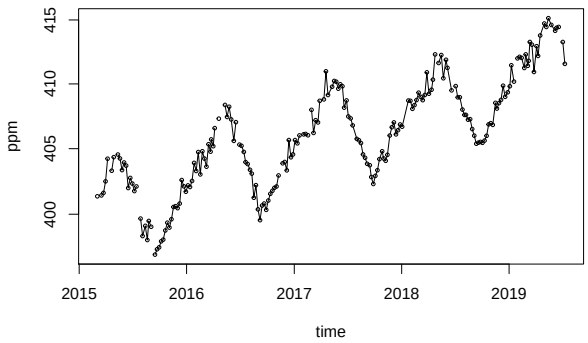


Figure 2: Estimated biweekly CO2 levels in ppm

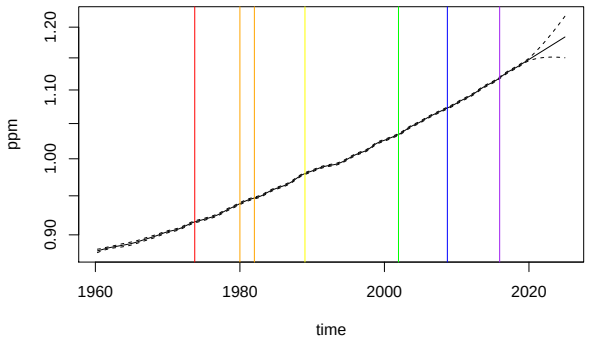
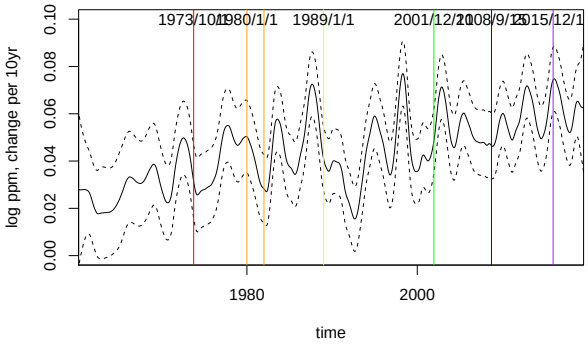


Figure 3: Rate of change of CO2 levels in log ppm



From Figure 2, we could find that the CO_2 kept increasing in the past 80 years, but there are periods it increased at a relatively lower rate or faster rate by checking the slope of the curve around our intersted years. Flat part of the curve indicates a lower increasing rate and versi versa. However, we will mainly focus on Figure 3 since it is easier to tell the rate of change of CO_2 from a derivative plot and we could get the same result as Figure 2. It is the change of rate between today and 10 years ago measured in log scale ppm. We could see that the rate of change is always above 0, meaning CO_2 kept increasing, and a lower rate of change indicates increasing at a relatively lower rate.

• The OPEC oil embargo which began in October 1973 caused rise in oil price and thus people switched from big engine cars to smaller ones, so the rate decreased from 0.032 to 0.028 in the following 2 years.
• The global economic recessions around 1980-1982 caused a lower consumption and production in the whole economy, resulting a drop of rate from 0.048 to 0.029 in this 2-year-period.
• The fall of the Berlin wall almost exactly 30 years ago, preceding a dramatic fall in industrial production in the Soviet Union and Eastern Europe. Therefore, we expect to see a dramatic fall in the rate and indeed it decreased from 0.042 to 0.018 in next 4 years, whcih was the lowest rate among these years.
• China joining the WTO on 11 December 2001, which was followed by rapid growth in industrial production. From the experience of last event, the rate should increase as industrial progress. We could see it rise by 0.014 started from 0.05 after China had joined for one year.
• The bankruptcy of Lehman Brothers on 15 September 2008, regarded as the symbolic start of the most recent global financial crisis, this event should lead to a decrease in rate since economy went into recession. However, the rate actually increased about 0.01 starting from 0.048. This may be one failure prediction of our model or some other events happened which has a significant effect on pushing up the CO_2 level.
• The last event is signing of the Paris Agreement on 12 December 2015, intended to limit CO_2 emissions. We could see the rate decreased from 0.07 to 0.05 in the following 2 years after signing the agreement. In the long term, we can see the slope seems fixed at 0.06 from Figure 2, but the credible interval becomes wider after we run out of data.

In conclusion, our model works pretty well on predicting the effects of huamanity activities to the CO_2 level, except for the event of The bankruptcy of Lehman Brothers.

"Report of IPCC Statement of Heat"

Summary

Through analysis of data set "Sable Island Temperature" by fitting a generalized additive mixed model with bayesian inference, we could confirm the statement from IPCC that human activities lead to increase in global temperature.

Introduction

In this report, we are intersted in the change of global temperature due to human activities. IPCC has stated that human activities are estimated to have caused approximately 1.0°C of global warming above pre-industrial levels, with a likely range of 0.8°C to 1.2°C. Global warming is likely to reach 1.5°C between 2030 and 2052 if it continues to increase at the current rate with high confidence. The pre-industrial level refers to time around 1900, which is about 11.5°C.

We will use the temperature data set from Sable Island to analyze the statement form IPCC. The data set contains our response variable max temperature measured in celsius degree and variables date and month, which can be acess from http://pbrown.ca/teaching/appliedstats/data/sableIsland.rds.

Methods

By looking at the Figure 4, the red dots are winter temperatues and the black ones are summers'. We can see temperatures in winter are much more variable and complex, thus we will focus on the summer part. It clearly shows some cycles in the summer temperatures. Therefore, we coud use a wiggly line with restrictions to approximate the trend and get a smooth cuvre to predict the temperature level since Jan 1, 1891. We would like to include variables week and year as random effect in the model since we have multiple observations at each week and year. In addition some summers are longer than others due to our large scale of data, leading to differences between years. In this case, a generalized additive mixed model from the Student T distribution family is what we need here and we could use the bayesian inference methodology with the INLA algorithm to analysis the results. The reason to use a Student T family here is beacause the heavy tails we got from the data. The model assuption here we have:

$$Y_{ijk} \sim StudentT(\theta_1, \theta_2) \quad \mu_{ijk} = X_i\beta + U(t_i) + V_j + W_k \quad [U_1 ... U_T]^T \sim RW2(0, \sigma_U^2)$$

$$V_j \sim N(0, \sigma_v^2) \quad W_k \sim N(0, \sigma_w^2)$$

Priors: $\sigma_U \sim Exponential(-log(0.4)/(0.1/(52 * 100))) \quad \sigma_v \sim Exponential(-log(0.5)/0.5)$
$\sigma_w \sim Exponential(-log(0.5)/0.5) \quad \theta_1 \sim Exponential(-log(0.5)/3) \quad \theta_2 \sim Exponential(-log(0.5)/10)$

where Y_{ijk} is the temperature level at $time_i$, $week_j$ and $year_k$ and $X_i\beta$ here contains 2 groups of fuctions of sin and cos, which are corresponding to yearly and half yearly fluctuations, to approximate the seasonal effect(cycles) on temperature levels. If we do not include this, then we will get large variance in our model in order to capture these fluctuations and it will be hard to forcast future levels of temperature. U_t is a second-order random walk which has second derivatives follows normal $(0, \sigma_U^2)$, resulting a random slope. This represents a longterm variation while the other week random effect represents the short term variation, because temperature goes up and down in short scales, which is an independent random effect. The penalized complexity(pc) prior we set on σ_U means the P($\sigma_U > 0.1/52$*100) = 0.4. The reason is we guess the slope should change by 10% through weekly data over a hundread years. We have 2 pc prior set on random effect of latent variables week and year according to the guessing of P(σ_v or $\sigma_w > 1$)= 0.5. In addition, we have 2 more priors set on the distribution family. Since we guess that 1 standard deviation of an individual daily observation probably got 1 dregree celsius up or down, we set the first pc prior as P($\theta_1 > 1$) = 0.5. The other dof-prior is also a pc prior which set on the degree freedom of Student T distibution by our guess, that is P(v < 10) = 0.5.

Results

Figure 4: Temperature levels in degree Celsius

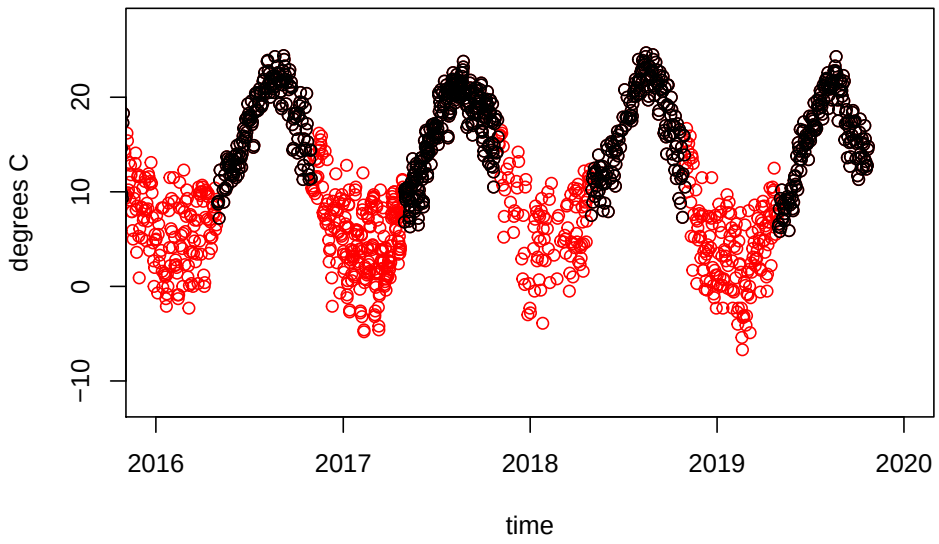


Figure 5: Esitimated temperature levels in degree Celsius

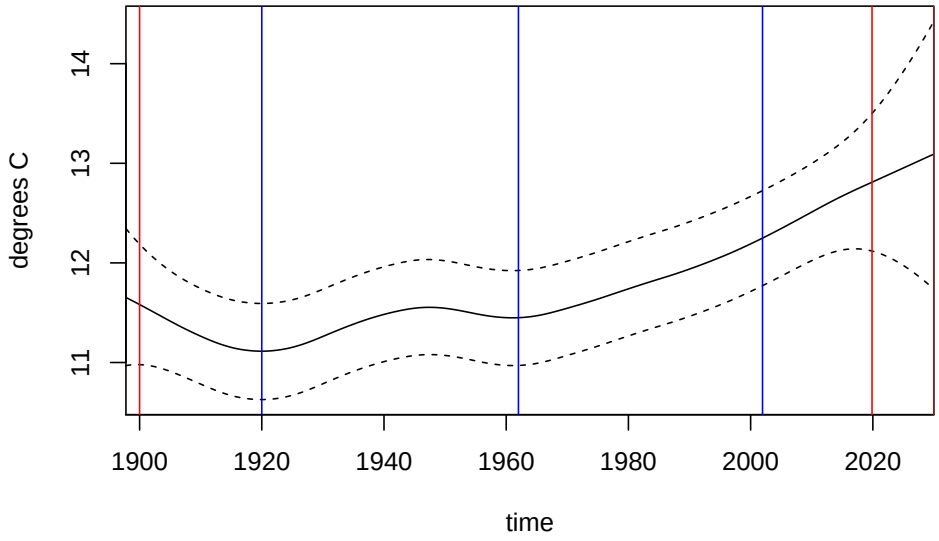


Figure 5 is the posterior distribution plot of the random effect from variable week through randomwalk2. The dotted lines represents 95% credible interval, while the solid line is the posterior median. We can see at year 1900, whcih is our pre-industrial level, the posterior median temperature is about 11.5°C with credible interval from 11°C to 12.1°C. The second red line is the recent time, and we find that the temperature has increased from 11.5°C to about 12.6°C with credible interval from 11.9°C to 13.3°C, which lead to the range of 0.9°C to 1.2°C. In addition, we have year 2030 at the every end, which gives the predition of posterior meadian at 13°C with credible interval from 11.9°C to 14.2°C. These results confirm the IPCC's statement that huaman activitie indeed increased the global warming effect.

The three blue lines indicates the time points where temperature rises at a higher rate. In year 1920, it was the opening of Texas and Persian Gulf oil fields lead to the era of cheap energy. This definitly increased consumption of oil and thus CO_2 emission and temperature increased. In year 1962, it was the peak of the Cold War which may lead to the rapid increase in industrial build for army. Hence, the CO_2 and temperature rised for a further step. The last blue line correspond to the event of China joined WTO, we could see the slope becomes a little bit steeper compare with the first two events.

Table 1: Table 1:Posterior Standard Deviation(SD) for Parameters

	mean	0.025quant	0.975quant
sd for t	1.7679	1.7478	1.7880
sd for week	0.0000	0.0000	0.0001
sd for weekIid	1.0914	1.0533	1.1327
sd for yearFac	0.6882	0.5934	0.7995

This table shows posterior SD for different parameters we used in the model. The fisrt one is SD for parameter θ_1 from Student T ditribution family. By comparing, we can see parameter WeekIid has higher posterior SD than others, which is saying the marjor fluctuation accounts for variation in shorterms caused by difference between weeks.

In conclude, temperature indeed increased for 1°C from the pre-industrial level due to human activities as IPCC states. Howevere, it may be a bit insufficient for us only using the temperature data from Sable Island to infer a global phonomenon. We could further apply the model on temperature data from different places to make a firm conclude.

Appendix

```
co2s = read_csv("CCOO2.csv")
co2s = co2s[,2:9]
co2s = as.data.frame(co2s)

co2s$date = strptime(paste(co2s$day, co2s$time),
                     format='%Y-%m-%d %H:%M', tz='UTC')
# remove low-quality measurements
co2s[co2s$quality>=1, 'co2'] = NA

#Figure 1
plot(co2s$date, co2s$co2, log='y', cex=0.3,
     col='#00000040', xlab='time', ylab='ppm',
     main = "Figure 1(a): Sequence plot of all data")
plot(co2s[co2s$date > ISOdate(2015,3,1, tz='UTC'),
          c('date','co2')], log='y', type='o',
     xlab='time', ylab='ppm', cex=0.5,
     main = "Figure 1(b): Sequence plot of recent data")


timeOrigin = ISOdate(1980,1,1,0,0,0, tz='UTC')
co2s$days = as.numeric(difftime(co2s$date,
                                timeOrigin, units='days'))
co2s$cos12 = cos(2*pi*co2s$days / 365.25)
co2s$sin12 = sin(2*pi*co2s$days / 365.25)
co2s$cos6 = cos(2*2*pi*co2s$days / 365.25)
co2s$sin6 = sin(2*2*pi*co2s$days / 365.25)
```

```r
cLm = lm(co2 ~ days + cos12 + sin12 + cos6 + sin6, data=co2s)
#summary(cLm)$coef[,1:2]

newX = data.frame(
  date=seq(ISOdate(1990,1,1,0,0,0,tz='UTC'),
           by = '1 days', length.out=365*30))
newX$days = as.numeric(difftime(newX$date, timeOrigin, units='days'))
newX$cos12 = cos(2*pi*newX$days / 365.25)
newX$sin12 = sin(2*pi*newX$days / 365.25)
newX$cos6 = cos(2*2*pi*newX$days / 365.25)
newX$sin6 = sin(2*2*pi*newX$days / 365.25)
coPred = predict(cLm, newX, se.fit=TRUE)
coPred = data.frame(est = coPred$fit,
  lower = coPred$fit - 2*coPred$se.fit,
  upper = coPred$fit + 2*coPred$se.fit)
#plot(newX$date,coPred$est, type='l')
#matlines(as.numeric(newX$date), coPred[,c('lower','upper','est')],
         #lty=1, col=c('yellow','yellow','black'))

newX = newX[1:365,]
newX$days = 0
#plot(newX$date, predict(cLm, newX))

library("INLA")
# time random effect
timeBreaks = seq(min(co2s$date), ISOdate(2025, 1, 1,tz = "UTC"), by = "14 days")
timePoints = timeBreaks[-1]
co2s$timeRw2 = as.numeric(cut(co2s$date, timeBreaks))
# derivatives of time random effect
D = Diagonal(length(timePoints)) - bandSparse(length(timePoints),k = -1)
derivLincomb = inla.make.lincombs(timeRw2 = D[-1, ])
names(derivLincomb) = gsub("^lc", "time", names(derivLincomb))
# seasonal effect
StimeSeason = seq(ISOdate(2009, 9, 1, tz = "UTC"),
                  ISOdate(2011, 3, 1, tz = "UTC"),
                  len = 1001)
StimeYear = as.numeric(difftime(StimeSeason, timeOrigin, "days"))/365.35
seasonLincomb = inla.make.lincombs(sin12 = sin(2 *pi * StimeYear),
                                   cos12 = cos(2 * pi * StimeYear),
                                   sin6 = sin(2 * 2 * pi * StimeYear),
                                   cos6 = cos(2 * 2 * pi * StimeYear))
names(seasonLincomb) = gsub("^lc", "season", names(seasonLincomb))
# predictions
StimePred = as.numeric(difftime(timePoints, timeOrigin, units = "days"))/365.35
predLincomb = inla.make.lincombs(timeRw2 = Diagonal(length(timePoints)),
                                 `(Intercept)` = rep(1, length(timePoints)),
                                 sin12 = sin(2 * pi * StimePred),
                                 cos12 = cos(2 * pi * StimePred),
                                 sin6 = sin(2 * 2 * pi * StimePred),
                                 cos6 = cos(2 * 2 * pi * StimePred))
names(predLincomb) = gsub("^lc", "pred", names(predLincomb))
StimeIndex = seq(1, length(timePoints))
timeOriginIndex = which.min(abs(difftime(timePoints, timeOrigin)))
```

```
# disable some error checking in INLA
library("INLA")
mm = get("inla.models", INLA:::inla.get.inlaEnv())
if(class(mm) == 'function') mm = mm()
mm$latent$rw2$min.diff = NULL
assign("inla.models", mm, INLA:::inla.get.inlaEnv())
### Model
co2res = inla(co2 ~ sin12 + cos12 + sin6 + cos6 +
                f(timeRw2, model = 'rw2',values = StimeIndex,
                  prior='pc.prec', param = c(0.005/26, 0.5)),
              data = co2s, family='gamma',
              lincomb = c(derivLincomb, seasonLincomb, predLincomb),
              control.family = list(hyper=list(prec=list(prior='pc.prec', param=c(0.2, 0.5)))),
              # add this line if your computer has trouble
              #  control.inla = list(strategy='gaussian', int.strategy='eb'),
              verbose=TRUE)
### PLOTS
matplot(timePoints,
        exp(co2res$summary.random$timeRw2[, c("0.5quant", "0.025quant", "0.975quant")]),
        type = "l", col = "black", lty = c(1, 2, 2), log = "y",
        xaxt = "n", xlab = "time", ylab = "ppm")
xax = pretty(timePoints)
axis(1, xax, format(xax, "%Y"))
abline(v = ISOdate(1973, 10, 1, tz = "UTC"), col = "red")
text(ISOdate(1980, 1, 1, tz = "UTC"),"1973.10.1")
abline(v = ISOdate(1980, 1, 1, tz = "UTC"), col = "orange")
abline(v = ISOdate(1982, 1, 1, tz = "UTC"), col = "orange")
abline(v = ISOdate(1989, 1, 1, tz = "UTC"), col = "yellow")
abline(v = ISOdate(2001, 12, 11, tz = "UTC"), col = "green")
abline(v = ISOdate(2008, 9, 15, tz = "UTC"), col = "blue")
abline(v = ISOdate(2015, 12, 12, tz = "UTC"), col = "purple")

derivPred = co2res$summary.lincomb.derived[
  grep("time",rownames(co2res$summary.lincomb.derived)),
  c("0.5quant", "0.025quant", "0.975quant")]
scaleTo10Years = (10 * 365.25/as.numeric(diff(timePoints,units = "days")))
matplot(timePoints[-1], scaleTo10Years * derivPred,type = "l", col = "black",
        lty = c(1, 2, 2), ylim = c(0, 0.1), xlim = range(as.numeric(co2s$date)),
        xaxs = "i", xaxt = "n", xlab = "time", ylab = "log ppm, change per 10yr")
axis(1, xax, format(xax, "%Y"))
abline(v = ISOdate(1973, 10, 1, tz = "UTC"), col = "red")
text(ISOdate(1973, 10, 1, tz = "UTC"),0.1,"1973/10/1", cex = 1)
abline(v = ISOdate(1980, 1, 1, tz = "UTC"), col = "orange")
text(ISOdate(1980, 1, 1, tz = "UTC"),0.1,"1980/1/1", cex = 1)
abline(v = ISOdate(1982, 1, 1, tz = "UTC"), col = "orange")

abline(v = ISOdate(1989, 1, 1, tz = "UTC"), col = "yellow")
text(ISOdate(1989, 1, 1, tz = "UTC"),0.1,"1989/1/1", cex = 1)
abline(v = ISOdate(2001, 12, 11, tz = "UTC"), col = "green")
text(ISOdate(2001, 12, 11, tz = "UTC"),0.1,"2001/12/11", cex = 1)
abline(v = ISOdate(2008, 9, 15, tz = "UTC"), col = "blue")
text(ISOdate(2008, 9, 15, tz = "UTC"),0.1,"2008/9/15", cex = 1)
abline(v = ISOdate(2015, 12, 12, tz = "UTC"), col = "purple")
```

```
text(ISOdate(2015, 12, 12, tz = "UTC"),0.1,"2015/12/12", cex = 1)

heatUrl = "http://pbrown.ca/teaching/appliedstats/data/sableIsland.rds"
heatFile = tempfile(basename(heatUrl))
download.file(heatUrl, heatFile)
x = readRDS(heatFile)
x$month = as.numeric(format(x$Date, "%m"))
xSub = x[x$month %in% 5:10 & !is.na(x$Max.Temp...C.),]
weekValues = seq(min(xSub$Date),
                 ISOdate(2030, 1, 1,0, 0, 0, tz = "UTC"), by = "7 days")
xSub$week = cut(xSub$Date, weekValues)
xSub$weekIid = xSub$week
xSub$day = as.numeric(difftime(xSub$Date,
                               min(weekValues), units = "days"))
xSub$cos12 = cos(xSub$day * 2 * pi/365.25)
xSub$sin12 = sin(xSub$day * 2 * pi/365.25)
xSub$cos6 = cos(xSub$day * 2 * 2 * pi/365.25)
xSub$sin6 = sin(xSub$day * 2 * 2 * pi/365.25)
xSub$yearFac = factor(format(xSub$Date, "%Y"))


lmStart = lm(Max.Temp...C. ~ sin12 + cos12 + sin6 + cos6, data = xSub)
startingValues = c(lmStart$fitted.values, rep(lmStart$coef[1], nlevels(xSub$week)),
                   rep(0, nlevels(xSub$weekIid) + nlevels(xSub$yearFac)), lmStart$coef[-1])
#INLA::inla.doc('^t$')
library("INLA")
mm = get("inla.models", INLA:::inla.get.inlaEnv())
if(class(mm) == 'function') mm = mm()
mm$latent$rw2$min.diff = NULL
assign("inla.models", mm, INLA:::inla.get.inlaEnv())

sableRes = INLA::inla(Max.Temp...C. ~ 0 + sin12 + cos12 + sin6 + cos6 +
                        f(week, model='rw2',constr=FALSE,prior='pc.prec',
                          param = c(0.1/(52*100), 0.4)) +
                        f(weekIid, model='iid', prior='pc.prec', param = c(0.5, 0.5)) +
                        f(yearFac, model='iid', prior='pc.prec', param = c(0.5, 0.5)),family='T',
                      control.family = list(hyper = list(prec = list(prior='pc.prec',param=c(3, 0.5)),
                                                         dof = list(prior='pc.dof', param=c(10, 0.5)))),
                      control.mode = list(theta = c(-1,2,20,0,1), x = startingValues, restart=TRUE),
                      control.compute=list(config = TRUE),
                      # control.inla = list(strategy='gaussian', int.strategy='eb'),
                      data = xSub, verbose=TRUE)

#mySample = inla.posterior.sample(
  #n = 24, result = sableRes,num.threads = 8,
  #selection = list(week = seq(1,nrow(sableRes$summary.random$week))))
#length(mySample)
#names(mySample[[1]])
#weekSample = do.call(cbind, lapply(mySample, function(xx) xx$latent))
#dim(weekSample)
#head(weekSample)

#plot(x$Date, x$Max.Temp...C., col = mapmisc::col2html("black", 0.3))
```

```
forAxis = ISOdate(2016:2020, 1, 1, tz = "UTC")

plot(x$Date, x$Max.Temp...C., xlim = range(forAxis),
     xlab = "time", ylab = "degrees C", col = "red", xaxt = "n")
points(xSub$Date, xSub$Max.Temp...C.)
axis(1, forAxis, format(forAxis, "%Y"))

matplot(weekValues[-1],
        sableRes$summary.random$week[,paste0(c(0.5, 0.025, 0.975), "quant")],
        type = "l", lty = c(1, 2, 2), xlab = "time", ylab = "degrees C",
        xaxt = "n", col = "black", xaxs = "i")
forXaxis2 = ISOdate(seq(1880, 2040, by = 20), 1, 1, tz = "UTC")
axis(1, forXaxis2, format(forXaxis2, "%Y"))
abline(v = ISOdate(1900, 1, 1, tz = "UTC"), col = "red")
abline(v = ISOdate(2019, 11, 12, tz = "UTC"), col = "red")
abline(v = ISOdate(2030, 1, 1, tz = "UTC"), col = "red")
abline(v = ISOdate(1920, 1, 1, tz = "UTC"), col = "blue")
abline(v = ISOdate(1962, 1, 1, tz = "UTC"), col = "blue")
abline(v = ISOdate(2001, 12, 11, tz = "UTC"), col = "blue")

sableRes$priorPost = Pmisc::priorPost(sableRes)
knitr::kable(sableRes$priorPost$summary[,c("mean","0.025quant","0.975quant")],digits = 4)
```