

CO2 Concentrations: Impacts of Global Events on Atmospheric Carbon Dioxide Levels

Vinky Wang

22/12/2020

Introduction

The rise in atmospheric carbon dioxide concentrations have been accelerating since the use of fossil fuels and other human activities. The 2015 Paris Agreement aims to limit global temperature rise to 1.5 degrees Celsius above the pre-industrial levels corresponding to 430ppm CO2 to significantly reduce the impacts of climate change.¹ As major global events may impact CO2 emissions, this report aims to investigate atmospheric CO2 concentrations in response to the following historical events:

- the first oil shock in 1973-1974, oil embargo and price rise
- the second oil shock in 1979
- the global economic recession in 1980-82, worldwide decline of economic activity and increase in unemployment
- the fall of the Berlin wall in 1989, preceding a dramatic fall in industrial production in the Soviet Union and Eastern Europe
- the global financial crisis of 2007-08 culminating in the collapse of Lehman Brothers and the Wall Street bailout with lasting impacts on the job market and home prices
- the signing of the Paris Agreement in 2015 to limit CO2 emissions
- the global lockdown during the COVID-19 pandemic starting in February 2020, shutting down much of the global economy

Method

The data is available through the Scripps CO2 program which records hourly global atmospheric carbon dioxide concentration beginning on January 1, 1980 to present. The data contains 2179 observations once removing the low quality measurements.

A seasonally adjusted gamma regression model was fit to the positive, continuous response of atmospheric carbon dioxide concentrations. The seasonal adjustment captures the natural fluctuations in atmospheric CO2 concentrations with higher levels during the winter months corresponding to the decay of plant material and lower levels during the spring months corresponding to the increase in photosynthesis activity in the Northern Hemisphere.² An exploratory analysis of the atmospheric CO2 concentrations over the years reveals an annual cyclical pattern which can be captured by sinusoidal basis functions of $\sin(2\pi x_i)$ and $\cos(2\pi x_i)$ corresponding to the annual fluctuations and $\sin(4\pi x_i)$ and $\cos(4\pi x_i)$ to the bi-annual. The model was implemented in the INLA, or Integrated Nested Laplace Approximation software in R which is a computationally efficient method for implementing Bayesian inferences.³

Model

$$Y_i \sim \text{Gamma}(\nu, \frac{\mu_i}{\nu})$$

$$\log(\mu_i) = X_i^T \beta + f_i(t_i)$$

$$f_i(t+1) - 2f(t) + f(t-1) \sim N(0, \sigma_u^2)$$

$$\beta \stackrel{iid}{\sim} MVN(0, \kappa I)$$

$$\frac{1}{\sqrt{\nu}} \sim \text{Exponential prior such that } Pr(\frac{1}{\sqrt{\nu}} > 0.1) = 0.5$$

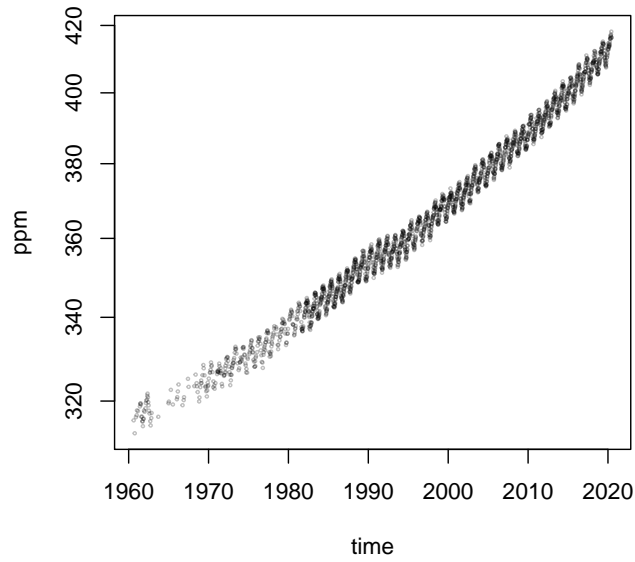
$$\sigma^2 \sim \text{Exponential prior such that } Pr(\sigma > 0.1) = 0.5$$

- Y_i : atmospheric carbon dioxide concentrations (ppm)
- μ_i : mean of atmospheric carbon dioxide concentrations (ppm)
- $\frac{1}{\sqrt{\nu}}$: coefficient of variation
- $f_i(t+1) - 2f(t) + f(t-1)$: seasonally adjusted trend following a second order Random walk

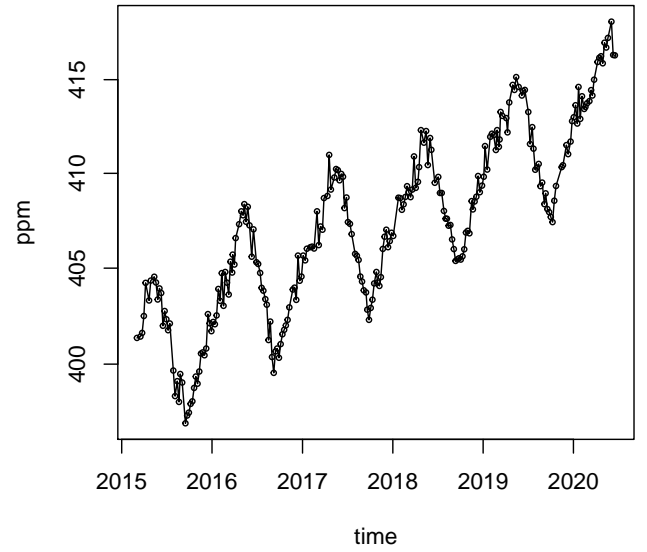
Prior Specification

A default weakly informative zero-mean Gaussian prior was assigned to the regression parameters β . By the favoured model principle of penalized least squares with a second derivative penalty, a second order random walk was assigned to the seasonally adjusted trend to encourage straight lines for achieving lower bias.^[4] A penalized complexity prior was used on the second order random walk to control the strength of the constraint to the favoured model.^[5] The prior of $Pr(\frac{1}{\sqrt{\nu}} > 0.1) = 0.5$ encodes our belief that the changes in mean concentrations of CO2 should be no greater than 10% as the prior median. The prior of $Pr(\sigma > 0.1) = 0.5$ encodes our belief that the annual change in slope should be no greater than 0.1 as the prior median. As the posterior samples generated by the model follow closely with the observed data, this suggests that our prior specifications are on the appropriate order of magnitude.

Results

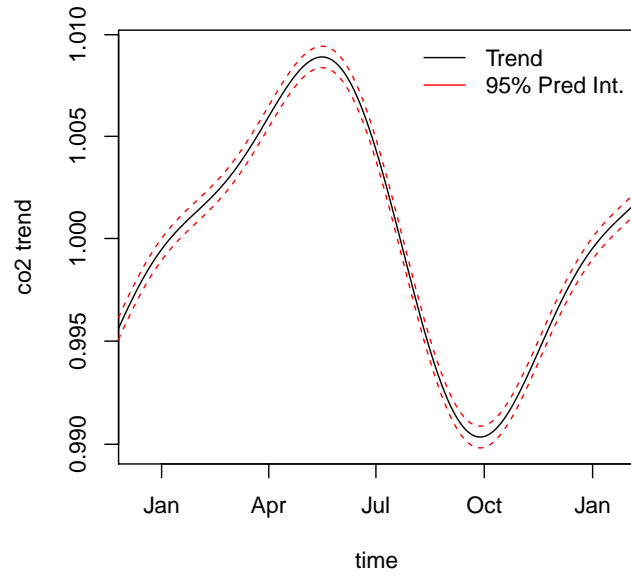


(a) Entire Period

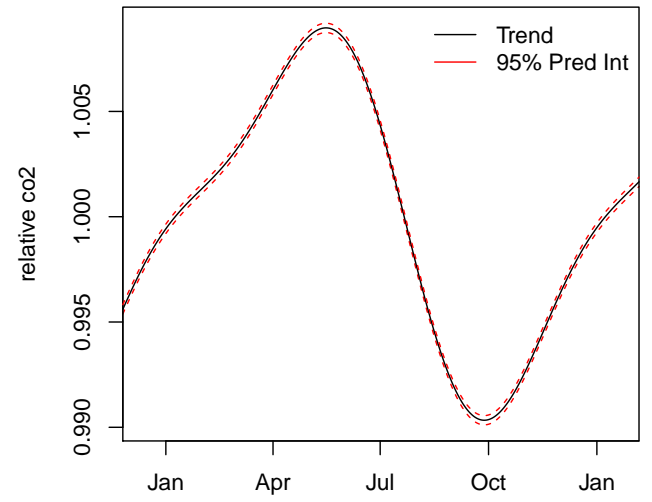


(b) Recent Years

Figure 1: Carbon Dioxide Concentration Observations at Mauna Loa Observatory

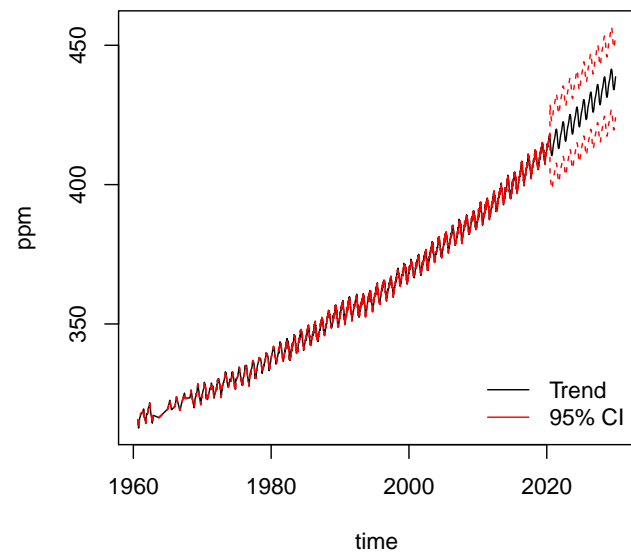


(a) CO2 concentration

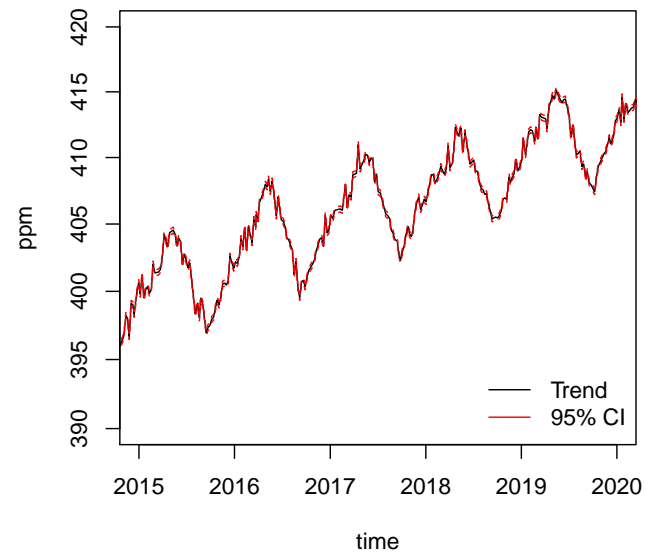


(b) Seasonal Trend

Figure 2: Carbon Dioxide Concentration Predictions.



(a) Overall



(b) Recent Years

Figure 3: Predicted Carbon Dioxide Concentration

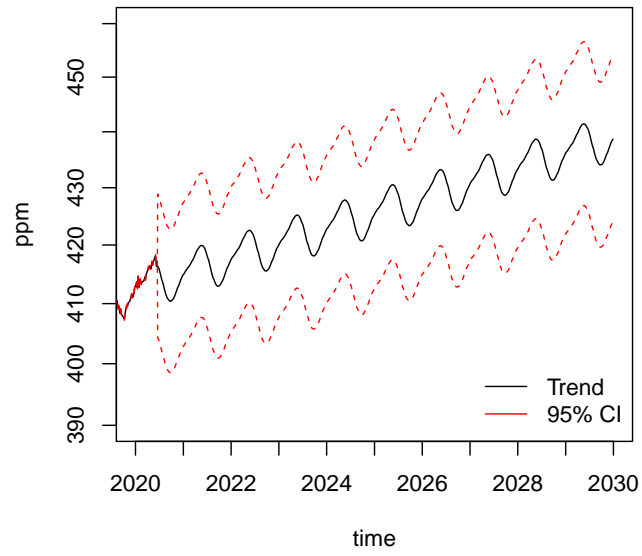


Figure 4: Predicted Carbon Dioxide Concentration

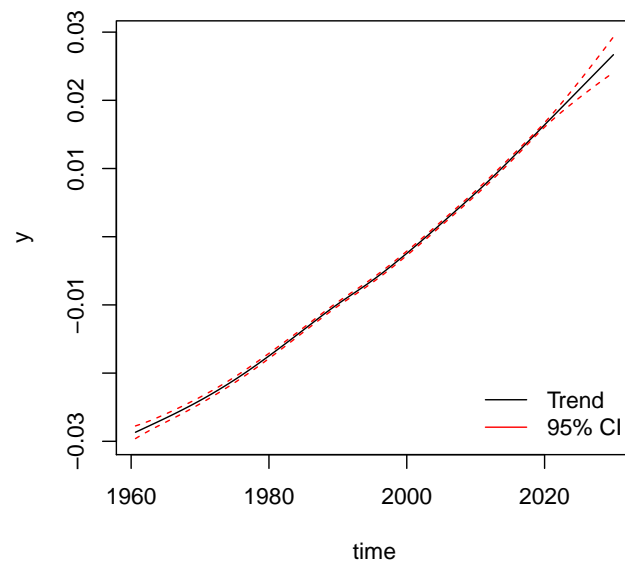
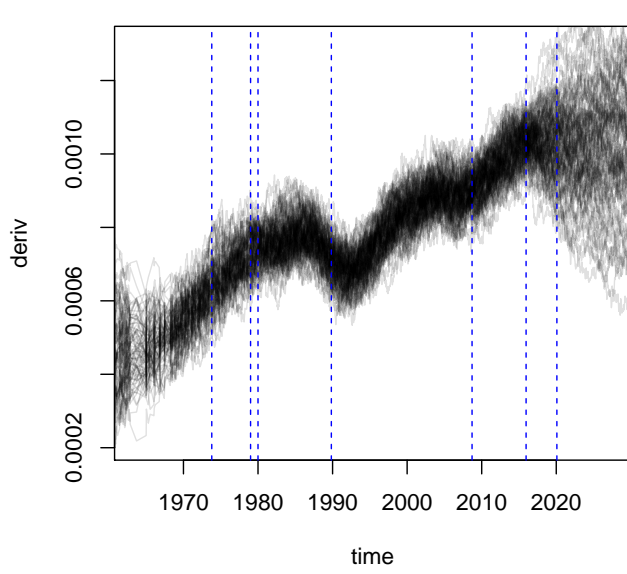


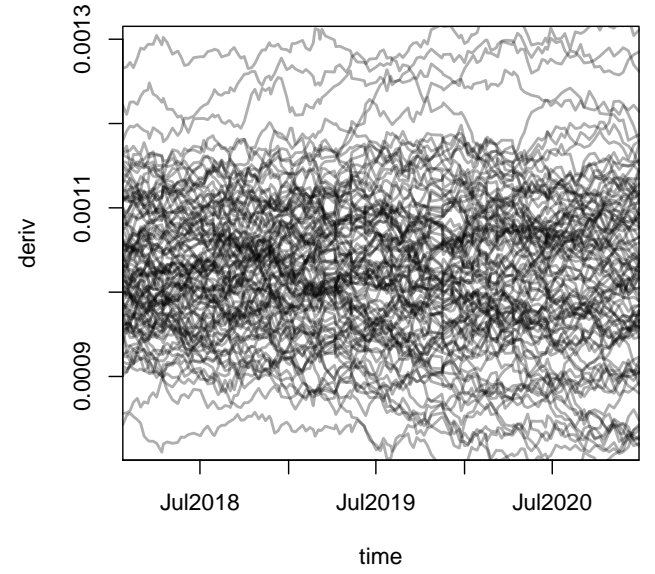
Figure 5: Random Effect. Yearly Change in Slope

Table 1: Summary outputs

| | 0.5quant | 0.025quant | 0.975quant |
|-----------------|------------|------------|------------|
| (Intercept) | 1.7790882 | 1.7788574 | 1.7793187 |
| sin12 | 0.0013296 | 0.0011829 | 0.0014762 |
| cos12 | -0.0004064 | -0.0005549 | -0.0002581 |
| sin6 | -0.0002836 | -0.0004305 | -0.0001368 |
| cos6 | 0.0003061 | 0.0001580 | 0.0004541 |
| sd for gamma | 0.0000457 | 0.0000453 | 0.0000459 |
| sd for timeInla | 0.0000490 | 0.0000475 | 0.0000503 |



(a) First Derivative



(b) Detailed First Derivative

Figure 6: Random Effect. Yearly Change in Slope

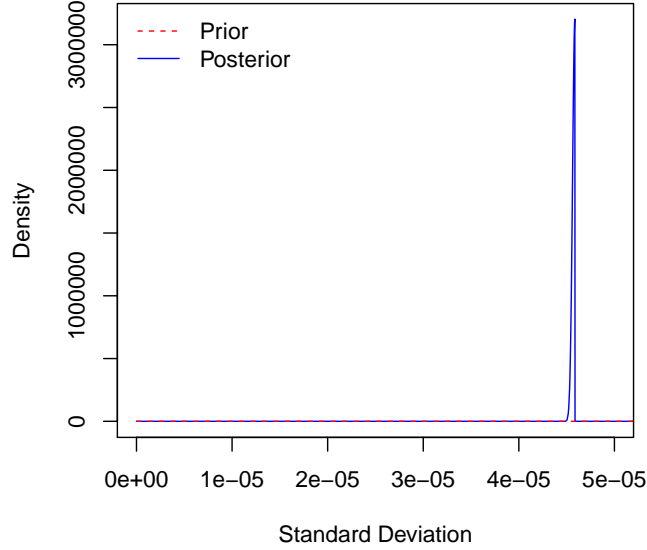


Figure 7: Prior and Posterior Distributions.

Discussion

The posterior standard deviation of $\frac{1}{\sqrt{\nu}}$ is $4.57\text{e-}5$ with a 95% credible interval ($4.53\text{e-}5$, $4.59\text{e-}5$) and σ is $4.90\text{e-}5$ with 95% credible interval ($4.75\text{e-}5$, $5.03\text{e-}5$) indicates a relatively small change in the average concentration of atmospheric carbon dioxide and annual changes in the slope. The seasonal trend shows atmospheric carbon dioxide reaching its peak concentration in May-June and lowest in October with relatively high certainty. The overall observed concentrations indicate increasing levels with noticeable plateaus near the 1989 Berlin wall collapse and the 2008 global economic recession. The first derivative of the random effect is positive everywhere indicating continuously increasing levels of atmospheric carbon dioxide. The oil shock events (1973-4, 1979) and global economic recession (1980-92) show an accelerating CO2 activity. There appears to be a dip, or slowdown in carbon dioxide production during the Berlin wall collapse (1989) and the global financial crisis (2007-8) followed by a rapid acceleration shortly after. There does not appear to be a significant slowdown in carbon dioxide production since the signing of the Paris Agreement (2015). The 2020 COVID-19 lockdown (since February) also does not appear to have caused a slowdown, however it may still also be too early to conclude as there is very high uncertainty.

The forecasted atmospheric carbon dioxide concentrations into 2025 suggests levels reaching around 425ppm (95% CI: 412ppm, 440ppm).

Conclusion

In conclusion, the global events of the 1989 Berlin wall collapse and the 2008 economic recession appears to have impacted the production of atmospheric carbon dioxide. In contrast, the 2015 Paris Agreement does not appear to have been as impactful as concentrations continue to steadily rise. The forecasted atmospheric carbon dioxide concentration suggests that we will surpass the 430ppm Paris Agreement benchmark before the year 2030. Therefore, global events have the potential to dramatically reduce atmospheric carbon

dioxide concentrations and it is imperative that considerable global measures are taken to limit productions and reduce the disasters of climate change.

References

- [1] Mitigation through reducing emissions and increasing carbon storage. In: UNDP. <https://www.undp.org/content/undp/en/home/2030-agenda-for-sustainable-development/planet/climate-change/reducingemissions-promoting-clean-energy-and-protecting-forest.html>. Accessed 20 Nov 2020
- [2] Monroe R (2013) Why are Seasonal CO₂ Fluctuations Strongest at Northern Latitudes? In: The Keeling Curve. <https://siweb.ucsd.edu/programs/keelingcurve/2013/05/07/why-are-seasonal-co2-fluctuationsstrongest-in-northern-latitudes/>. Accessed 20 Nov 2020
- [3] Bayesian computing. (2015). Spatial and Spatio-temporal Bayesian Models with R-INLA, 75-126. doi: 10.1002/9781118950203.ch4
- [4] Heckman, Nancy E. and James O. Ramsay (2000). “Penalized Regression with Model-Based Penalties”. In: The Canadian Journal of Statistics / La Revue Canadienne de Statistique 28.2, pp. 241–258. url: <http://www.jstor.org/stable/3315976>.
- [5] Simpson, Daniel, Haavard Rue, Andrea Riebler, Thiago G. Martins, and Sigrunn H. Sorbye (Feb. 2017). “Penalising Model Component Complexity: A Principled, Practical Approach to Constructing Priors”. In: Statistical Science 32.1, pp. 1–28. doi: 10.1214/16-STS576

Appendix

```
#load library
# disable some error checking in INLA
library(knitr)
library(bookdown)
library(kableExtra)
library('INLA', verbose=FALSE)

#load the data
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"))

#change format of date
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$date, co2s$co2, log = "y", cex = 0.3, col = "#00000040",
xlab = "time", ylab = "ppm")
```



```

plot(co2s[co2s$date > ISOdate(2015, 3, 1, tz = "UTC"),
c("date", "co2")], log = "y", type = "o", xlab = "time",
ylab = "ppm", cex = 0.5)

#time variables
#create a numeric time variable number of days since 1 Jan 2000
timeOrigin = as.Date("2000/1/1")
co2s$day = as.Date(co2s$date)
co2s$timeNumeric = as.numeric(co2s$day - timeOrigin)
co2s$timeYears = co2s$timeNumeric/365.25

#create 12 month cycles
co2s$sin12 = sin(2*pi*co2s$timeYears)
co2s$cos12 = cos(2*pi*co2s$timeYears)
#create 6 month cycles
co2s$sin6 = sin(2*pi*co2s$timeYears*2)
co2s$cos6 = cos(2*pi*co2s$timeYears*2)

#log transform
co2s$logCo2 = log(co2s$co2)

#fit model with  $s(t) = \text{sum of 4 sinusoids}$ 
res = lm(logCo2 ~ timeNumeric + sin12 + cos12 +
sin6 + cos6, data=co2s)

#create new data to predict seasonal effect
StimeYears = seq(-0.1, 1.1, len=1001)
newData = cbind(
sin12 = sin(2*pi*StimeYears),
cos12 = cos(2*pi*StimeYears),
sin6 = sin(2*2*pi*StimeYears),
cos6 = cos(2*2*pi*StimeYears))

#predict function on new data
res2 = cbind(
est = newData %%% res$coef[colnames(newData)],
se.fit = sqrt(diag(newData %%%
vcov(res)[
colnames(newData), colnames(newData)
] %%% t(newData)))
)
#95% prediction interval
res2ci = res2 %%% Pmisc::ciMat(0.95)

#create to forecast time points
toAdd = data.frame(day = seq(max(co2s$day) + 3, as.Date("2030/1/1"),
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)
co2ext$logco2 = log(co2ext$co2)

#added stability for 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(logco2 ~ 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)

#predict seasonal effect
newDataSeason = data.frame(
cos12 = cos(2*pi*StimeYears),
sin12 = sin(2*pi*StimeYears),
cos6 = cos(2*2*pi*StimeYears),
sin6 = sin(2*2*pi*StimeYears))

forLincombs = do.call(
inla.make.lincombs,
newDataSeason)

resInla2 = inla(co2res$.args$formula,
data=co2ext,
control.family = co2res$.args$control.family,
lincomb = forLincombs)

forX = as.Date('2020/1/1') + 365.25*StimeYears

{matplot(forX, exp(res2ci), xlab='time', ylab='co2 trend', type='l', xaxt='n',xaxs='i', log='y', col=c(
forXlab = seq(as.Date('2020/1/1'), len=12, by='3 months')
axis(1,as.numeric(forXlab), format(forXlab, '%b'))
legend("topright", col=c('black','red'), legend=c("Trend", "95% Pred Int."), bty="n", lty=1, lwd=1)
}

qcols = paste0(c('0.025','0.5','0.975'), 'quant')

{matplot(forX,

```

```

exp(resInla2$summary.lincomb.derived[,qcols]),
type='l', xlab='', ylab='relative co2',
xaxt='n', lty=c(2,1,2), col=c('red','black','red'),
xaxs='i')
axis(1,as.numeric(forXlab), format(forXlab, '%b'))
legend("topright", col=c('black','red'), legend=c("Trend", "95% Pred Int"), bty="n", lty=1, lwd=1)
}

#plotting parameters
quant = c('0.5quant','0.025quant','0.975quant')
recentYears = as.Date(c("2015/1/1", "2020/1/1"))
recentYears = seq(recentYears[1], recentYears[2], by = "1 year")

newx = as.Date(c("2020/1/1", "2030/1/1"))
forecastYears = seq(newx[1], newx[2], by="1 year")

Stime = timeOrigin + round(365.25 * co2res$summary.random$timeInla$ID)
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(co2ext$day, exp(co2res$summary.fitted.values[,
quant]), type = "l", col = c("black", "red", "red"), lty = c(1, 2,
2), xlab = "time", ylab = "ppm")
legend("bottomright", col=c('black','red'), legend=c("Trend", "95% CI"), bty="n", lty=1, lwd=1)
}

{matplot(co2ext$day,exp(co2res$summary.fitted.values[,
quant]), type = "l", col = c("black", "red", "red"), lty = c(1, 2,
2), log = "y", xlab = "time", ylab = "ppm", yaxt="n", xlim = c(as.Date("2015/1/1"), as.Date("2020/1/1")),
axis(1, as.numeric(recentYears), format(recentYears, "%Y"))
legend("bottomright", col=c('black','red'), legend=c("Trend", "95% CI"), bty="n", lty=1, lwd=1)
}

{matplot(co2ext$day,exp(co2res$summary.fitted.values[,
quant]), type = "l", col = c("black", "red", "red"), lty = c(1, 2,
2), log = "y", xlab = "time", ylab = "ppm", yaxt="n", xlim = c(as.Date("2020/1/1"), as.Date("2030/1/1")),
axis(1, as.numeric(forecastYears), format(forecastYears, "%Y"))
legend("bottomright", col=c('black','red'), legend=c("Trend", "95% CI"), bty="n", lty=1, lwd=1)
}

#Posterior samples
sampleList = INLA::inla.posterior.sample(100, co2res,
selection = list(timeInla = 0))
sampleMean = do.call(cbind, Biobase::subListExtract(sampleList,
"latent"))
sampleDeriv = apply(sampleMean, 2, diff)/diff(co2res$summary.random$timeInla$ID)

#random effect
{matplot(Stime, co2res$summary.random$timeInla[, quant],

```

```

type = "l", col = c("black", "red", "red"), lty = c(1, 2, 2), xlab = "time",
ylab = "y")
legend("bottomright", col=c('black','red'), legend=c("Trend", "95% CI"), bty="n", lty=1, lwd=1)
}

#first derivative
{matplot(Stime[-1], sampleDeriv, type = "l", lty = 1,
xaxs = "i", col = "#00000020", xlab = "time", ylab = "deriv",
ylim = quantile(sampleDeriv, c(0.00, 0.995)))
abline(v= as.Date("1973/10/19"), col="blue", lty=2)
abline(v= as.Date("1979/1/1"), col="blue", lty=2)
abline(v= as.Date("1980/1/1"), col="blue", lty=2)
abline(v= as.Date("1989/11/1"), col="blue", lty=2)
abline(v= as.Date("2008/9/15"), col="blue", lty=2)
abline(v= as.Date("2015/12/15"), col="blue", lty=2)
abline(v= as.Date("2020/02/1"), col="blue", lty=2)
}

#detailed first derivative
{matplot(Stime[toPlot], sampleDeriv[toPlot, ], type = "l",
lty = 1, lwd = 2, yaxs = "i", col = "#00000050",
xlab = "time", ylab = "deriv", xaxt = "n", ylim = quantile(sampleDeriv[toPlot,
], c(0.01, 0.995)))
axis(1, as.numeric(forX), format(forX, "%b%Y"))
}

theSd = Pmisc::priorPost(co2res)$summary[,quant]

knitr::kable(rbind(co2res$summary.fixed[, quant], theSd), caption = "Summary outputs")

#prior/posterior plot
theSdtotal= Pmisc::priorPost(co2res)

{plot(theSdtotal$`sd for gamma`$posterior, type='l', xlab='Standard Deviation', ylab='Density', col='blue',
lines(theSdtotal$`sd for gamma`$prior, col='red', lty=2)
legend("topleft", col=c('red','blue'), legend=c("Prior", "Posterior"), bty="n", lty=2:1, lwd=1)
}

```