Blair Bilodeau 1001232230
STA2201 Homework 3
April 9, 2019

1 Carbon Dioxide Data

1.1 Introduction

For the last 60 years, the Scripps CO_2 program has been observing Carbon Dioxide levels in Hawaii. The goal of this report is to build a model and assess how quickly these values are rising, how they change over the course of the year, and to predict future concentration levels.

Figure 1 displays concentration levels over the period of 1960-2018, as well as the trend in recent years. Clearly, concentration levels have been steadily increasing, although with some sort of periodic trend within each year.
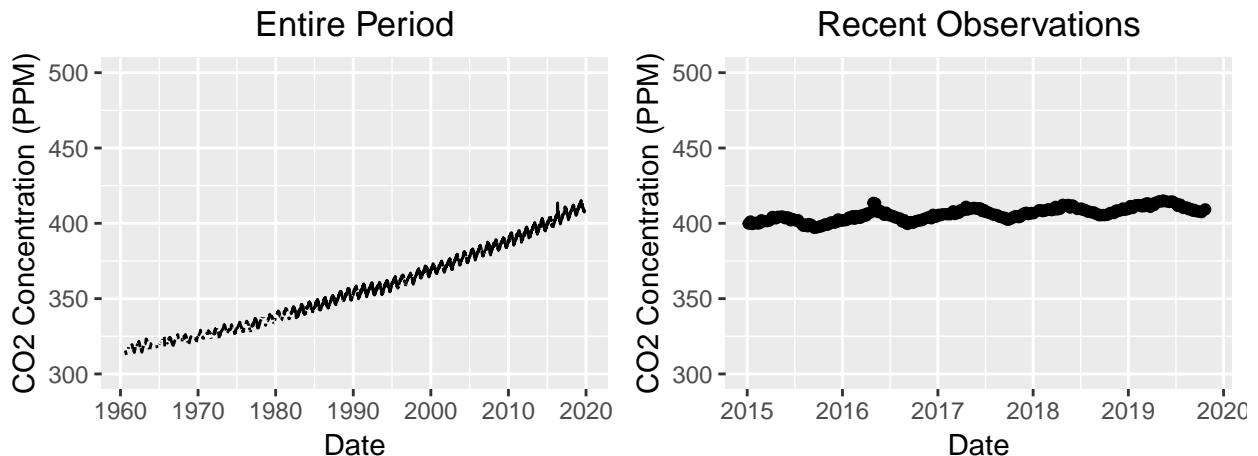

Figure 1: Visualization of Carbon Dioxide concentration observations made in Hawaii over the entire period and recently.

1.2 Model

Let X_1 be the number of days since Jan 1, 1980 that the measurement was taken on and X_2 be the week of the year in which the measurement was taken. Note that X_1 is negative for measurements taken before 1980. Then, the model fit was a GAM of the form

$$Y_i \sim \text{Gamma}(\mu_i/\phi, \phi)$$
$$\log(\mu_i) = \beta_0 + f_{i,1}(X_1; \nu_1) + f_{i,2}(X_2, \nu_2),$$

where the $f_{i,j}$'s are linear combinations of spline basis functions.

This model allows for both a non-linear annual trend, which means the rate of increase can be estimated, and a non-linear subannual trend, which is particularly useful for data that appears

to have periodic seasonality. A gamma distribution was used since concentration levels must be positive.

Another model was tested using month instead of week for the second smoothing function, although since week still has a significant effect in the model it is preferable as it allows for a smoother function. Finally, once the weekly trend is accounted for, any additional monthly trend becomes insignificant. X_1 and X_2 could also be included with linear coefficients, although then the effect of the smoothing parameters is almost minimal and it becomes more difficult to estimate the derivatives.

1.3   Results

First, consider the overall fit of the model. Figure 2 shows that the model is approximating the trend of the data fairly well over the long term. Note that there are error bars on the estimated values in the plot, although at this macro scale the errors are essentially null.
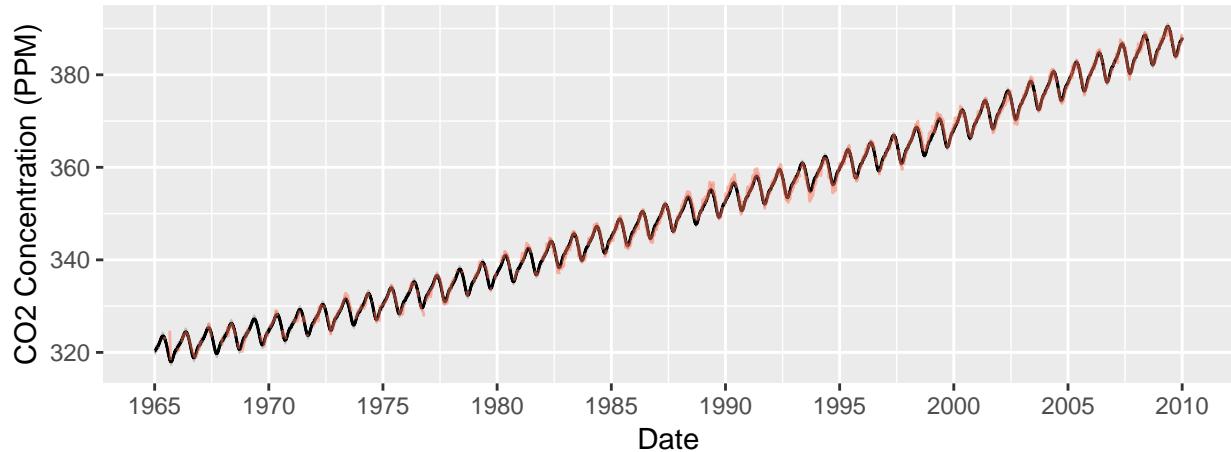

Figure 2: GAM model (—) compared to observed values (—) for Carbon Dioxide concentration levels in Hawaii over the sample period of 1965-2010.

To understand the performance of the model at a more granular scale, Figure 3 compares the model for the sample period of 2010-2015. It is now more clear that the model is capturing the overall trend of the data, while perhaps oversmoothing the individual estimates.
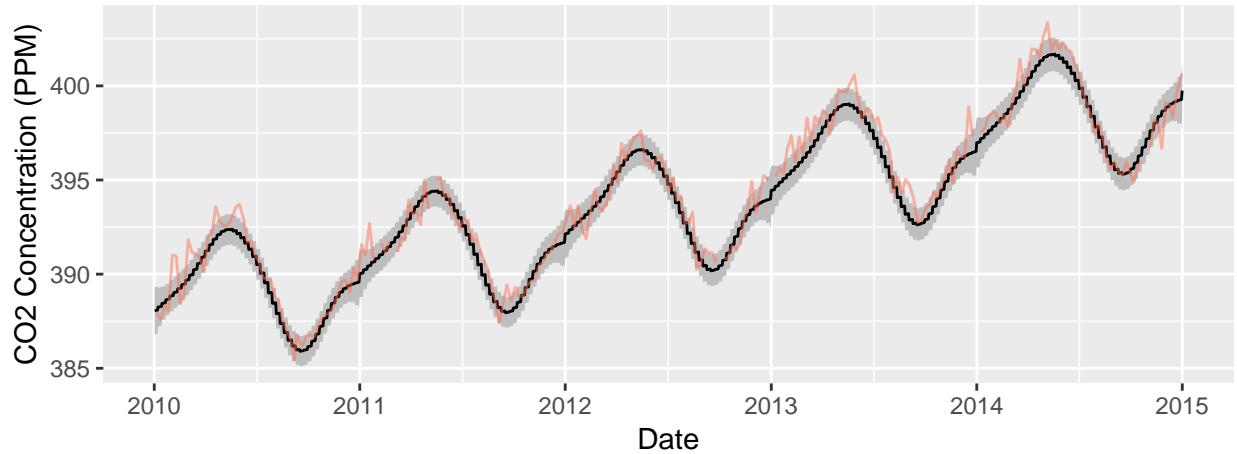
Figure 3: GAM model (—) compared to observed values (—) for Carbon Dioxide concentration levels in Hawaii over the sample period of 2010-2015.

1.3.1 Rate of Change

Next, consider the rate at which the concentration levels are increasing. To determine this, a finite differences approach is used. That is, the model is used to predict for two sets of covariates that are within ε of each other, and then the difference of the predicted values divided by ε is an estimate for the derivative of the smoothed functions at that point. The same process is repeated on the first derivative estimates to get the second derivative estimates.
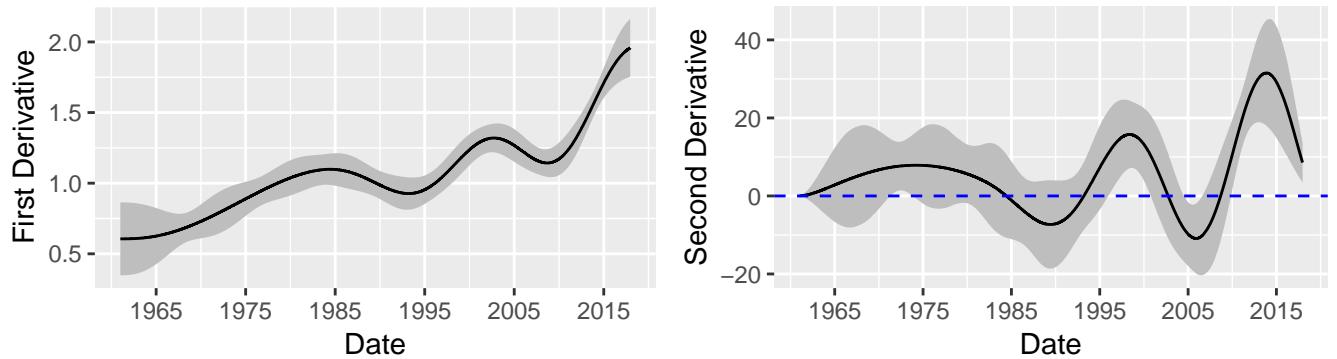

Figure 4: Estimated first and second derivatives with 95% error bars for $f_{i,1}$ from GAM model for Carbon Dioxide concentration levels.

1.3.2 Subannual Trend

In addition to the long term increasing trend from year to year, there is clearly seasonality within each year. The function $f_{i,2}$ captures this, and is estimated with 95% error bars in Figure 5.
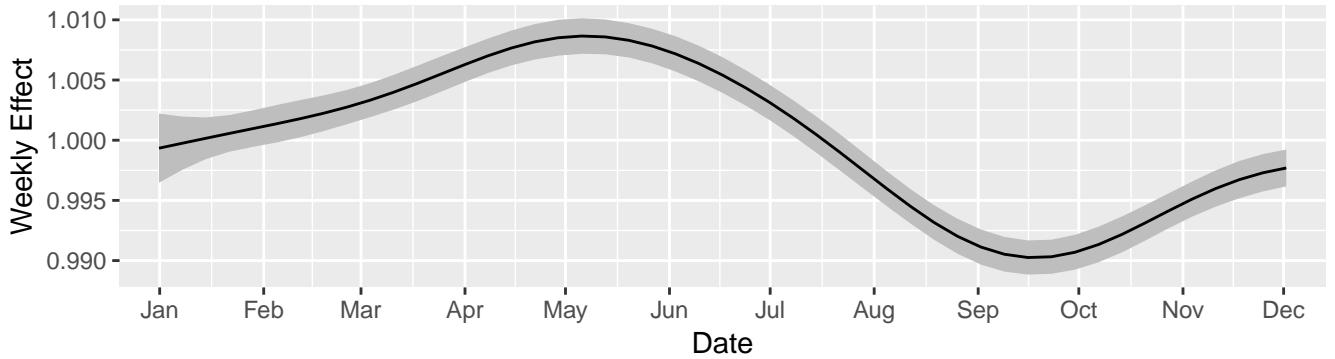
Figure 5: Estimated curve for weekly effect on Carbon Dioxide levels in GAM model with 95% error bars.

1.3.3 Future Extrapolation

Finally, the model can be used to make predictions for future Carbon Dioxide concentration levels, which are presented in Figure 6.
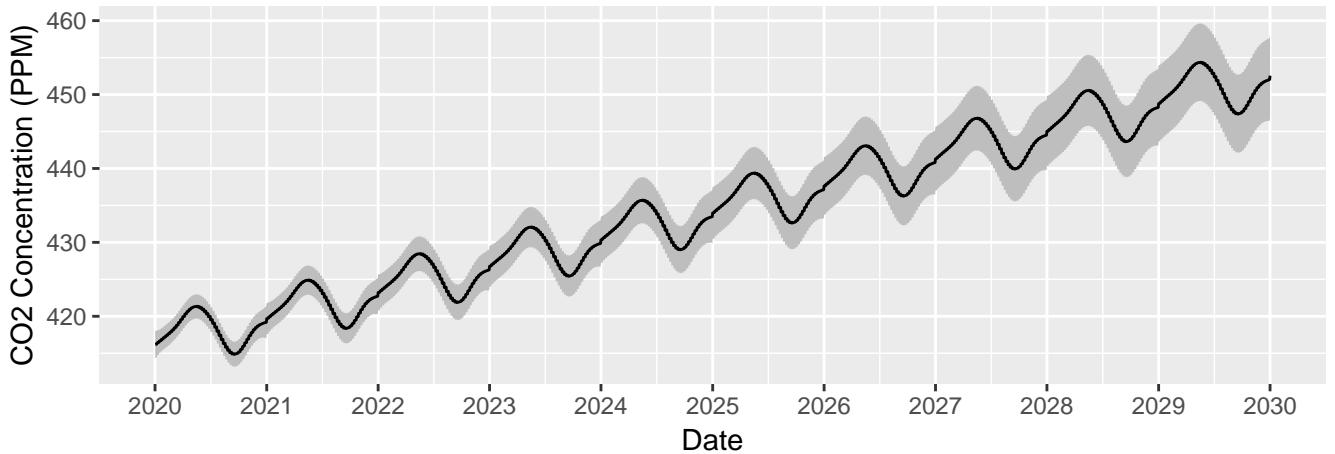

Figure 6: Predicted Carbon Dioxide levels for the next decade from GAM model.

1.4 Conclusion

Overall, it seems that the GAM model captures most of the trend in the Carbon Dioxide data. The derivative plots show that perhaps the CO_2 levels are on the cusp of slowing their rate of increase, although at the very end of the data the second derivative is still positive and consequently the levels are still increasing faster. Further, there are clear declines in the rate of increase around 1985 and 2000. However, these periods of decline precede the 1989 collapse of the Soviet Union and the 2008 financial crisis. Another analysis could investigate whether these levels could be used as indicators for anticipating global economic events. Looking at the subannual trend, the first half of the year clearly has higher levels than in the second half, and in particular October has lower levels than March. Finally, while the prediction is difficult to estimate seven years beyond the end of the

data, it seems that by 2025 levels will still be just shy of the 430 ppm mark, although they are likely to cross that the next year.

2 Malaria Data

2.1 Introduction

Testing children for Malaria in remote areas of Gambia can be extremely expensive due to the invasive nature of the blood test. Data has been gathered which measures whether or not a child has Malaria along with information about how they are protected from mosquitos (which transmit the disease) and the vegetation level around them (a proxy for how large the mosquito population will be). This report aims to determine if these factors can be used to predict whether a child has Malaria, which would allow for non-invasive surveys to inform decisions about where best money could be spent on Malaria testing and treatment.

Figure 7 displays the locations of villages where Malaria testing was performed and where positive tests were found, while Figure 8 shows the vegetation levels for the same area.
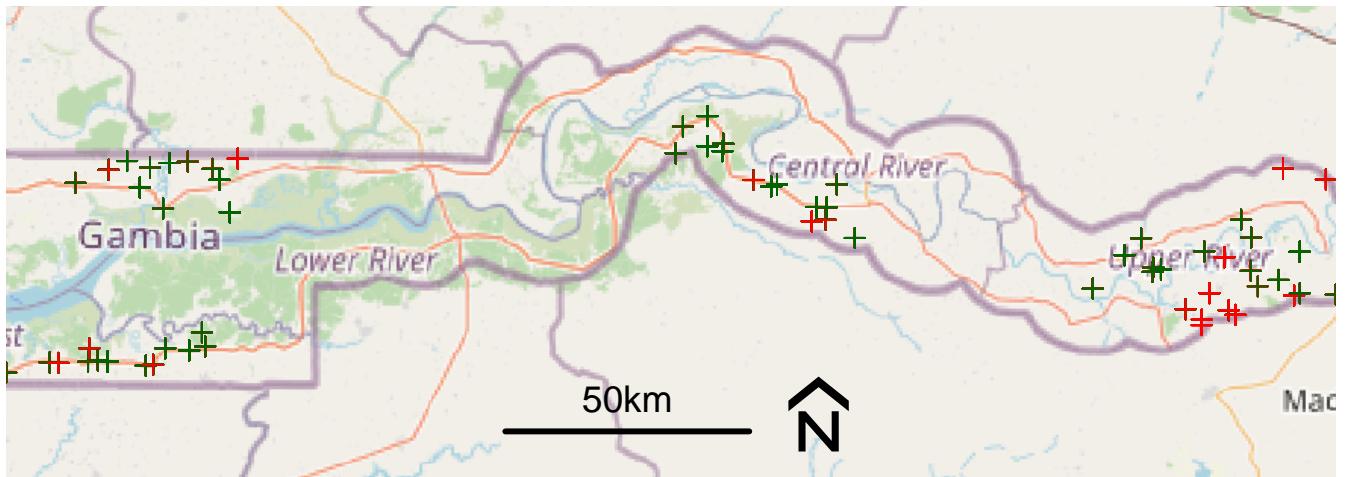

Figure 7: Locations of Gambian villages surveyed for positive (—) and negative (—) Malaria tests in children.
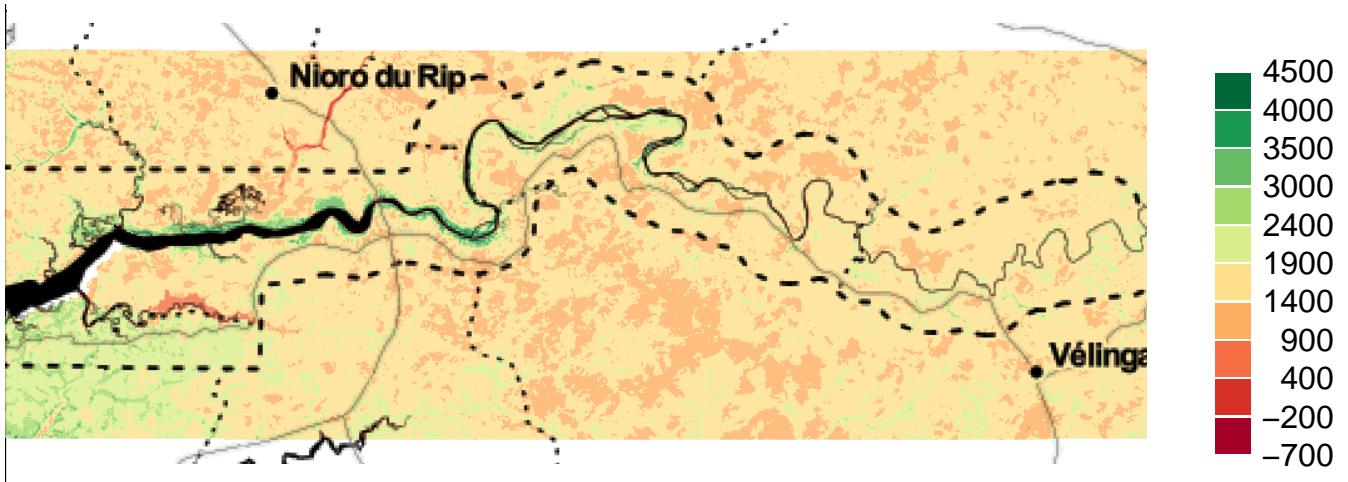
Figure 8: Vegetation levels for area of Gambia surveyed.

2.2 Methods

For the ith child, let Y_i be an indicator for whether or not they test positive for Malaria and s_i be their location. Define the covariates as

$X_1(s_i)$: Child's age, in days,

$X_2(s_i)$: Indicator for whether the child sleeps with a mosquito net; 1 for yes and 0 for no,

$X_3(s_i)$: Indicator for whether the child's mosquito net is treated with insecticide; 1 for yes and 0 for no,

$X_4(s_i)$: Indicator for whether the child's village has a public health centre; 1 for yes and 0 for no,

$X_5(s_i)$: Average vegetation level in the child's village.

Then, the model fit was of the form

$$Y_i \mid U(s_i), \beta, \sigma, \phi \overset{ind}{\sim} \text{Bernoulli}(\lambda(s_i))$$

$$\log\left(\frac{\lambda(s_i)}{1 - \lambda(s_i)}\right) \mid U(s_i), \beta = \beta_0 + \beta_1 X_i^{(1)}(s_i) + \beta_2 X_i^{(2)}(s_i) + \beta_3 X_i^{(3)}(s_i) + \beta_4 X_i^{(4)}(s_i) + \beta_5 X_i^{(5)}(s_i) + U(s_i)$$

$$\text{cov}[U(s_i), U(s_j)] = \sigma^2 \rho\left[(s_i - s_j)/\phi; \nu = 1\right]$$

with the prior distributions

$$\beta_0 \sim \mathcal{N}(0, 10^2)$$
$$\beta \sim \mathcal{N}(0, 10^2 I)$$
$$\phi \sim \text{Exponential such that } P(\phi/1000 < 60) = 0.5$$
$$\sigma \sim \text{Exponential such that } P(\sigma > 0.1) = 0.01.$$

Notice that ρ is the Matern covariance function. That is, the spatial variation is modelled as depending on the locations nearest you in a Gaussian manner, with this dependence decaying exponentially as the locations get farther away. The interpretation of this model is that the probability of a child testing positive for Malaria is dependent on the covariates and some additional variation due to their spatial coordinates.

2.3 Results

The results of fitting this model are contained in Table 1, which has quantiles for the multiplicative effects of $\exp(\beta)$ on the odds ratio. Based on the credible intervals, it seems that having a mosquito net will lower the risk of malaria, and to nearly the same extent treating this net will further lower the risk. Age and vegetation level have virtually no impact, and we cannot say anything about the significance of whether a village has a public health centre. To better understand the random effects, Figure 9 displays the prior and posterior distributions for σ and $\rho/1000$. Also, notice that the standard deviation is reasonably high.

Table 1: Quantile estimates on the natural scale for parameter effects on odds ratio for logistic spatial model fit to Malaria data.

|	Mean	2.5% Quant	97.5% Quant
(Intercept)	0.239	0.045	0.783
Age	1.001	1.000	1.001
Net use: Yes	0.726	0.531	0.968
Treated: Yes	0.726	0.496	1.023
Public Health Centre	0.819	0.550	1.185
Vegetation Level	1.001	1.000	1.001
$\phi/1000$	31.313	16.637	57.910
σ	1.026	0.684	1.615
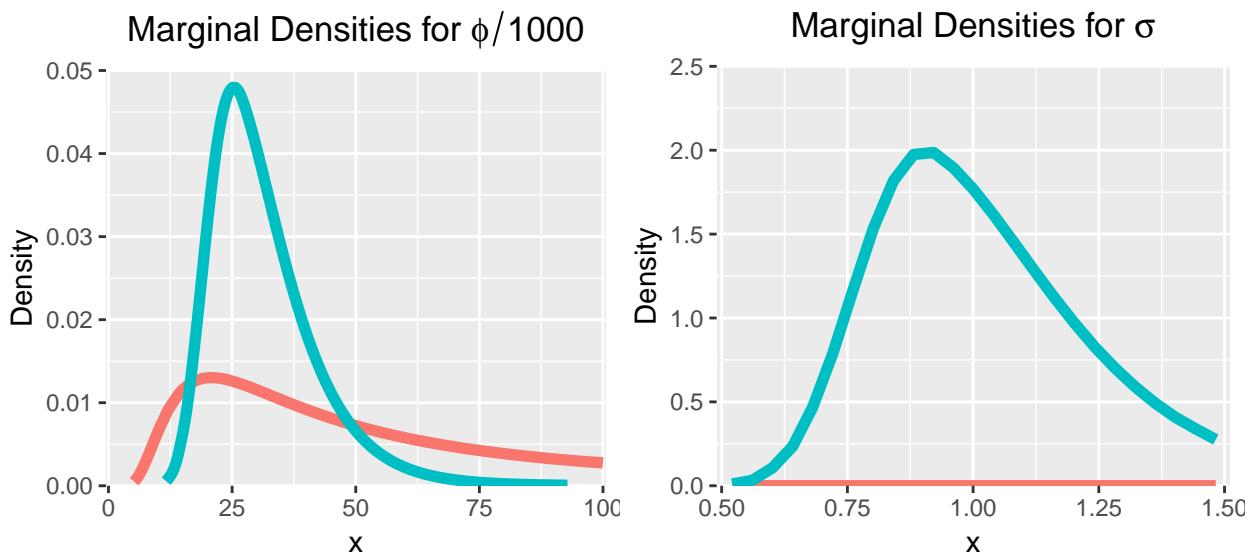
Figure 9: Prior (—) and posterior (—) marginal densities for standard deviation and scaled range of Malaria model.

Next, Figure 10 shows the predicted mean of $\lambda(s)$ conditional on the covariates. This is fairly consistent outside of the clusters where children were actually tested for Malaria in the dataset. To complement this, Figure 11 shows the probability of the random effects being greater than zero, which again is only informative where data was actually collected.
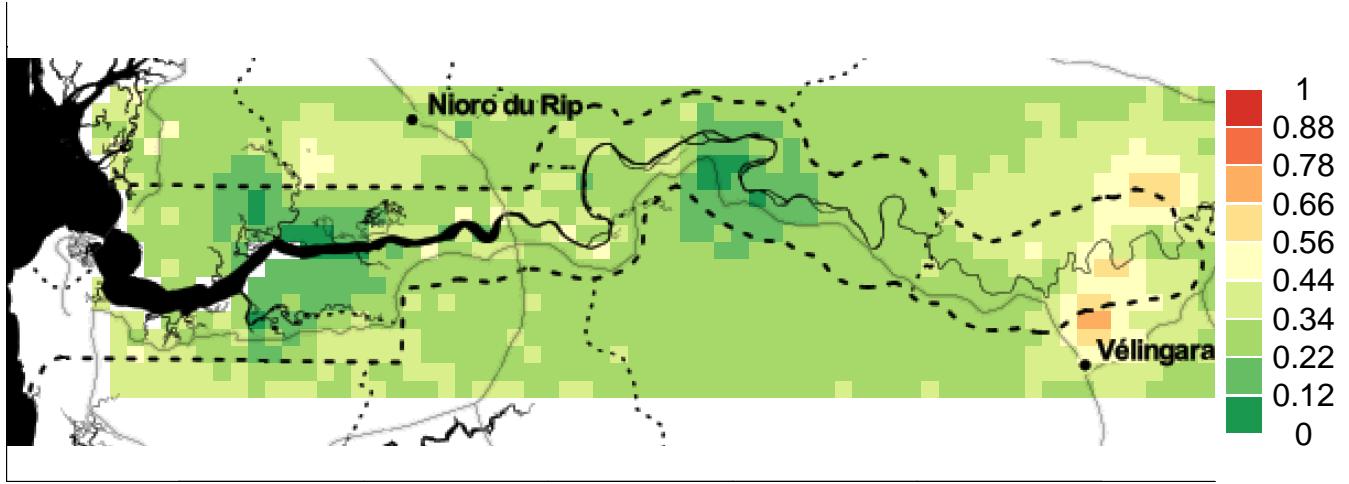
Figure 10: $E\left[\lambda(s) \mid Y\right]$ for Malaria model.
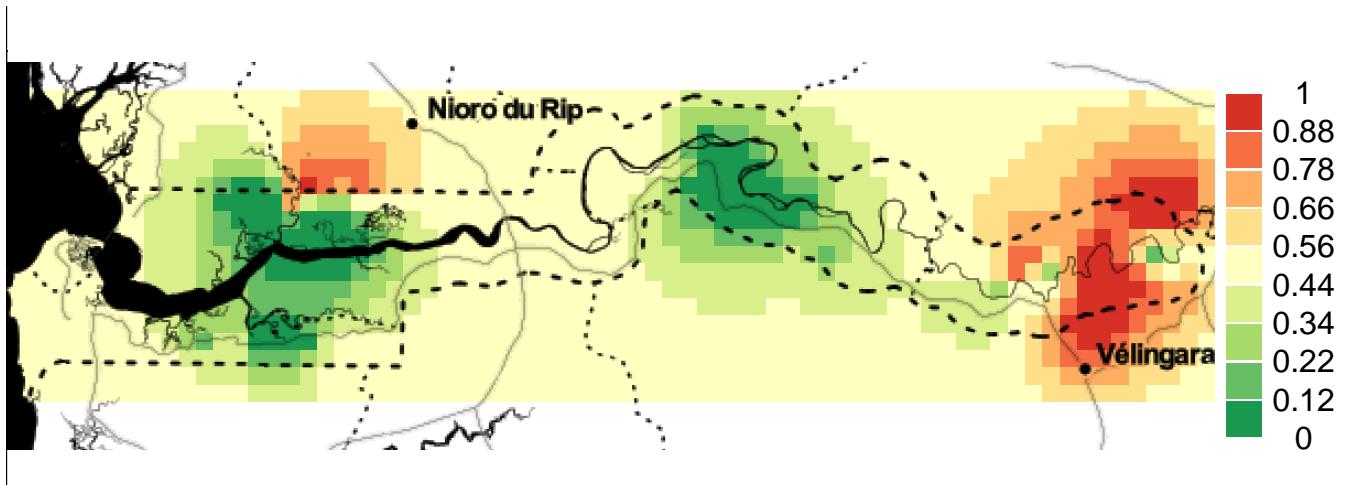
Figure 11: $P\left[U(s) > 0 \mid Y\right]$ for Malaria model.

Finally, Figures 12 and 13 show the 2.5% and 97.5% quantiles for $\lambda(s)$. Notice that outside of where data was available, this credible interval ranges virtually from zero to one. That is, there is no predictive information.
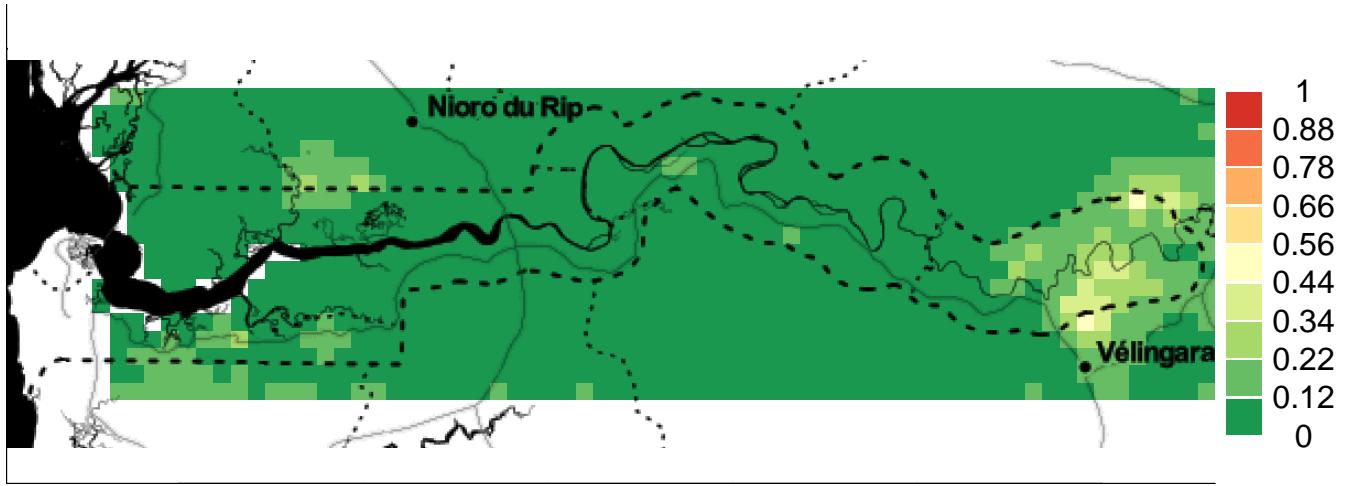
Figure 12: 2.5% posterior quantile for $\lambda(s)$ in Malaria model.
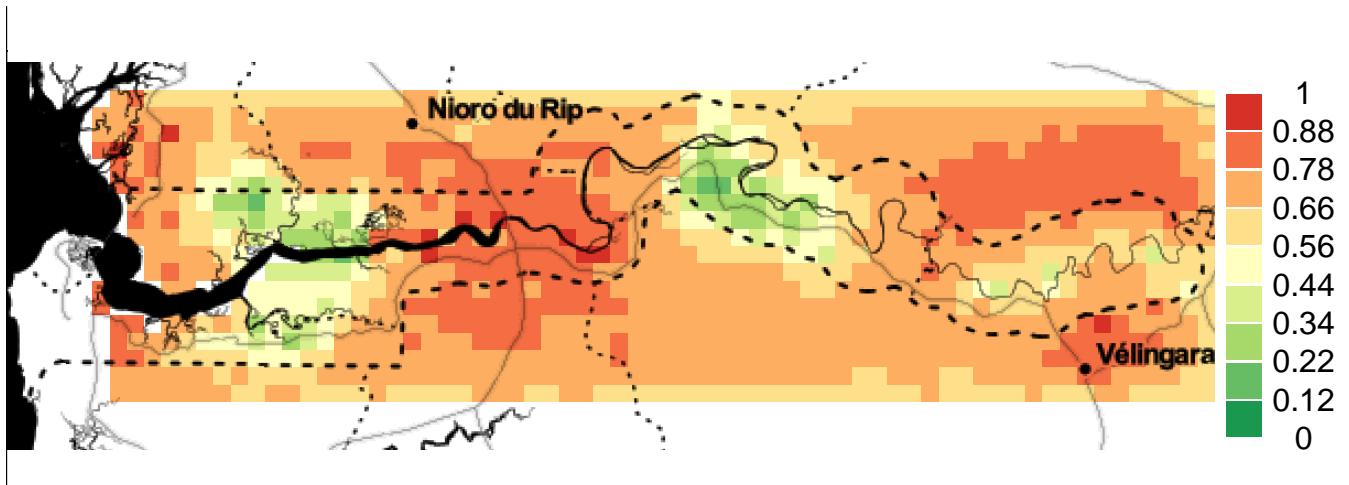
Figure 13: 97.5% posterior quantile for $\lambda(s)$ in Malaria model.

2.4 Conclusion

While the model leads to the conclusion that it is preferable to have a mosquito net, it cannot be used for the purpose of determining new locations that are susceptible to Malaria. The residual spatial variation is simply too high, as evidenced by the 95% prediction intervals, to make any meaningful conclusion from just the covariates available.

Appendix A – Code

```r
# Blair Bilodeau
# 1001232230
# STA2201 Homework 3

### Setup

knitr::opts_chunk$set(fig.pos = 'h')

def.chunk.hook  <- knitr::knit_hooks$get("chunk")
knitr::knit_hooks$set(chunk = function(x, options) {
  x <- def.chunk.hook(x, options)
  ifelse(options$size != "normalsize", paste0("\\", options$size,"\n\n", x, "\n\n \\normalsize"), x)
})

# Formatting Libraries
library(stargazer)
library(ggplot2)
library(latex2exp)
library(gridExtra)
library(tseries)
library(xtable)
library(plyr)
library(knitr)
library(mapmisc)
library(scales)
library(mapmisc)
library(raster)
options(xtable.comment = FALSE)

# Analysis Libraries
library(MEMSS)
library(Pmisc)
library(nlme)
library(INLA)
library(MASS)
library(tidyverse)
library(abind)
library(mgcv)
library(lubridate)
library(geostatsp)

## Retrieve data
#cUrl <- "http://scrippsco2.ucsd.edu/assets/data/atmospheric/stations/flask_co2/daily/daily_flask_co2_mlo.csv"
cUrl <- "https://scrippsco2.ucsd.edu/assets/data/atmospheric/stations/flask_co2/daily/daily_flask_co2_mlo.csv"
co2s = read.table(cUrl, header = FALSE, sep = ",", skip = 69,
                  stringsAsFactors = FALSE, col.names = c("day", "time", "junk1", "junk2", "Nflasks", "quality", "co2"))
# remove low-quality measurements
co2s[co2s$quality > 2, "co2"] = NA
co2s$date <- as.POSIXct(strptime(paste(co2s$day, co2s$time), format = "%Y-%m-%d %H:%M", tz = "UTC"))

# add in extra columns
co2s$week <- week(co2s$date)
timeOrigin <- ISOdate(1980, 1, 1, 0, 0, 0, tz = "UTC")
co2s$days <- as.numeric(difftime(co2s$date, timeOrigin, units = "days"))

# Exploratory plots
co2.p1 <- ggplot(co2s, aes(x=date, y=co2)) +
  geom_line() +
  xlab('Date') + ylab('CO2 Concentration (PPM)') +
  ggtitle('Entire Period') +
  theme(plot.title = element_text(hjust = 0.5)) +
  ylim(300,500) +
  scale_x_datetime(breaks=seq(as.POSIXct("1960-01-01 00:00:00 UTC"),
                              as.POSIXct("2020-01-01 00:00:00 UTC"), "10 years"),
                   labels=date_format("%Y"))

co2.p2 <- ggplot(co2s[co2s$date > ISOdate(2015, 1, 1, tz = "UTC"), c("date", "co2")], aes(x=date, y=co2)) +
  geom_line() +
```

```
  geom_point() +
  xlab('Date') + ylab('CO2 Concentration (PPM)') +
  ggtitle('Recent Observations') +
  theme(plot.title = element_text(hjust = 0.5)) +
  ylim(300,500) +
  scale_x_datetime(breaks=seq(as.POSIXct("2015-01-01 00:00:00 UTC"),
                              as.POSIXct("2020-01-01 00:00:00 UTC"), "1 year"),
                   labels=date_format("%Y"))

grid.arrange(co2.p1, co2.p2, ncol=2)

# GAM model
cGam <- gam(co2 ~ s(days) + s(week), data=co2s, family=Gamma(link=log))

# plot model against data
newX = data.frame(date = as.POSIXct(seq(ISOdate(1961, 1, 1, 0, 0, 0, tz = "UTC"), by = "1 days", length.out = 365 * 100)))
newX$days = as.numeric(difftime(newX$date, timeOrigin, units = "days"))
newX$week <- week(newX$date)

# full observation history
start <- ISOdate(1965, 1, 1, tz = "UTC")
end <- ISOdate(2010, 1, 1, tz = "UTC")
newX.window <- newX[newX$date >= start & newX$date <= end,]
co2s.window <- co2s[co2s$date >= start & co2s$date <= end,]
cGam.fit <- predict(cGam, newX.window, se.fit=T)
co2.fit.samp <- ggplot() +
  geom_ribbon(aes(x=newX.window$date,
                  ymin=exp(cGam.fit$fit - 1.96 * cGam.fit$se.fit),
                  ymax=exp(cGam.fit$fit + 1.96 * cGam.fit$se.fit)),
              fill='grey') +
  geom_line(aes(x=newX.window$date, y=exp(cGam.fit$fit)), colour='black') +
  geom_line(aes(x=co2s.window$date, y=co2s.window$co2), colour='coral1', alpha=0.5) +
  xlab('Date') + ylab('CO2 Concentration (PPM)') +
  scale_x_datetime(breaks=seq(as.POSIXct("1965-01-01 00:00:00 UTC"),
                              as.POSIXct("2010-01-01 00:00:00 UTC"), "5 years"),
                   labels=date_format("%Y"))
co2.fit.samp

# sample observation history
start2 <- ISOdate(2010, 1, 1, tz = "UTC")
end2 <- ISOdate(2015, 1, 1, tz = "UTC")
newX.window2 <- newX[newX$date >= start2 & newX$date <= end2,]
co2s.window2 <- co2s[co2s$date >= start2 & co2s$date <= end2,]
cGam.fit2 <- predict(cGam, newX.window2, se.fit=T)
co2.fit.samp2 <- ggplot() +
  geom_ribbon(aes(x=newX.window2$date,
                  ymin=exp(cGam.fit2$fit - 1.96 * cGam.fit2$se.fit),
                  ymax=exp(cGam.fit2$fit + 1.96 * cGam.fit2$se.fit)),
              fill='grey') +
  geom_line(aes(x=newX.window2$date, y=exp(cGam.fit2$fit)), colour='black') +
  geom_line(aes(x=co2s.window2$date, y=co2s.window2$co2), colour='coral1', alpha=0.5) +
  xlab('Date') + ylab('CO2 Concentration (PPM)') +
  scale_x_datetime(breaks=seq(as.POSIXct("2010-01-01 00:00:00 UTC"),
                              as.POSIXct("2015-01-01 00:00:00 UTC"), "1 year"),
                   labels=date_format("%Y"))
co2.fit.samp2

## daily first and second derivatives
x.mesh <- as.POSIXct(seq(ISOdate(1961, 1, 1, 0, 0, 0, tz = "UTC"), by = "1 days", length.out = 365 * 57))
x.mesh2 <- x.mesh + 60*60*24 # one day later
x.mesh0 <- x.mesh - 60*60*24 # one day before
x.mesh.df <- data.frame(date=x.mesh, days=as.numeric(difftime(x.mesh, timeOrigin, units = "days")), week=week(x.mesh))
x.mesh2.df <- data.frame(date=x.mesh2, days=as.numeric(difftime(x.mesh2, timeOrigin, units = "days")), week=week(x.mesh2))
x.mesh0.df <- data.frame(date=x.mesh0, days=as.numeric(difftime(x.mesh0, timeOrigin, units = "days")), week=week(x.mesh0))
X0 <- predict(cGam, x.mesh0.df, type='lpmatrix')
X1 <- predict(cGam, x.mesh.df, type="lpmatrix")
X2 <- predict(cGam, x.mesh2.df, type="lpmatrix")

Xd <- (X2 - X1) * 60*60*24
Xi <- Xd*0
Xi[,2:10] <- Xd[,2:10]
fd1 <- Xi %*% coef(cGam)
```

```r
fd1.sd <- sqrt(rowSums(Xi %*% cGam$Vp * Xi))

co2.deriv1 <- ggplot() +
  geom_ribbon(aes(x=x.mesh, ymin=fd1-2*fd1.sd, ymax=fd1+2*fd1.sd), fill='grey') +
  geom_line(aes(x=x.mesh, y=fd1), colour='black') +
  xlab('Date') + ylab('First Derivative') +
  scale_x_datetime(breaks=seq(as.POSIXct("1965-01-01 00:00:00 UTC"),
                              as.POSIXct("2015-01-01 00:00:00 UTC"), "10 years"),
                   labels=date_format("%Y"))

Xdd <- (X2 + X0 - 2*X1) * (60*60*24)^2
Xii <- Xdd*0
Xii[,2:10] <- Xdd[,2:10]
fd2 <- Xii %*% coef(cGam)
fd2.sd <- sqrt(rowSums(Xii %*% cGam$Vp * Xii))

co2.deriv2 <- ggplot() +
  geom_ribbon(aes(x=x.mesh, ymin=fd2-2*fd2.sd, ymax=fd2+2*fd2.sd), fill='grey') +
  geom_line(aes(x=x.mesh, y=fd2), colour='black') +
  geom_hline(yintercept=0, linetype='dashed', colour='blue') +
  xlab('Date') + ylab('Second Derivative') +
  scale_x_datetime(breaks=seq(as.POSIXct("1965-01-01 00:00:00 UTC"),
                              as.POSIXct("2015-01-01 00:00:00 UTC"), "10 years"),
                   labels=date_format("%Y"))

grid.arrange(co2.deriv1, co2.deriv2, ncol=2)
```

```r
## weekly trend of data
weekX <- data.frame(date = as.POSIXct(seq(ISOdate(2000, 1, 1, 0, 0, 0, tz = "UTC"), by = "1 week", length.out = 49)))
weekX$days <- as.numeric(difftime(weekX$date, timeOrigin, units = "days"))
weekX$week <- week(weekX$date)
co2.trend2 <- predict(cGam, weekX, type='terms', se.fit=T)

co2.week <- ggplot() +
  geom_ribbon(aes(x=weekX$date,
                  ymin=exp(co2.trend2$fit[,'s(week)']-1.96*co2.trend2$se.fit[,'s(week)']),
                  ymax=exp(co2.trend2$fit[,'s(week)']+1.96*co2.trend2$se.fit[,'s(week)'])),
              fill='grey') +
  geom_line(aes(x=weekX$date, y=exp(co2.trend2$fit[,'s(week)'])), colour='black') +
  xlab('Date') + ylab('Weekly Effect') +
  scale_x_datetime(breaks=seq(as.POSIXct("2000-01-01 00:00:00 UTC"),
                              as.POSIXct("2001-01-01 00:00:00 UTC"), "1 month"),
                   labels=date_format("%b"), expand=c(0.035,0.035))
co2.week
```

```r
# future predictions
start <- ISOdate(2020, 1, 1, tz = "UTC")
end <- ISOdate(2030, 1, 1, tz = "UTC")
newX.window <- newX[newX$date >= start & newX$date <= end,]
cGam.fit <- predict(cGam, newX.window, se.fit=T)
co2.fit.pred <- ggplot() +
  geom_ribbon(aes(x=newX.window$date,
                  ymin=exp(cGam.fit$fit - 1.96 * cGam.fit$se.fit),
                  ymax=exp(cGam.fit$fit + 1.96 * cGam.fit$se.fit)),
              fill='grey') +
  geom_line(aes(x=newX.window$date, y=exp(cGam.fit$fit)), colour='black') +
  xlab('Date') + ylab('CO2 Concentration (PPM)') +
  scale_x_datetime(breaks=seq(as.POSIXct("2020-01-01 00:00:00 UTC"),
                              as.POSIXct("2030-01-01 00:00:00 UTC"), "1 year"),
                   labels=date_format("%Y"))
co2.fit.pred
```

```r
## read in data
data("gambiaUTM")

## eviMean
eUrl <- "http://pbrown.ca/teaching/astwo/data/eviMean.RData"
eFile <- basename(eUrl)
if (!file.exists(eFile)) download.file(eUrl, eFile)
load(eFile)
```

```
## plot village locations
gambiaUTM.cols <- ifelse(gambiaUTM$pos == 0, "darkgreen", "red")

gmap <- openmap(gambiaUTM, fact = 2)
map.new(gambiaUTM)
plot(gmap, add = TRUE)
points(gambiaUTM, pch=3, col=gambiaUTM.cols)
scaleBar(gambiaUTM, "bottom", cex = 1.2, bty = "n")

## plot vegetation level
eviMean.gambia <- projectRaster(eviMean, crs=crs(gambiaUTM))
eviMean.cols <- colourScale(eviMean.gambia, style = "equal", breaks = 11, dec = 1, col = "RdYlGn",
                            rev = FALSE, opacity = 0.8)
gmap2 = tonerToTrans(openmap(gambiaUTM, path = "stamen-toner", fact = 2))
map.new(gambiaUTM, legendRight = 0.85)
plot(eviMean.gambia, add=TRUE, col=eviMean.cols$colOpacity, breaks=eviMean.cols$breaks, legend=FALSE)
plot(gmap2, add=TRUE, maxpixels=10^6)
eviMean.cols$breaks <- round_any(eviMean.cols$breaks, 100)
legendBreaks("right", eviMean.cols, outer = TRUE, bty = "n", inset = 0)

gborder <- raster::getData("GADM", country = "GMB", level = 0)
gborder <- spTransform(gborder, projection(gambiaUTM))

malaria.fit <- glgm(pos ~ age + netuse + treated + phc + evi,
             data = gambiaUTM,
             grid = squareRaster(gborder, 70),
             shape=1,
             family = "binomial",
             control.fixed=list(
               mean.intercept=0, prec.intercept=1/10^2,
               mean=0, prec=1/10^2),
             prior = list(sd = c(u = 0.1, alpha = 0.01),
                          range = c(u = 60 * 1000, alpha = 0.5)),
             covariates = list(evi = eviMean))

malaria.table <- malaria.fit$parameters$summary[, c(1, 3, 5)]
malaria.table[1:6, 2:3] <- exp(malaria.table[1:6, 2:3])
malaria.names <- malaria.fit$inla$names.fixed
for (row in 1:6){
  malaria.table[row,1] <- inla.emarginal(function(x) exp(x), malaria.fit$inla$marginals.fixed[[malaria.names[row]]])
}
rownames(malaria.table) <- c('(Intercept)',
                             'Age',
                             'Net use: Yes',
                             'Treated: Yes',
                             'Public Health Centre',
                             'Vegetation Level',
                             '$\\phi$/1000',
                             '$\\sigma$')
colnames(malaria.table) <- c('Mean',
                             '2.5% Quant',
                             '97.5% Quant')

kable(malaria.table, digits = 3,
      caption='\\label{tab:malaria}Quantile estimates on the natural scale for parameter effects on odds ratio for logistic spatial

## marginal plots

# phi
malaria.phi.posterior.transform <- inla.tmarginal(function(x) x/1000, malaria.fit$parameters$range$posterior[,c('x','y')])
malaria.phi.prior.transform <- inla.tmarginal(function(x) x/1000, malaria.fit$parameters$range$posterior[,c('x','prior')])
malaria.phi.df <- as.data.frame(cbind(x=c(malaria.phi.prior.transform[,'x'], malaria.phi.posterior.transform[, 'x']),
                                      type=rep(c(1,2),each=nrow(malaria.phi.posterior.transform)),
                                      y=c(malaria.phi.prior.transform[,'y'], malaria.phi.posterior.transform[, 'y'])))
malaria.phi.df$type <- factor(malaria.phi.df$type, labels=c('Prior', 'Posterior'))

malaria.phi.plot <- ggplot(malaria.phi.df, aes(x=x, y=y)) +
  geom_line(aes(color=type), size=2) +
  scale_x_continuous(limits = c(0,100), expand=c(0.005,0.005)) +
  scale_y_continuous(limits = c(0,0.05), expand=c(0,0)) +
  xlab('x') + ylab('Density') +
```

```
  ggtitle(TeX('Marginal Densities for $\\phi / 1000$')) +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(legend.position = 'none')

# sigma
malaria.sigma.posterior <- malaria.fit$parameters$sd$posterior
malaria.sigma.df <- as.data.frame(cbind(x=rep(malaria.sigma.posterior[,'x'],2),
                                   type=rep(c(1,2),each=nrow(malaria.sigma.posterior)),
                                   y=c(malaria.sigma.posterior[,'prior'], malaria.sigma.posterior[,'y'])))
malaria.sigma.df$type <- factor(malaria.sigma.df$type, labels=c('Prior', 'Posterior'))

malaria.sigma.plot <- ggplot(malaria.sigma.df, aes(x=x, y=y)) +
  geom_line(aes(color=type), size=2) +
  scale_x_continuous(limits = c(0.5,1.5), expand=c(0.005,0.005)) +
  scale_y_continuous(limits = c(0,2.5), expand=c(0,0)) +
  xlab('x') + ylab('Density') +
  ggtitle(TeX('Marginal Densities for $\\sigma$')) +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(legend.position = 'none')

grid.arrange(malaria.phi.plot, malaria.sigma.plot, ncol=2)

### colour sequence for probabilities
predict.cols <- colourScale(seq(0,1,by=0.05),
                            style = "equal",
                            breaks = 10,
                            dec = -log10(0.02),
                            col = "RdYlGn",
                            rev = TRUE,
                            opacity = 1)
predict.cols$breaks <- round(predict.cols$breaks, 2)
predict.cols$breaks[10] <- 1

### expected mean
map.new(malaria.fit$raster, legendRight = 0.9)
plot(malaria.fit$raster[["predict.invlogit"]], add = TRUE, col = predict.cols$colOpacity,
     breaks = predict.cols$breaks, legend = FALSE)
plot(gmap2, add = TRUE, maxpixels = 10^6)
legendBreaks("right", predict.cols, outer = TRUE, bty = "n", inset = 0)

## predicted 2.5% quantile
predict025.raster <- exp(malaria.fit$raster[["predict.0.025quant"]]) / (1 + exp(malaria.fit$raster[["predict.0.025quant"]]))
map.new(malaria.fit$raster, legendRight = 0.9)
plot(predict025.raster, add = TRUE, col = predict.cols$colOpacity, breaks = predict.cols$breaks, legend = FALSE)
plot(gmap2, add = TRUE, maxpixels = 10^6)
legendBreaks("right", predict.cols, outer = TRUE, bty = "n", inset = 0)

## predicted 97.5% quantile
predict975.raster <- exp(malaria.fit$raster[["predict.0.975quant"]]) / (1 + exp(malaria.fit$raster[["predict.0.975quant"]]))
map.new(malaria.fit$raster, legendRight = 0.9)
plot(predict975.raster, add = TRUE, col = predict.cols$colOpacity, breaks = predict.cols$breaks, legend = FALSE)
plot(gmap2, add = TRUE, maxpixels = 10^6)
legendBreaks("right", predict.cols, outer = TRUE, bty = "n", inset = 0)
```