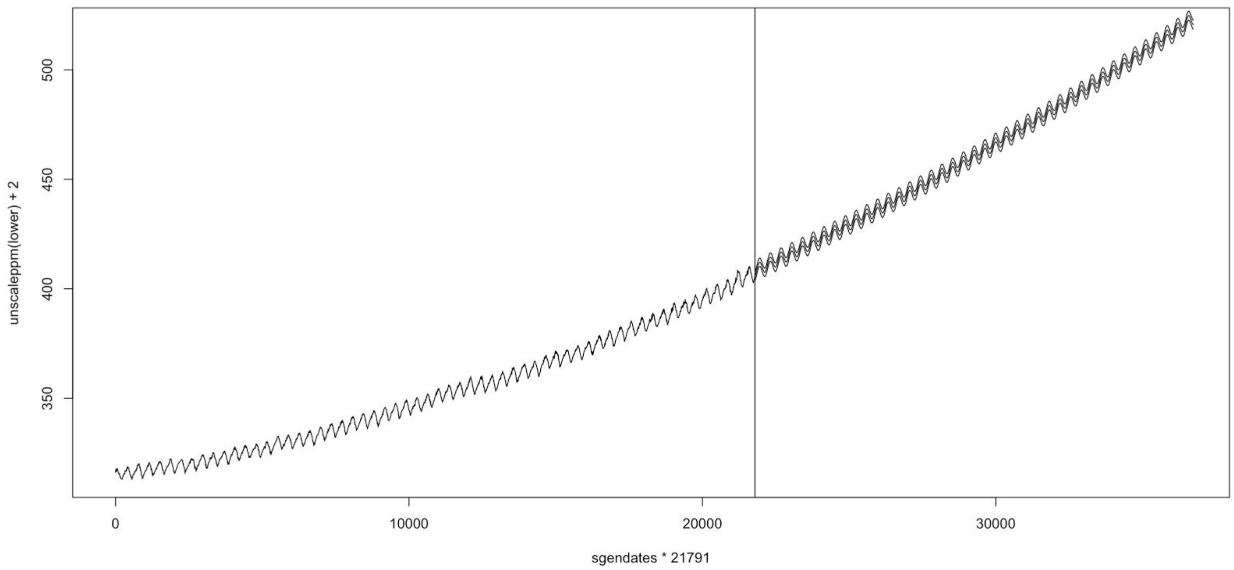
```
library("rstan")
# th BRRR package allows me to play rap ad libs to notify me when my stan code
is finished running
library("BRRR")
library("ggplot2")
setwd("/Users/jakeschaeffer/Desktop/School/
      Junior Year/CS146/Final Project")

df <- read.csv("co2_mlo.csv")

rescale <- function(x){
  # Rescales data vector 0 < data < 1
  (x - min(x)) / (max(x) - min(x))
}

# will need this later to plot my predictions
unscaleppm <- function(x){
  x * (410.18 - 313.04) + 313.04
}

sDate <- rescale(df$day_int)
sCO2 <- rescale(df$co2)

# the last day integer is 21791
# the day integer forty years from now will be 36391
# the length of gendates is the number of datapoints to predict for
# sgendates will let me plot the predicted values directly
gendates <- seq(21791,36391,7)
sgendates <- seq(1,1.685508,length.out = 2086)

# I tried running the model with and without scaled data (Hint: it did better #
with scaled data)
data <- list(N = length(df$day_int),ppm = df$co2,t = df$day_int,
             N_future = length(gendates),t_future = gendates)

sdata <- list(N = length(sDate),ppm = sCO2,t = sDate,
             N_future = length(sgendates),t_future = sgendates)


model <- "
data {
int<lower=0> N; // length dates, scaled
real ppm[N]; // co2 levels in ppm, scaled
real t[N]; // timesteps
int<lower=0> N_future;
real t_future[N_future];
}
```

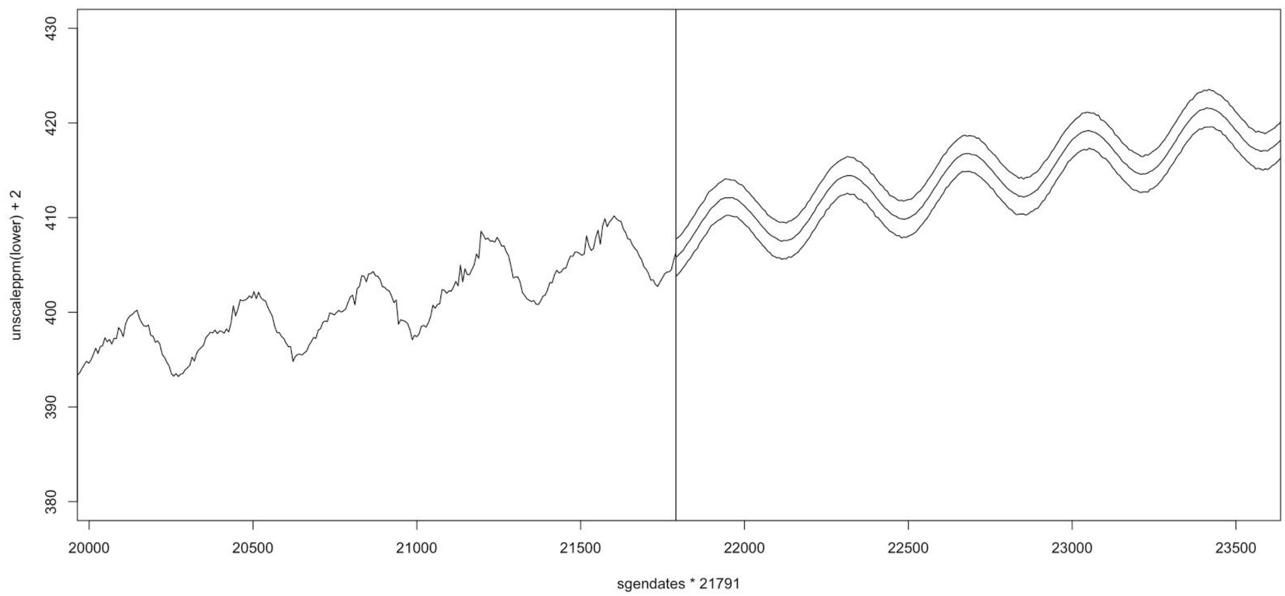
```
parameters {
real<lower=0,upper=1> c0;
real c1;
real c2;
real c3;
real c4;
real cs2;
}

transformed parameters {
real<lower=0> c0_transf;
real<lower=0> c1_transf;
real<lower=0> cs2_transf;
real<lower=0> c2_transf;
real<lower=0> c3_transf;
real<lower=0> c4_transf;

c0_transf = exp(c0);
c1_transf = exp(c1);
cs2_transf = exp(cs2);
c2_transf = exp(c2);
c3_transf = exp(c3);
c4_transf = exp(c4);
}

model {
c0 ~ normal(0,1); // after normalization, y-intercept will be near 0
c1 ~ normal(1,2); // after normalization, the slope over time is ~1
c2 ~ normal(1,10);
cs2 ~ normal(1,3);
c3 ~ normal(1,3); // should be periodic on [0,2pi], could use von mises dist
c4 ~ normal(1,2);
for(n in 1:N) {
  ppm[n] ~ normal(c0 + c1*t[n] + cs2*t[n]*t[n] + c2*cos(2*3.14*t[n]/0.01676151
+ c3),c4*c4);
}
}

generated quantities {
real ppm_future[N_future]; // future co2 levels

for(n in 1:N_future) {
  ppm_future[n] = normal_rng(c0 + c1*t_future[n] + cs2*t_future[n]*t_future[n]
+ c2*cos(2*3.14*t_future[n]/0.01676151 + c3), c4*c4);
}
}
"
```

```
fit <- stan(
  model_code = model,
  data = sdata,
  chains = 2,             # number of Markov chains
  warmup = 1000,          # number of warmup iterations per chain
  iter = 2000,            # total number of iterations per chain
  cores = 3,              # number of cores (using 2 just for the vignette)
  refresh = 1000,         # show progress every 'refresh' iterations
  control = list(adapt_delta = 0.99)
)

# plays me a nice notification noise letting me know stan is done
skrrrahh("traviscott")

print(fit, probs=c(.05, 0.95))
######################################
#                  5%   95% n_eff Rhat
# c0             0.01  0.02   526 1.01
# c1             0.48  0.48   532 1.00
# c2             0.03  0.03  2000 1.00
# c3            -0.33 -0.30   723 1.00
# c4            -0.10 -0.10  1731 1.00
# cs2            0.46  0.47   582 1.00
######################################

samples <- extract(fit)
results <- apply(samples$ppm_future, 2, quantile, probs = c(0.025, 0.5, 0.975))
pred <- samples$ppm_future

n <- length(results)
lower <- results[seq(1, n, 3)]
median <- results[seq(2, n, 3)]
upper <- results[seq(3, n, 3)]
intervals <- cbind(lower,median,upper)
intervalsdf <- data.frame(intervals)

# Prediction of CO2 level 40 years from now
ppmpred <- unscaleppm(results[,2086])
# Intervals |      2.5%      50%    97.5%
# ppm lvls  |  516.6028 518.6575 520.6636

# Generates Scaled Graph of Data and Predictions
plot(sgendates,lower,xlim=c(0,1.7),ylim=c(0,2),type="l")
lines(sgendates,lower)
lines(sgendates,upper)
lines(sgendates,median)
lines(sDate,sCO2)
```

```
# Generates Graph of Data and Predictions until 2058
plot(sgendates*21791,unscaleppm(lower),xlim=c(0,36500),ylim=c(313,520),type="l"
)
lines(sgendates*21791,unscaleppm(upper))
lines(sgendates*21791,unscaleppm(median))
lines(df$day_int,df$co2)
abline(v=21791)
```




```
# Generates Close-Up at the Cutoff
plot(sgendates*21791,unscaleppm(lower),xlim=c(20100,23500),ylim=c(380,430),type
="l")
lines(sgendates*21791,unscaleppm(upper))
lines(sgendates*21791,unscaleppm(median))
lines(df$day_int,df$co2)
abline(v=21791)
```




```
# Takes a close-up look at where the median passes 450 ppm for the first time
plot(sgendates*21791,unscaleppm(median),xlim=c(27500,28000),ylim=c(450,452))
abline(v=27781))

# By day 27781 after March 29, 1958, or Thursday, April 20, 2034, there is a
# 50% chance we will hit 450 ppm for the first time.


# Takes a close-up look at where there is over 95% prediction that
# CO2 levels will be above 450 ppm
plot(sgendates*21791,unscaleppm(lower),xlim=c(28000,32000),ylim=c(450,453))
abline(v=28150)

# By day 28150 after March 29, 1958, or Tuesday, April 24, 2035, there
# is a 97.5% prediction that CO2 levels will be at or above 450 ppm
```