Assignment 7: Growth

Bri Baker, Ian Brunjes, & Scout Leonard

05/17/2022

Consider the following model of forest growth (where forest size in measured in units of carbon (C)):

- dC/dt = r*C for forests where C is below a threshold canopy closure
- dC/dt = g*(1- C/K) for forests where carbon is at or above the threshold canopy closure
- K = carrying capacity in units of carbon

The size of the forest (C), canopy closure threshold (thresh) and carrying capacity (K) are all in units of kg carbon.

The canopy closure threshold represents the size of the forest at which growth rates change from exponential to linear.

r represents early exponential growth rate and g the linear growth rate once canopy closure has been reached.

1 Part 1: Create Model

```
#source function
source(here("R", "dgrowth.R"))

#view function:
dgrowth
```

```
## function (Time, C, parms)
## {
##     if (C < parms$thresh) {
##         rate = parms$r * C
##     }
##     else {
##         rate = parms$g * (1 - (C/parms$K))
##     }
##     return(list(rate))
## }
```

2 Part 2: Run Model

We ran the model for 300 years (using the ODE solver) starting with an initial forest size of 10 kgC, and using the following parameters:

- canopy closure threshold of 50 kgC
- K = 250 kgC (carrying capacity)
- r= 0.01 (exponential growth rate before before canopy closure)
- g = 2 kg/year (linear growth rate after canopy closure)

```
# model run time: 300 years
tm = seq(1, 300, 1)

# initial kilograms of Carbon
Cinit = 10 # initial kilograms of Carbon

#growth parameters:
gps = list(

  K = 250, # carrying capacity kg of Carbon

  r = 0.01, # exponential rate of growth before canopy closure

  g = 2, # linear growth after canopy closure kilograms/year

  thresh = 50 # canopy closure threshold (kilograms of carbon)

)

#results
res = ode(Cinit, tm, dgrowth, parms = gps)

#rename results columns
colnames(res) = c("time", "carbon")
```

```
ggplot(as.data.frame(res), aes(time, carbon)) +
  geom_point() +
  theme_minimal() +
  labs(title = "Forest Growth Model: 300 Years",
       x = "Time (years)",
       y = "Carbon (kilograms)") +
  theme(plot.title = element_text(hjust = 0.5))
```

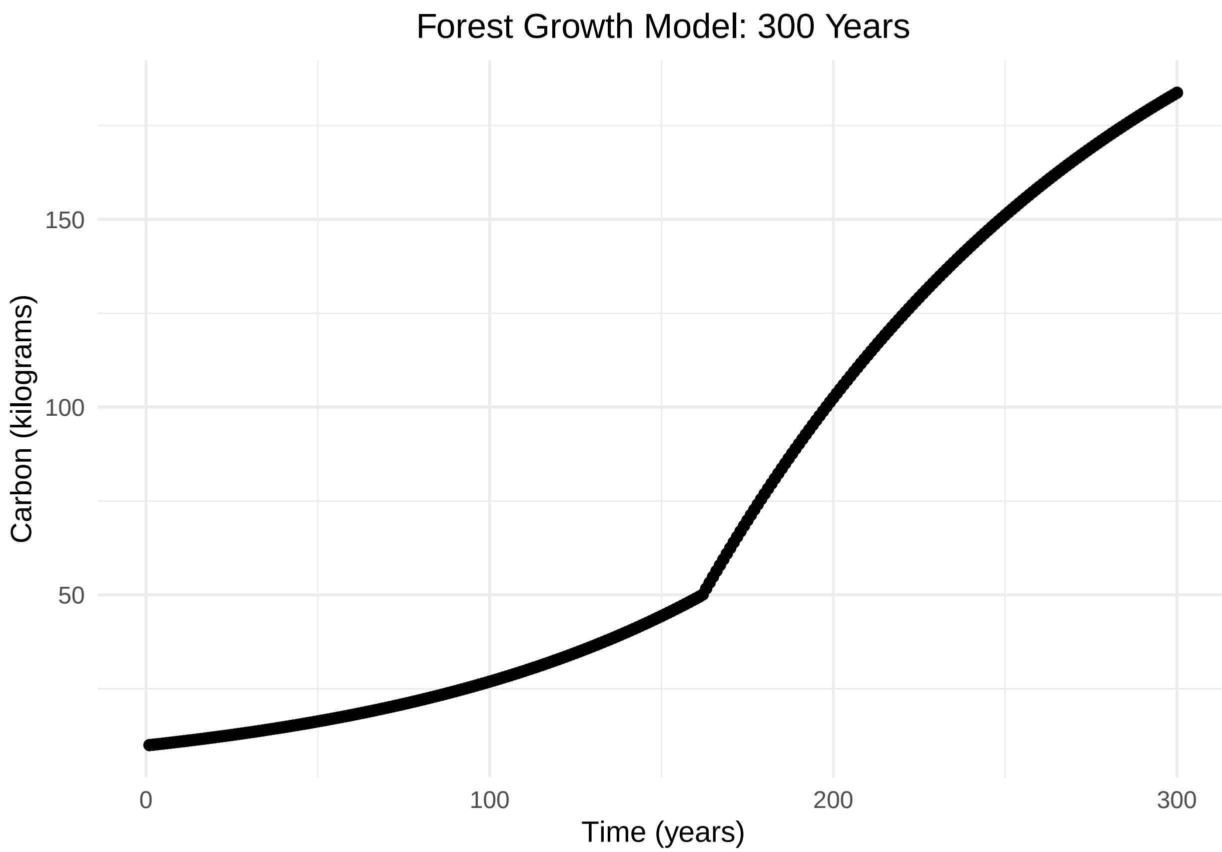


Figure 1: Growth model over 300 years using ODE Solver, starting with initial forest size of 10 kilograms of carbon and the following parameters: 250 kilograms of carbon carrying capacity, 0.01 exponential growth rate before canopy closure, and 2 kilograms per year growth rate after canopy closure.

3 Part 3: Sobol Sensitivity Analysis

We utilized Sobol sensitivity analysis to explore how the estimated maximum and mean forest size varies with the four input parameters: pre canopy closure growth rate (r), post-canopy closure growth rate (g) and canopy closure threshold (thresh) and carrying capacity(K).

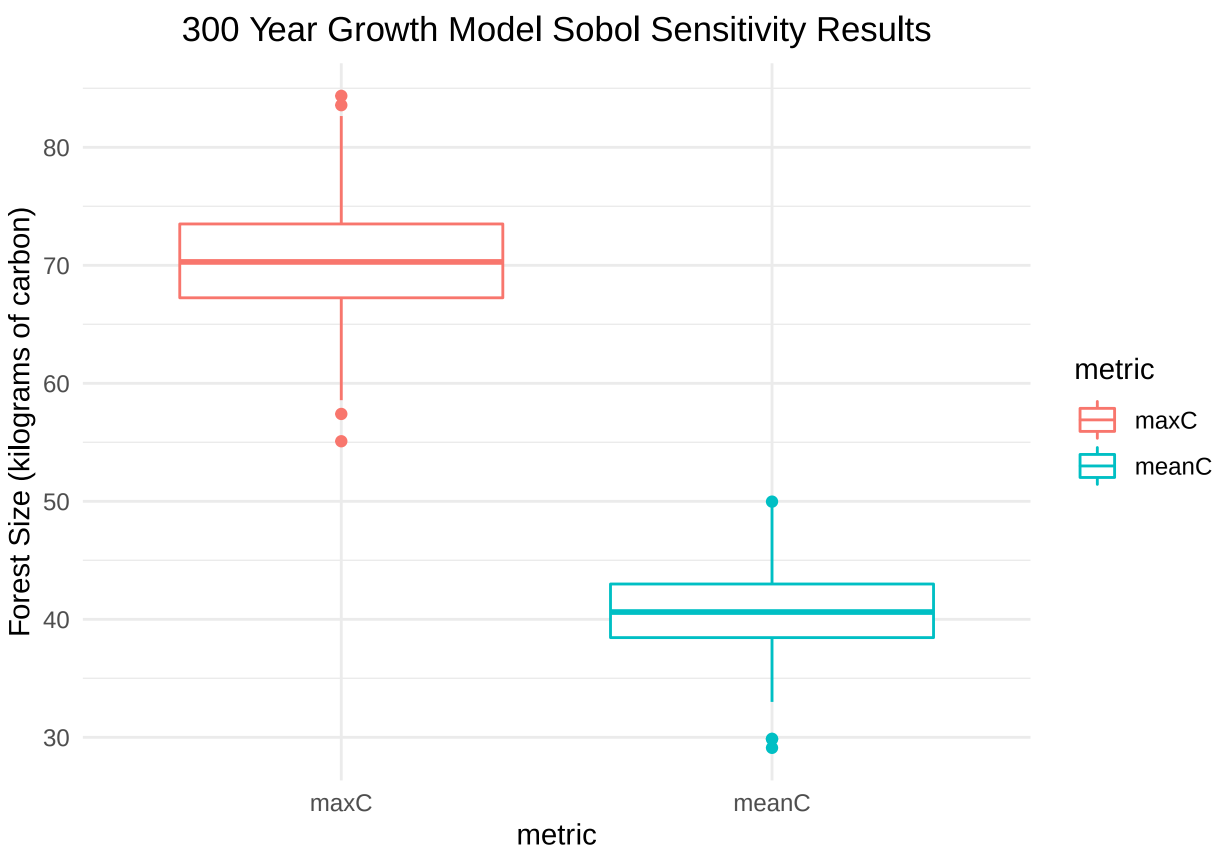
```
# Mean parameter values
u_K = 250
u_thresh = 50
u_r = 0.01
u_g = 0.2

# Standard deviation factor
sdf = 0.1

# Build two sample parameter sets
Cinit = 10
np = 100

#sample parmeter set 1:
X1 = cbind.data.frame(

  K = rnorm(mean = u_K, sd = u_K * sdf, n = np),

  thresh = rnorm(mean = u_thresh, sd = u_thresh * sdf, n = np),

  r = rnorm(mean = u_r, sd = u_r * sdf, n = np),

  g = rnorm(mean = u_g, sd = u_g * sdf, n = np)

)

#sample parameter set 2:
X2 = cbind.data.frame(

  K = rnorm(mean = u_K, sd = u_K * sdf, n = np),

  thresh = rnorm(mean = u_thresh, sd = u_thresh * sdf, n = np),

  r = rnorm(mean = u_r, sd = u_r * sdf, n = np),

  g = rnorm(mean = u_g, sd = u_g * sdf, n = np)

)

# Create sobol object with parameter sets
sens_carbon = sobolSalt(model = NULL,
                        X1,
                        X2,
                        nboot = 300)

#rename columns of sobol object
colnames(sens_carbon$X) = c("K",
                            "thresh",
```

```
                             "r",
                             "g")

parms = sens_carbon$X
```

```
# Run our diffEQ model, extracting max and mean forest carbon

# Function to compute relevant metrics
compute_metrics = function(result) {

  maxC = max(result$carbon)
  meanC = mean(result$carbon)

  return(list(maxC = maxC, meanC = meanC))

}

# Wrapper function for running solver and getting metrics for all params
C_wrapper = function(K, thresh, r, g, Cinit, simtimes, func) {

  gps = list(
    K = K,
    r = r,
    g = g,
    thresh = thresh
  )

  result = ode(Cinit, simtimes, func, gps, method = "daspk")
  result = as.data.frame(result)
  colnames(result) = c("time","carbon")

  metrics = compute_metrics(result)
  return(metrics)
}

# Use pmap with wrapper function to execute against sobol parameters
allresults = as.data.frame(parms) %>%
  pmap(C_wrapper,
       Cinit = Cinit,
       simtimes = tm,
       func = dgrowth)

# extract out results from pmap into a data frame
allres = allresults %>% map_dfr(`[`,c("maxC","meanC"))
```

```
# plot results
tmp = allres %>%
  gather(key = "metric",
         value = "value")

ggplot(tmp, aes(metric, value, col = metric)) +
  geom_boxplot() +
  theme_minimal() +
```

```r
  labs(y = "Forest Size (kilograms of carbon)",
       title = "300 Year Growth Model Sobol Sensitivity Results") +
  theme(plot.title = element_text(hjust = 0.5))
```



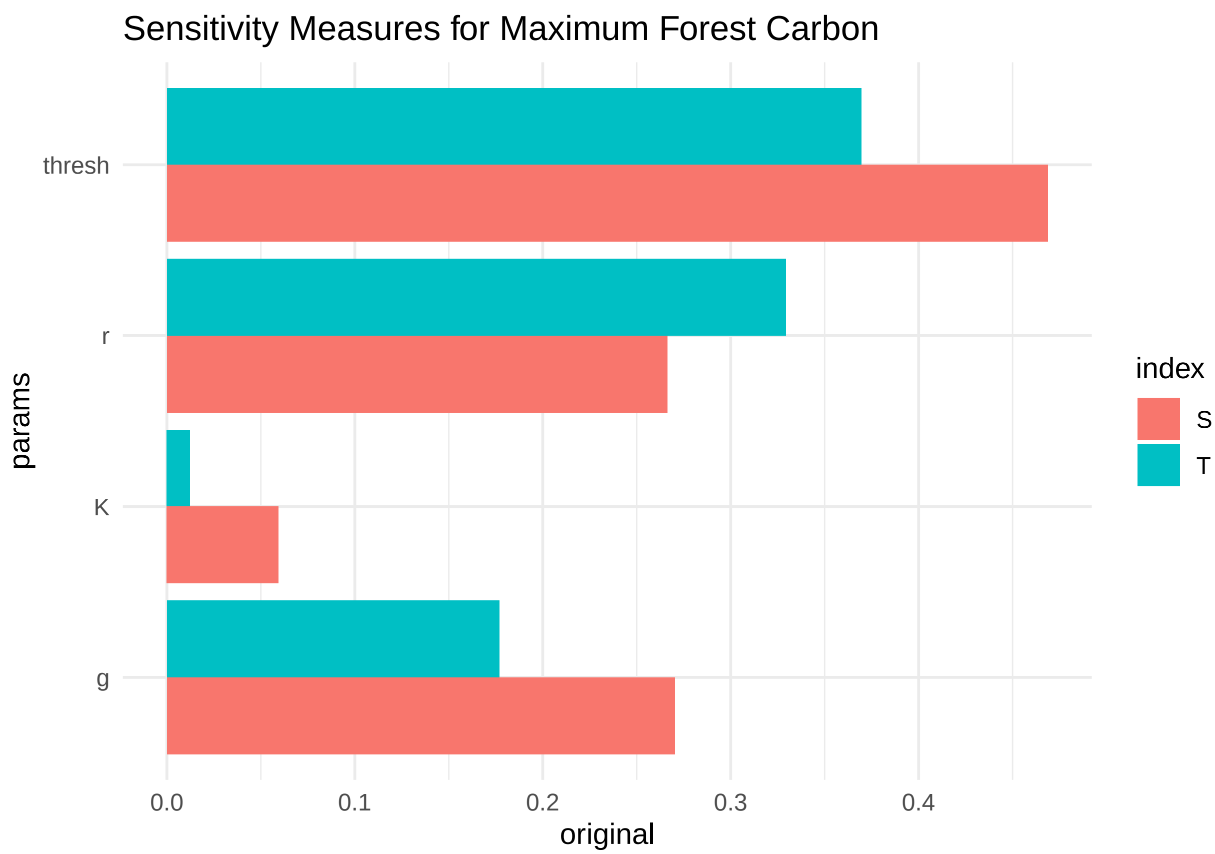
Finally, we computed the first order and total sensitivity Sobol indices for both max and mean forest carbon metrics.

```r
# max carbon
sens_carbon_max = sensitivity::tell(sens_carbon, allres$maxC)

#first order indices (main effect without co-variance)
row.names(sens_carbon_max$S) = colnames(parms)

#total sensitivity index
row.names(sens_carbon_max$T) = colnames(parms)

print(sens_carbon_max)
```

```
##
## Call:
## sobolSalt(model = NULL, X1 = X1, X2 = X2, nboot = 300)
##
## Model runs: 600
##
## Model variance: 21.13394
```

```
##
## First order indices:
##          original         bias std. error   min. c.i. max. c.i.
## K      0.05939035 -0.005996932 0.09081438 -0.09786750 0.2380396
## thresh 0.46872443 -0.008025133 0.06752136  0.35382086 0.6211217
## r      0.26628075 -0.004404401 0.10172915  0.08908401 0.4792279
## g      0.27023818 -0.008705965 0.09053108  0.10824471 0.4713066
##
## Total indices:
##          original         bias  std. error   min. c.i.  max. c.i.
## K      0.01230315 0.0001832158 0.002270695 0.006932932 0.01608105
## thresh 0.36941314 0.0044576406 0.068171685 0.222644250 0.48554050
## r      0.32936554 0.0040171611 0.061036512 0.183042485 0.42911183
## g      0.17678763 0.0040795148 0.033675842 0.090710901 0.23255188
```

```r
#add index column and param column to T df
sens_carbon_max$T$index <- 'T'

sens_carbon_max_T <- as.data.frame(sens_carbon_max$T) %>%
  rownames_to_column("params") %>%
  select("params", "original", "index")

#add index column and param column to S df
sens_carbon_max$S$index <- 'S'

sens_carbon_max_S <- as.data.frame(sens_carbon_max$S)  %>%
  rownames_to_column("params") %>%
  select("params", "original", "index")

index_combined <- rbind(sens_carbon_max_T, sens_carbon_max_S)

ggplot(data = index_combined, aes(x = params, y = original, fill = index)) +
  geom_bar(stat = "identity", position = "dodge") +
  theme_minimal() +
  coord_flip() +
  labs(title = "Sensitivity Measures for Maximum Forest Carbon")
```

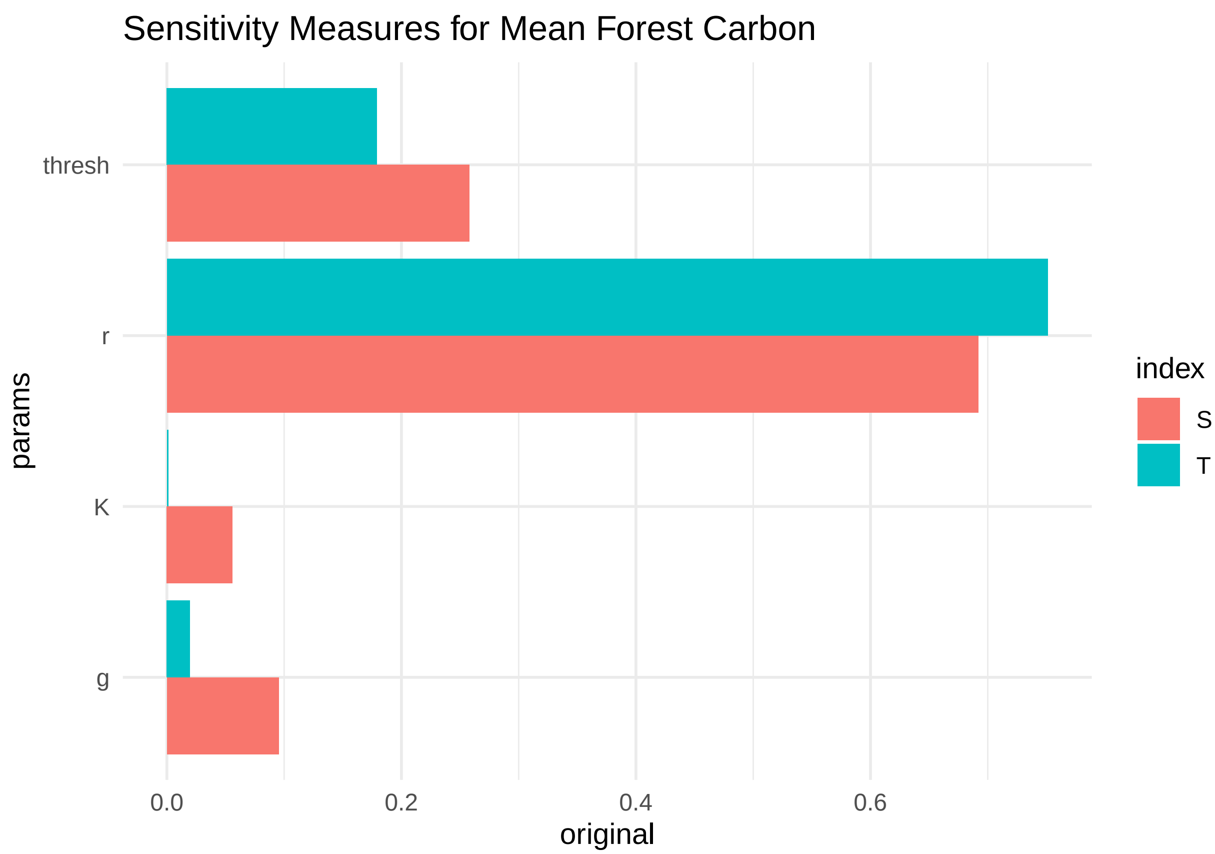



```
#mean carbon
sens_carbon_mean = sensitivity::tell(sens_carbon, allres$meanC)

#first order indices (main effect without co-variance)
row.names(sens_carbon_mean$S) = colnames(parms)

#total sensitivity index
row.names(sens_carbon_mean$T) = colnames(parms)

print(sens_carbon_mean)
```

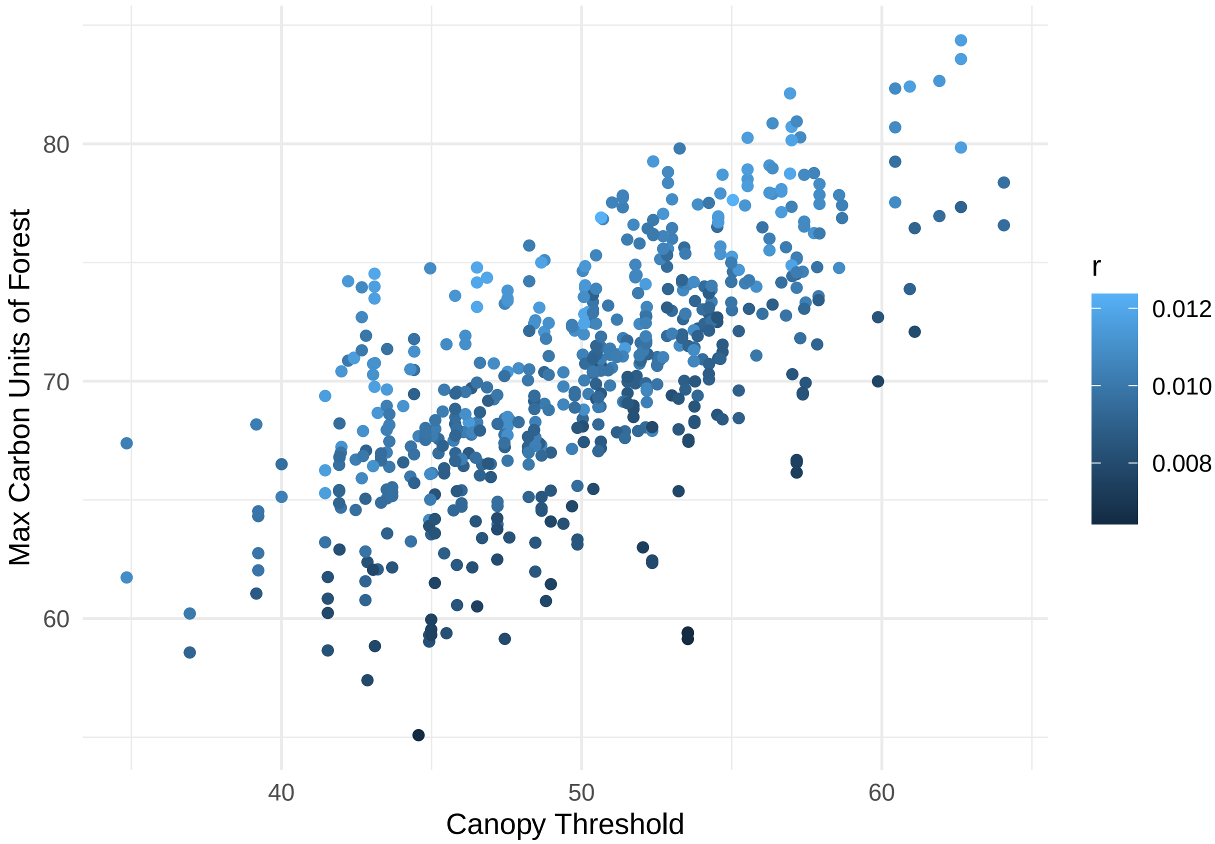
```
##
## Call:
## sobolSalt(model = NULL, X1 = X1, X2 = X2, nboot = 300)
##
## Model runs: 600
##
## Model variance: 8.544345
##
## First order indices:
##          original         bias std. error   min. c.i. max. c.i.
## K      0.05589482 -0.009160138 0.08162132 -0.08985897 0.2230460
## thresh 0.25789049 -0.007260057 0.08770691  0.07907397 0.4448616
## r      0.69182405 -0.011639203 0.06262366  0.59615731 0.8325971
## g      0.09531587 -0.009230233 0.08105697 -0.04632774 0.2535803
##
```

```
## Total indices:
##           original         bias   std. error    min. c.i.   max. c.i.
## K      0.001252729 5.988548e-05 0.0002866173 0.0005062301 0.001675042
## thresh 0.179085197 9.395082e-03 0.0378143878 0.0824385718 0.237096451
## r      0.751165220 6.413043e-03 0.0953886624 0.5643954093 0.953992858
## g      0.019696040 1.035812e-03 0.0054034968 0.0072000049 0.027221627
```

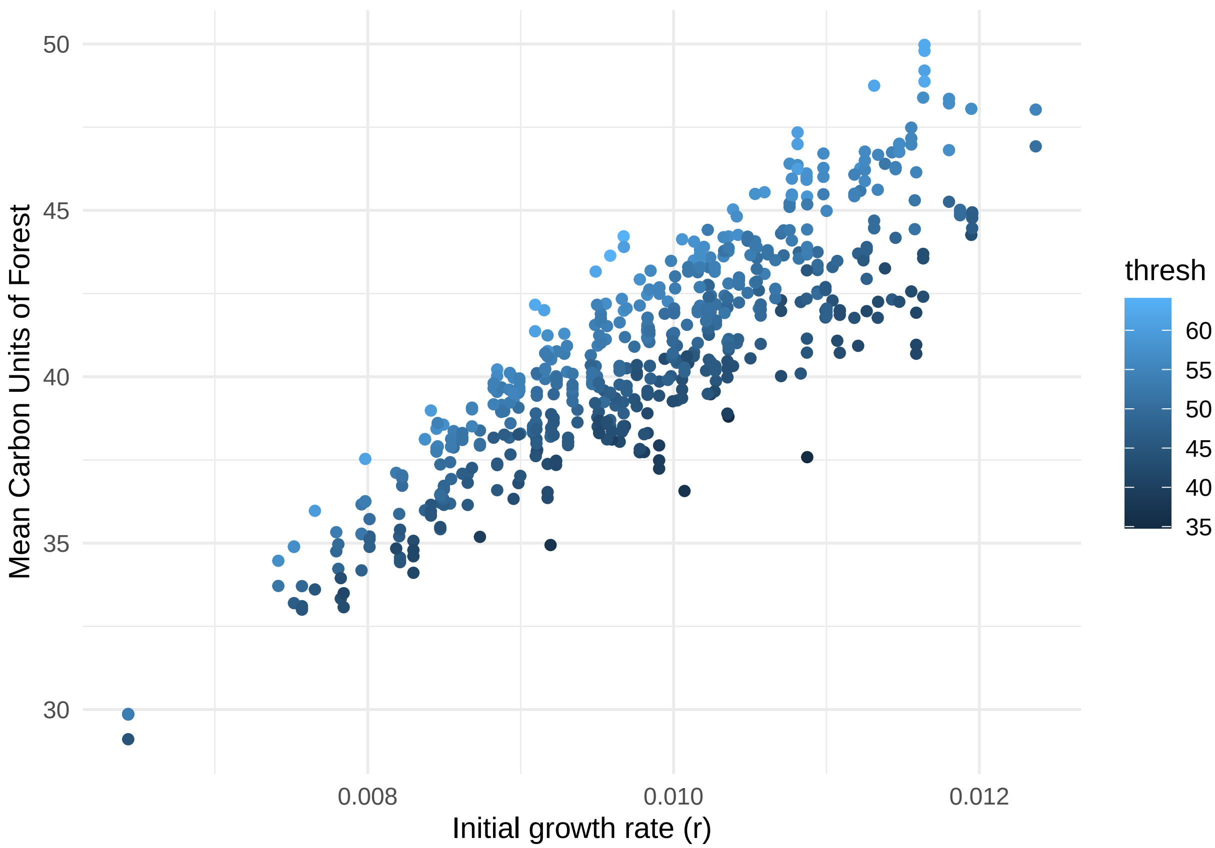
```r
#add index column and param column to T df
sens_carbon_mean$T$index <- 'T'

sens_carbon_mean_T <- as.data.frame(sens_carbon_mean$T) %>%
  rownames_to_column("params") %>%
  select("params", "original", "index")

#add index column and param column to S df
sens_carbon_mean$S$index <- 'S'

sens_carbon_mean_S <- as.data.frame(sens_carbon_mean$S)  %>%
  rownames_to_column("params") %>%
  select("params", "original", "index")

index_combined <- rbind(sens_carbon_mean_T, sens_carbon_mean_S)

ggplot(data = index_combined, aes(x = params, y = original, fill = index)) +
  geom_bar(stat = "identity", position = "dodge") +
  theme_minimal() +
  coord_flip() +
  labs(title = "Sensitivity Measures for Mean Forest Carbon")
```




```r
# More plotting
# Make data frame for plotting
results_df = cbind.data.frame(

  parms,
  mean = sens_carbon_mean$y,
  max = sens_carbon_max$y

)

# For max values, threshold then r have most impact
ggplot(results_df, aes(x = thresh, y = max, col = r)) +
  geom_point() +
  labs(y = "Max Carbon Units of Forest",
       x = "Canopy Threshold") +
  theme_minimal()
```




```
# For mean values, r then threshold have most impact
ggplot(results_df, aes(x = r, y = mean, col = thresh)) +
  geom_point() +
  labs(y = "Mean Carbon Units of Forest",
       x = "Initial growth rate (r)") +
  theme_minimal()
```



4 Climate and Forest Carbon

While some of the factors of forest growth are dependent more on the biotic properties of the forest (threshold and linear growth rate in particular), climate is still likely to influence forest growth dynamics. The parameters that climate is likely to have stronger effects on are early exponential growth rate (r) and carrying capacity (K), although the particular forcings on these parameters will be based on local climate (different places will have different changes in precipitation and temperature which limit water availability). Higher atmospheric carbon will have a positive effect on r globally.

Our model is particularly sensitive to changes in the r parameter, so climate change is likely to have strong effects on forest growth.