Chapter 7 Homework

Emily Maloney

February 18, 2019

Chapter 7 Homework

```r
library(rethinking)
library(tidyverse)
```

Easy Problems

7E1

1. A hypothetical third variable that would lead to an interaction effect for the relationship "bread dough rises because of yeast" is the temperature of the oven.

2. A hypothetical third variable that would lead to an interaction effect for the relationship "education leads to a higher income" is gender.

3. A hypothetical third variable that would lead to an interaction effect for the relationship "gasoline makes a car go" is the type of car.

7E2

The only explanation to invoke an interaction is:
1. Caramelizing onions requires cooking over low heat and making sure the onions do not dry out.

7E3

Linear models to express the stated relationships in 7E2 are: 1. $caramelizing\ i = \alpha + \beta 1(Heat)\ x\ i + \beta 2(Dry)\ x\ i + \beta 3(Heat * Dry)\ x\ i$ 2. $speed\ i = \alpha + \beta 1(cyl)\ x\ i + \beta 2(fuel)\ x\ i$
3. $beliefs\ i = \alpha + \beta 1(parent)\ x\ i + \beta 2(friend)\ x\ i$
4. $intelligence\ i = \alpha + \beta 1(social)\ x\ i + \beta 2(manip)\ x\ i$

Medium Problems

7M1

The finding that in hot temperatures, none of the plants bloomed, regardless of water and shade levels indicates that there are negative interactions between water and temperature, shade and temperature, and temperature and water and shade all together. In this way, the interactions with temperature are supressors, in that if temperature is high, then the shade and water have no effect on whether or not the plants bloom.

7M2

For simplicity's sake, the variable for temperature will take the value of 1 when the temperature is hot and 0 when it is cold in the following regression equation:

bloom size = 5 + 2(water) + -1(shade) + -2.5(water x shade) + -5(temp) + -2(water x temp) + 1(shade x temp) + 2.5(water x shade x temp)

Because every element in the regression equation before not including temperature ($\alpha + \beta 1(water) + \beta 2(shade)$ $+ \beta 3(waterxshade)$)) is counteracted by a negative version in the new terms including temperature ($\beta 4(temp)$ $+ \beta 5(waterxtemp) + \beta 6(shadextemp) + \beta 7(shadextempxwater)$), then when temperature is hot, the bloom size will equal 0. alpha = -B4(temp) B1(water) = -B5(water x temp) B2(shade) = -B6(shade x temp) B3(water x shade) = -B7(water x shade x temp)

7M3

```
rho1 <- 0.3 #corr b/w food and popsize for no wolves
rho2 <- 0.7 #corr b/w food and popsize with wolves

rho3 <- 0.2 #corr b/w wolves and popsize

#get food simulations
food1 <- rnorm(50)
food2 <- rnorm(50)
food <- as.data.frame(cbind(food1, food2))
food <- gather(food, key = "time", value = "food", food1:food2)

#get popsize simulations
popsize1 <- rnorm(50, rho1*food1, sqrt(1- rho2^2))
popsize2 <- rnorm(50, rho2*food2, sqrt(1 - rho2^2))
popsize <- as.data.frame(cbind(popsize1, popsize2))
popsize <- gather(popsize, key = time, value = "popsize", popsize1:popsize2)

#wolves present?
wolves1 <- c(rep(0, 50))
wolves2 <- rnorm(50, rho3*popsize2, sqrt(1 - rho3^2))
wolves <- as.data.frame(cbind(wolves1, wolves2))
wolves <- gather(wolves, key = time, value = "wolves", wolves1:wolves2)

d <- data.frame(popsize, food, wolves)

h7.1 <- rethinking::map(
            alist(
            popsize ~ dnorm(mu, sigma),
            mu <- a + bf*food + bw*wolves,
            a ~ dnorm(0, 1),
            bf ~ dnorm(0, 1),
            bw ~ dnorm(0, 1),
            sigma ~ dunif(0, 10)
            ),
            data = d)

h7.2 <- rethinking::map(
              alist(
                popsize ~ dnorm(mu, sigma),
                mu <- a + gamma*food + bw*wolves,
                gamma <- bf + bfw*wolves,
                a ~ dnorm(0, 1),
                bf ~ dnorm(0, 1),
                bw ~ dnorm(0, 1),
                bfw ~ dnorm(0, 1),
                sigma ~ dunif(0, 10)
              ),
              data = d)

coeftab(h7.1, h7.2)
```

```
##       h7.1    h7.2
## a        0.00    0.01
## bf       0.48    0.47
## bw       0.20    0.19
## sigma    0.65    0.65
## bfw        NA   -0.07
## nobs      100     100
```

```
compare(h7.1, h7.2)
```

```
##       WAIC pWAIC dWAIC weight    SE  dSE
## h7.1 207.5   4.1   0.0    0.6 15.29   NA
## h7.2 208.3   4.8   0.8    0.4 15.99 1.81
```

In the simulated data set, the relationship between wolves, food, and population size of ravens is more accurately conveyed using a model with a statistical interaction. The model including the interaction term (h7.1) has a lower WAIC value than the model without, and gets a majority of the weight when compared to the model without an interaction between the variables containing information about the presence of wolves and food. In reality, I doubt that the biological interaction would be linear because I imagine there is some threshold effect of population size, such that even when there are many wolves helping provide food for ravens, the population size will stop increasing at a certain point. Also, it may be the case that the difference in population size between no wolves and a few wolves is much larger than the difference in population size for a few wolves and many wolves, because population growth tends to be exponential. The initial increase in food via the wolves may have a very large effect on population growth, while later increase in food with more wolves may have a lesser effect on population growth.

Hard Problems

7H1

```
#load data
data(tulips)
d <- tulips

#create dummy variables for bed and center shade and water
d <- d %>% mutate(bed_b = ifelse(bed == "b", 1, 0),
                  bed_c = ifelse(bed == "c", 1, 0),
                  water.c = water - mean(water),
                  shade.c = shade - mean(water))

h3 <- rethinking::map(
      alist(
        blooms ~ dnorm(mu, sigma),
        mu <- a + bB*bed_b + bC*bed_c + bW*water.c + bS*shade.c + bWS*water.c*shade.c,
        a ~ dnorm(130, 100),
        bB ~ dnorm(0, 100),
        bC ~ dnorm(0, 100),
        bW ~ dnorm(0, 100),
        bS ~ dnorm(0, 100),
        bWS ~ dnorm(0, 100),
        sigma ~ dunif(0, 100)
              ),
              data = d,
              start = list(a = mean(d$blooms), bW = 0, bS = 0, bWS = 0,
                           bB = mean(d$bed_b),
                           bC = mean(d$bed_c),
                           sigma = sd(d$blooms))
      )

precis(h3)
```

```
##         Mean StdDev   5.5%  94.5%
## a      99.33  12.76  78.94 119.72
## bW     75.15   9.20  60.44  89.85
## bS    -41.24   9.20 -55.94 -26.54
## bWS   -52.25  11.24 -70.22 -34.29
## bB     42.45  18.04  13.62  71.28
## bC     47.06  18.04  18.23  75.89
## sigma  39.19   5.34  30.66  47.72
```

The results from the model with bed as a dummy variable indicate that the intercept for bed A plants is 99.3294993, for plants in bed B is 141.781247, and for plants in bed C is 146.3917524. This suggests that there is a relatively big difference in the expected bloom size for plants in bed A versus bed B and C, but not as much difference between the latter two themselves.

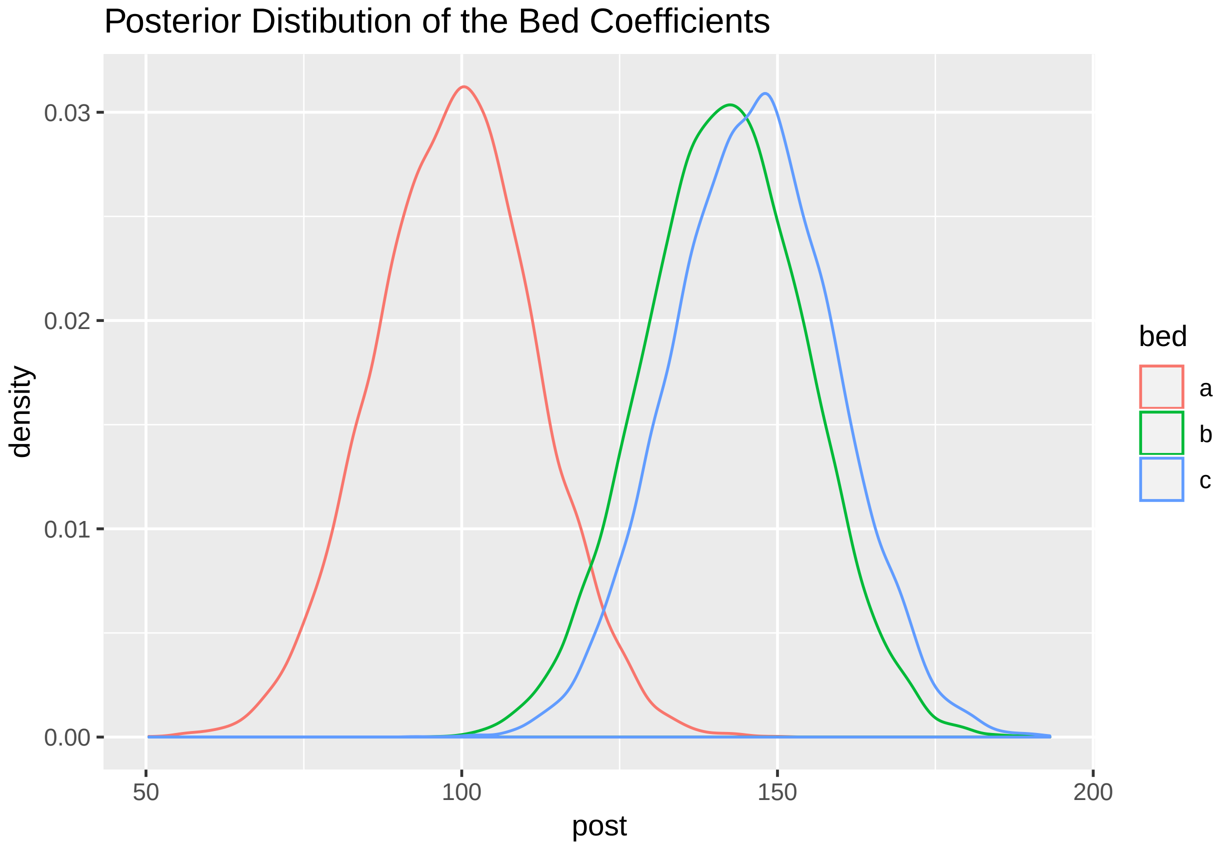
7H2

```
h4 <- rethinking::map(
              alist(
                blooms ~ dnorm(mu, sigma),
                mu <- a + bW*water.c + bS*shade.c + bWS*water.c*shade.c,
                a ~ dnorm(130, 100),
                bW ~ dnorm(0, 100),
                bS ~ dnorm(0, 100),
                bWS ~ dnorm(0, 100),
                sigma ~ dunif(0, 100)
              ),
              data = d,
              start = list(a = mean(d$blooms), bW = 0, bS = 0, bWS = 0,
                           sigma = sd(d$blooms)) )

(c <- compare(h3, h4))
```

```
##     WAIC pWAIC dWAIC weight    SE dSE
## h3 294.1   9.4   0.0   0.69  9.49  NA
## h4 295.7   6.5   1.6   0.31 10.33 7.6
```

To see whether we should keep these dummy variables in the model, we can compare the WAIC and weight values for this model and the model without the dummy variables. The difference in WAIC between the two models is relatively small, with the model including the dummy variable about 1.6222522 points lower. The WAIC and the weight both suggest that the model including the dummy variables is doing a better job, but the small difference in WAIC and the fact that the model without the dummy variables is still retaining about a quarter of the weight, indicates that perhaps there is something that the dummy variables are not capturing well. To investigate this possibility, we can plot the posterior distributions of the bed coefficients.

```
post <- extract.samples(h3)

p <- tibble(a = post$a,
                  b = post$a + post$bB,
                  c = post$a + post$bC)

p <- gather(p, key = bed, value = "post", a:c)

ggplot(data = p, mapping = aes(x = post, color = bed)) + geom_density() +
       ggtitle("Posterior Distibution of the Bed Coefficients")
```



Following the point estimates reported in the first hard problem, the posterior distributions depicted here show that while there seems to be a distinct difference between the effect of bed A as compared to bed B and C, it is difficult to tell whether there is really a difference in the bloom sizes of bed B and C themselves. Since each dummy variable adds a coefficient to the model, which then is included in the penalty term for the WAIC, the fact that there are 2 coefficients estimated - one for beds B and one for bed C - although they do not appear all that different, may be what is driving up the WAIC. Perhaps another model which collapses the dummy variables for beds B and C into a single dummy variable indicating that the plant was in either bed B or C would reduce the WAIC even more.

7H3

a) One possibility for the finding that ruggedness has a different relationship with GDP for African nations as opposed to non-African nations is that the country Seychelles, which is in Africa, is very rugged, but maintains a high economic livelihood due to its tourism industry. This makes it different than most other African nations, and as such, may be an outlier that is driving the interaction effect. To test this, we can estimate models with all of the data and then models using data that excludes Seychelles.

```
#load data
data(rugged)
r <- rugged
r <- r %>% mutate(log_gdp = log(rgdppc_2000)) %>%
           filter(!is.na(log_gdp))

h5 <- rethinking::map(
              alist(
                log_gdp ~ dnorm(mu, sigma),
                mu <- a + bA*cont_africa + bR*rugged + bAR*(cont_africa*rugged),
                a ~ dnorm(0, 100),
                bA ~ dnorm(0, 1),
                bR ~ dnorm(0, 1),
                bAR ~ dnorm(0, 1),
                sigma ~ dunif(0, 100)
              ),
              data = r)

r2 <- r %>% filter(country != "Seychelles")

h6 <- rethinking::map(
              alist(
                log_gdp ~ dnorm(mu, sigma),
                mu <- a + bA*cont_africa + bR*rugged + bAR*(cont_africa*rugged),
                a ~ dnorm(0, 100),
                bA ~ dnorm(0, 1),
                bR ~ dnorm(0, 1),
                bAR ~ dnorm(0, 1),
                sigma ~ dunif(0, 100)
              ),
              data = r2)

coeftab(h5, h6)
```

```
##       h5      h6
## a        9.18    9.19
## bA      -1.85   -1.78
## bR      -0.18   -0.19
## bAR      0.35    0.25
## sigma    0.93    0.93
## nobs      170     169
```

Looking at the coefficients estimated by the two models, a few differences emerge. First, the estimated effect on log GDP of being an African nation is more positive in the model without Seychelles, but only by 0.061455. The more interesting difference is that of the coefficient estimated for the interaction effect. The model estimated using data not including Seychelles is much smaller in magnitutde than the model estimated using all of the data - a full 0.0958019 of a point lower. The smaller, but still positive interaction effect suggests that when Seychelles is not included in the data, then the effect of ruggedness depends on continent to a much lesser degree than it does when Seychelles is included.

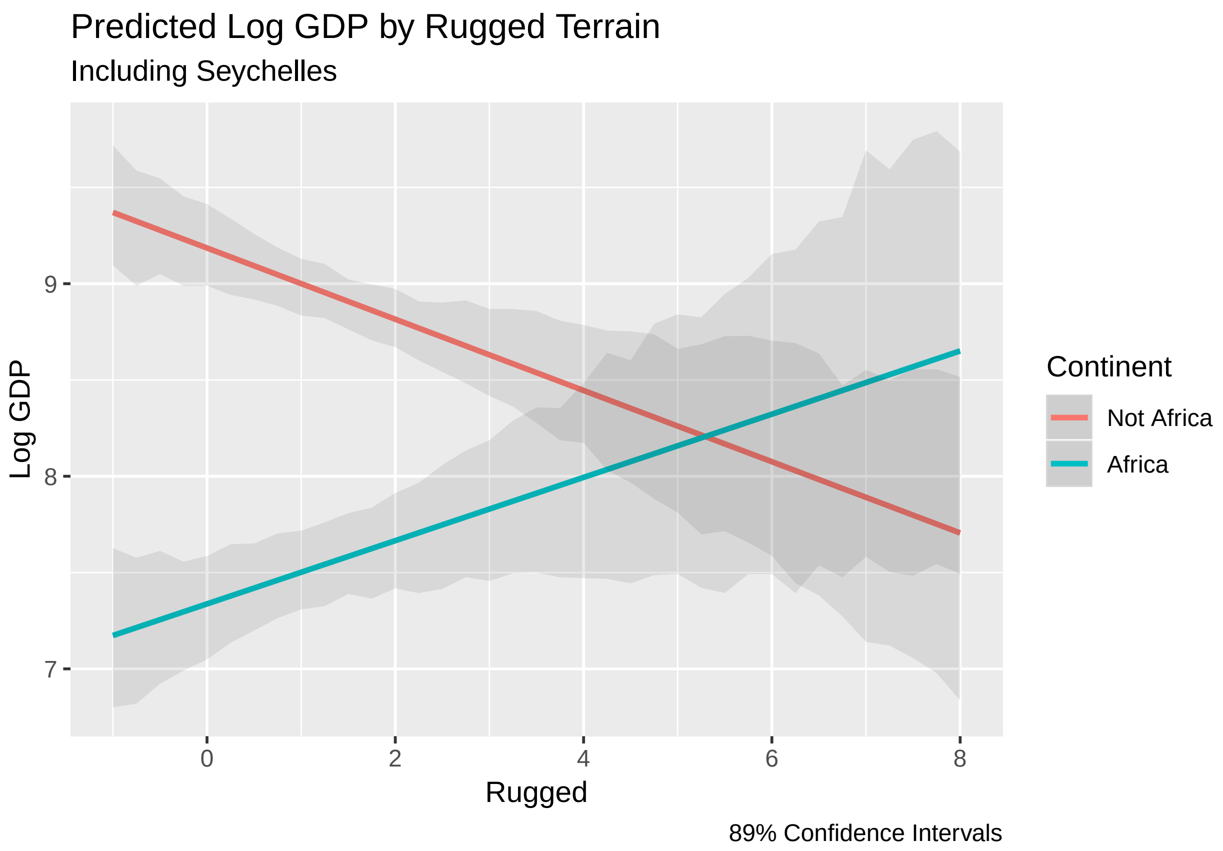
b) To understand the difference in interaction effects here more concretely, we should visualize the predictions for each model.

```r
#steve's code for plotting interactions
#PLOT MODEL PREDICTIONS WITH SEYCHELLES
# construct fake data
cont_af <- c(0, 1)
rugged.seq <- seq(-1, 8, by= 0.25)
predvals <- as.tibble(expand.grid(cont_af, rugged.seq))   # get all combinations with expand grid
```

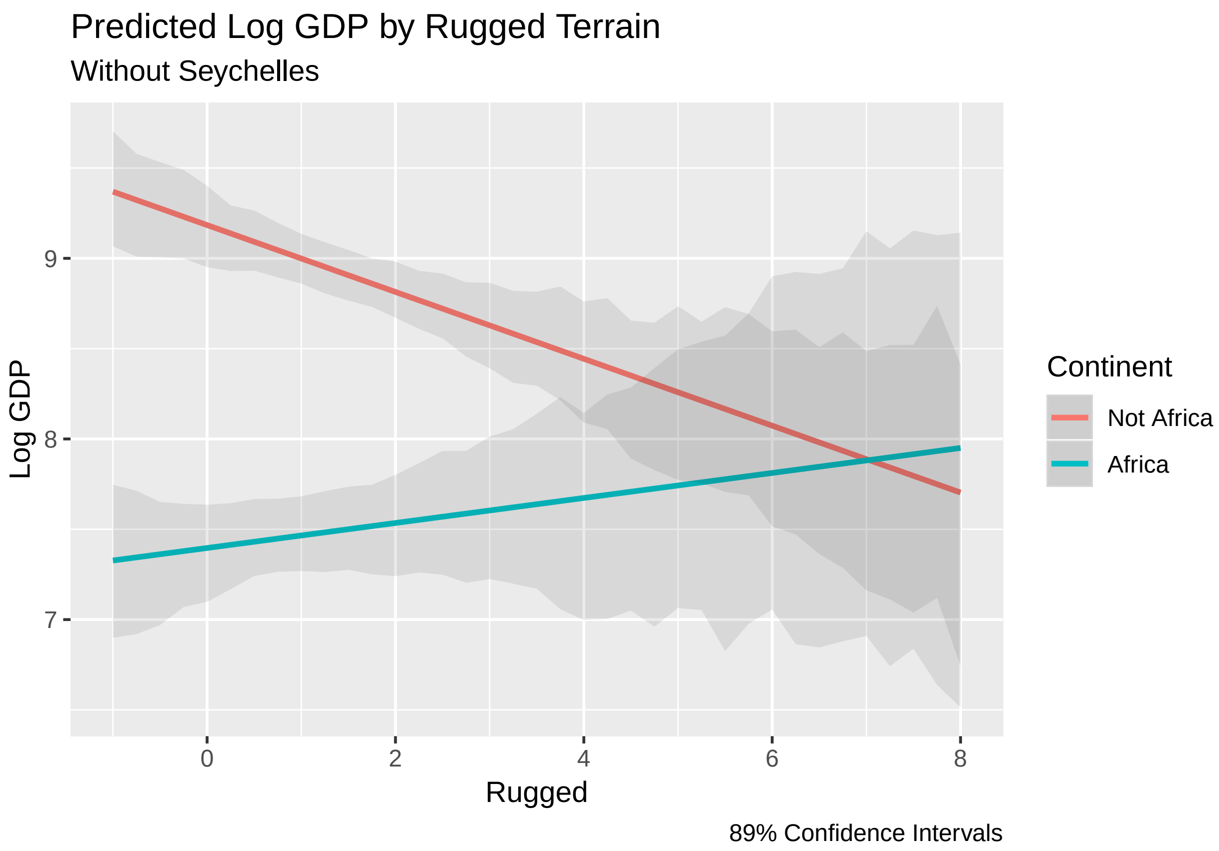
```
## Warning: `as.tibble()` is deprecated, use `as_tibble()` (but mind the new semantics).
## This warning is displayed once per session.
```

```r
colnames(predvals) <- c("continent", "rugged")      # make sure cols are labeled right
predvals$combo <- seq(1:nrow(predvals))                       # index each combo of the vars

# set values for simulation
dimx <- nrow(predvals)                                        # dim of prediction matrix
nsamps <- 500                                                 # how many samples per combo
samples <-
  MASS::mvrnorm( mu = h5@coef ,
                          Sigma = h5@vcov ,
                 n = dimx*nsamps ) %>%                        # length is number of combos X sims per c
  as.tibble %>%
  mutate(combo = rep(1:dimx , nsamps ))                       # index for joining

# merge together and create plotting values
samples <-
  full_join( samples, predvals, by = "combo") %>%             # merge in predvals then get yhat (below)
  mutate( yhat = a + bA*continent + bR*rugged +
            bAR*(continent*rugged)) %>%         # yhat for each draw
  group_by(continent, rugged) %>%                              # group by unique combos for calcs
  mutate( mmu = mean(yhat) ,                                  # mean of estimate
          lbmu = HPDI(yhat , prob = .89)[1] ,                 # LB of estimate
          ubmu = HPDI(yhat , prob = .89)[2] ) %>%             # UB of estimate
  slice(1)

# plot
p_s <-  ggplot(samples, aes(x = rugged, group = factor(continent) ) ) +
        geom_smooth(aes(y = mmu , color = factor(continent) ) , method = "lm" ) +
        geom_ribbon(aes(ymin = lbmu , ymax = ubmu) , alpha = .1)  +
        scale_color_hue(labels = c("Not Africa", "Africa")) +
        guides(color=guide_legend("Continent")) +
        labs(title = "Predicted Log GDP by Rugged Terrain",
             subtitle = "Including Seychelles",
             x = "Rugged" ,
             y = "Log GDP" ,
             caption = "89% Confidence Intervals")

#PLOT WITHOUT SEYCHELLES
cont_af <- c(0, 1)
rugged.seq <- seq(-1, 8, by= 0.25)
predvals <- as.tibble(expand.grid(cont_af, rugged.seq))   # get all combinations with expand grid
colnames(predvals) <- c("continent", "rugged")      # make sure cols are labeled right
predvals$combo <- seq(1:nrow(predvals))                       # index each combo of the vars

# set values for simulation
```

```
dimx <- nrow(predvals)                                        # dim of prediction matrix
nsamps <- 500                                                 # how many samples per combo
samples <-
  MASS::mvrnorm( mu = h6@coef ,
                          Sigma = h6@vcov ,
                 n = dimx*nsamps ) %>%                        # length is number of combos X sims per c
  as.tibble %>%
  mutate(combo = rep(1:dimx , nsamps ))                       # index for joining

# merge together and create plotting values
samples <-
  full_join( samples, predvals, by = "combo") %>%             # merge in predvals then get yhat (below)
  mutate( yhat = a + bA*continent + bR*rugged +
            bAR*(continent*rugged)) %>%         # yhat for each draw
  group_by(continent, rugged) %>%                              # group by unique combos for calcs
  mutate( mmu = mean(yhat) ,                                  # mean of estimate
          lbmu = HPDI(yhat , prob = .89)[1] ,                 # LB of estimate
          ubmu = HPDI(yhat , prob = .89)[2] ) %>%             # UB of estimate
  slice(1)

# plot
p_ws <- ggplot(samples, aes(x = rugged, group = factor(continent) ) ) +
        geom_smooth(aes(y = mmu , color = factor(continent) ) , method = "lm" ) +
        geom_ribbon(aes(ymin = lbmu , ymax = ubmu) , alpha = .1)  +
        scale_color_hue(labels = c("Not Africa", "Africa")) +
        guides(color=guide_legend("Continent")) +
        labs(title = "Predicted Log GDP by Rugged Terrain",
             subtitle = "Without Seychelles",
             x = "Rugged" ,
             y = "Log GDP" ,
             caption = "89% Confidence Intervals")

p_s
```


Predicted Log GDP by Rugged Terrain
Including Seychelles
Log GDP
9
8
7
0
2
4
6
8
Rugged
Continent
Not Africa
Africa
89% Confidence Intervals

```
p_ws
```

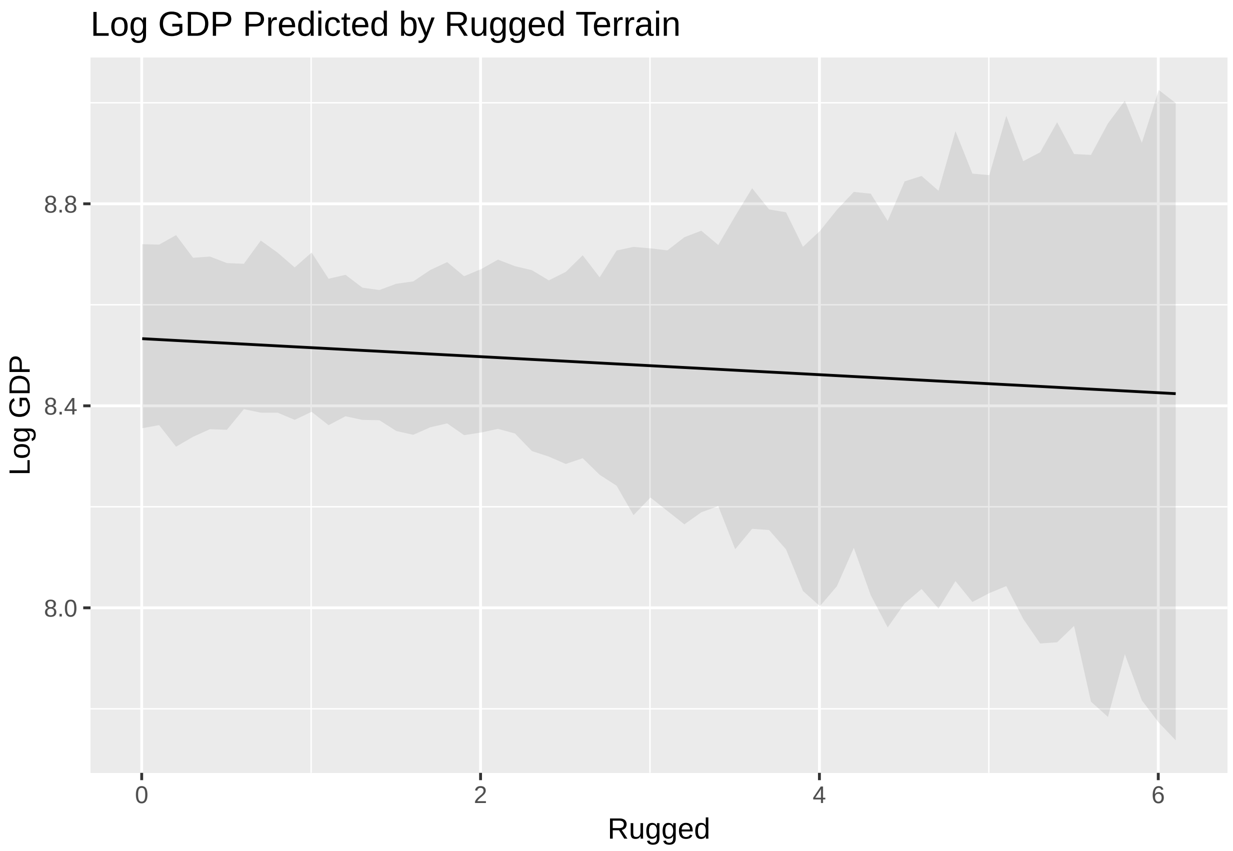


The two plots here visualize the interactions reported previously. In the second model, the slopes of the lines for the effect of rugged terrain on log GDP for African nations and non-African nations are flatter than they are in the first plot. However, the slope for Africa remains positive and that for non-African nations remains negative, suggesting that even without Seychelles, the relationship between rugged terrain and log GDP may still depend on continent. However, to be sure we should compare models of the data without Seychelles with the interaction and without, to see which model performs best, using both WAIC and weight.

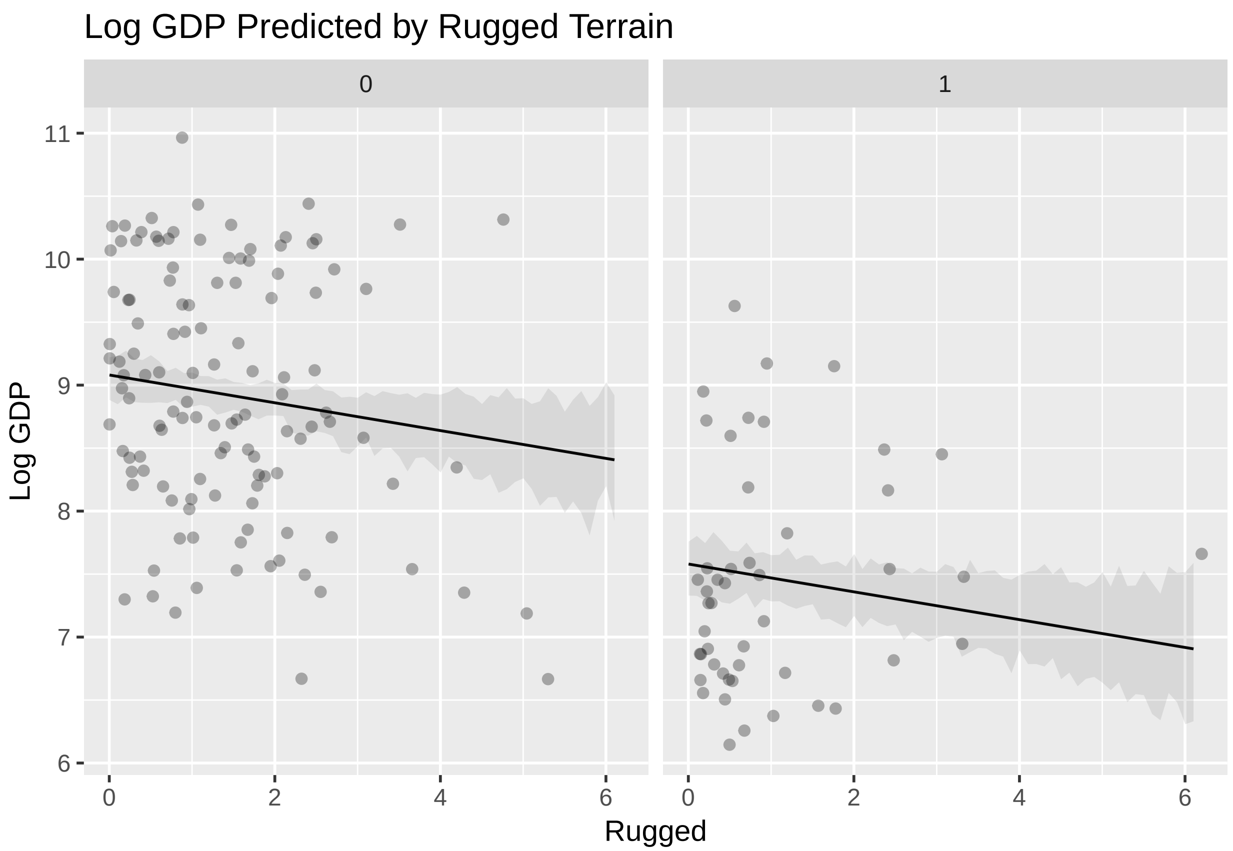
```
#just rugged variable
h7 <- rethinking::map(
              alist(
                log_gdp ~ dnorm(mu, sigma),
                mu <- a + bR*rugged,
                a ~ dnorm(0, 100),
                bR ~ dnorm(0, 1),
                sigma ~ dunif(0, 100)
              ),
              data = r2)

#with rugged and continent binary
h8 <- rethinking::map(
              alist(
                log_gdp ~ dnorm(mu, sigma),
                mu <- a + bA*cont_africa + bR*rugged,
                a ~ dnorm(0, 100),
                bA ~ dnorm(0, 1),
                bR ~ dnorm(0, 1),
                sigma ~ dunif(0, 100)
              ),
              data = r2)

#h6 already has with interaction

###PLOTS
N <- 1e4
preds <-
  as.tibble(MASS::mvrnorm(mu = h7@coef,
                          Sigma = h7@vcov , n = N )) %>%      # rather than extract.samples
  mutate(rugged = sample(seq(0.003, 6.202, by = .1), N, replace = T),
         predgdp = a + bR*rugged ,                            # line uncertainty
         preddata = rnorm(N, a + bR*rugged, sigma )) %>%         # data uncertainty
  group_by(rugged) %>%
  mutate(lb_mu = rethinking::HPDI(predgdp, prob = .89)[1],
         ub_mu = rethinking::HPDI(predgdp, prob = .89)[2],
         lb_ht = rethinking::HPDI(preddata, prob = .89)[1],
         ub_ht = rethinking::HPDI(preddata, prob = .89)[2]) %>%
  slice(1) %>%
  mutate(yhat = h7@coef["a"] + h7@coef["bR"] * rugged) %>%       # yhat for reg line
  select(rugged, predgdp, yhat, lb_mu, ub_mu, lb_ht, ub_ht)

#plot
h7_p <- ggplot(r2, aes(x = rugged)) +
  geom_line(data = preds, aes(y = yhat)) +
  geom_ribbon(data = preds, aes(ymin = lb_mu, ymax = ub_mu), alpha = .1) +
  labs(x = "Rugged",
       y = "Log GDP",
       title = "Log GDP Predicted by Rugged Terrain")

N <- 1e4
preds <-
  as.tibble(MASS::mvrnorm(mu = h8@coef,
                          Sigma = h8@vcov , n = N )) %>%      # rather than extract.samples
  mutate(rugged = sample(seq(0.003, 6.202, by = .1), N, replace = T),
         cont_africa = sample(c(0, 1), N, replace = T),
         predgdp = a + bR*rugged + bA*cont_africa,                # line uncertainty
         preddata = rnorm(N, a + bR*rugged + bA*cont_africa, sigma )) %>%         # data uncertainty
  group_by(rugged, cont_africa) %>%
  mutate(lb_mu = rethinking::HPDI(predgdp, prob = .89)[1],
         ub_mu = rethinking::HPDI(predgdp, prob = .89)[2],
         lb_ht = rethinking::HPDI(preddata, prob = .89)[1],
         ub_ht = rethinking::HPDI(preddata, prob = .89)[2]) %>%
  slice(1) %>%
  mutate(yhat = h8@coef["a"] + h8@coef["bR"] * rugged + h8@coef["bA"] * cont_africa) %>%
  select(rugged, predgdp, yhat, lb_mu, ub_mu, lb_ht, ub_ht)
```

```
## Adding missing grouping variables: `cont_africa`
```

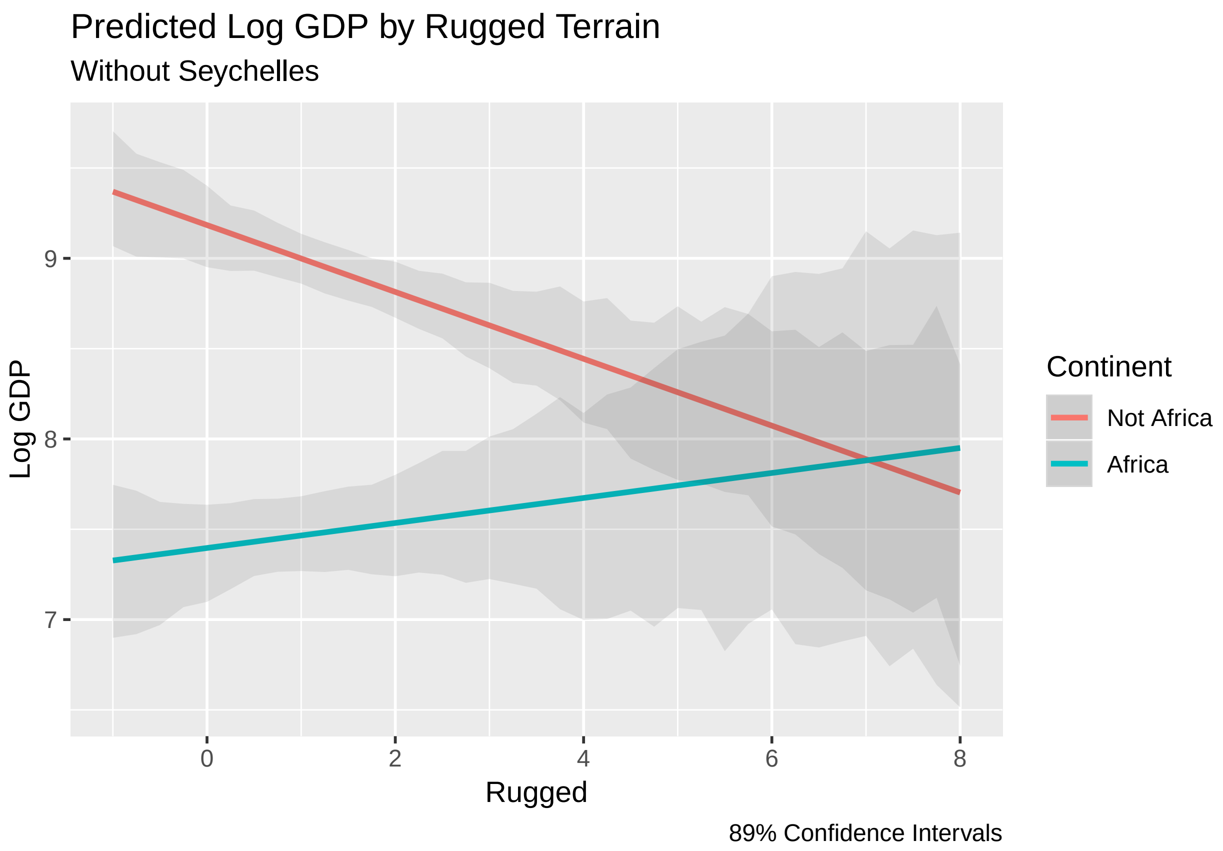
```
#plot
h8_p <- ggplot(r2, aes(x = rugged)) +
  geom_jitter(aes(y = log_gdp), alpha = .3) +
  geom_line(data = preds, aes(y = yhat)) +
  geom_ribbon(data = preds, aes(ymin = lb_mu, ymax = ub_mu),
              alpha = .1) +
  facet_wrap(.~cont_africa) +
  labs(x = "Rugged",
       y = "Log GDP",
       title = "Log GDP Predicted by Rugged Terrain")

h7_p
```



```
h8_p
```


Log GDP Predicted by Rugged Terrain
0
1
Log GDP
11
10
9
8
7
6
0
2
4
6
Rugged

```
p_ws
```



Predicted Log GDP by Rugged Terrain

Without Seychelles

89% Confidence Intervals

```r
compare(h7, h8, h6)
```

```
##      WAIC pWAIC dWAIC weight    SE   dSE
## h6 463.7   4.8   0.0   0.77 15.08    NA
## h8 466.2   4.0   2.5   0.23 14.33  3.36
## h7 536.3   2.7  72.6   0.00 13.44 15.29
```

c) After estimating these 3 models, the first only predicting log GDP by the rugged terrain, the second including a dummy variable for whether the continent is in Africa or not, and the third adding an interaciton effect between the two, I stick to the hypothesis I proposed in the previous problem that even without Seychelles, the effect of rugged terrain on log GDP does vary by continent. The evidence supporting this hypothesis comes from both the plots of predictions for each of the three models as well as the WAIC and weights when the models are compared.

First, in the plots of the predictions by the models without the interaction effects, the data are not well captured by the 89% intervals containing the data uncertainty. In particular, in the second plot, which depicts the predictions generated by the model including a dummy variable for whether the nation is in Africa or not but does not have the interaction term, the model predictions do a particularly poor job of capturing the data from the African countries, overshooting their estimates at the lower end of ruggedness and undershooting towards the higher end of the rugged terrain.

Similarly, the WAIC for the model with the interaction term is much lower than the model with only ruggedness predicting log GDP and a bit lower than the second model. Additionally, that model also gets the vast majority of the weight. All of this leads me to believe that even without Seychelles, the relationship between rugged terrain and log GDP does vary depending on continent, but not to as much of an extent as was suggested by the model fitted to the entire data.

7H4

a) One hypothesis advanced by ecologists is that language diversity may be influenced by the food security of a country. The logic being that as food security increases, the need for dependence on many others for trading and sustenance decreases, resulting in smaller communities that develop their own languages. To evaluate this hypothesis, I will model the relationship between a few measures of food security and the log of language diversity per capita. Since I am not an ecologist and know almost nothing about food security and language proliferation, the priors for all of my models are uninformative.

```
#load in data
data("nettle")
n <- nettle

#create variables
net <- n %>%
     mutate(lang_percap = num.lang/k.pop,
            log_lpc = log(lang_percap),
            log_area = log(area))

h9 <- rethinking::map(
      alist(
      log_lpc ~ dnorm(mu, sigma),
      mu <- a + bA*log_area + bM*mean.growing.season,
      a ~ dnorm(0, 1),
      bA ~ dnorm(0, 1),
      bM ~ dnorm(0, 1),
      sigma ~ dunif(0, 100)
      ),
      data = net)

h9na <- rethinking::map(
      alist(
      log_lpc ~ dnorm(mu, sigma),
      mu <- a + bM*mean.growing.season,
      a ~ dnorm(0, 1),
      bM ~ dnorm(0, 1),
      sigma ~ dunif(0, 100)
      ),
      data = net)

compare(h9, h9na)
```

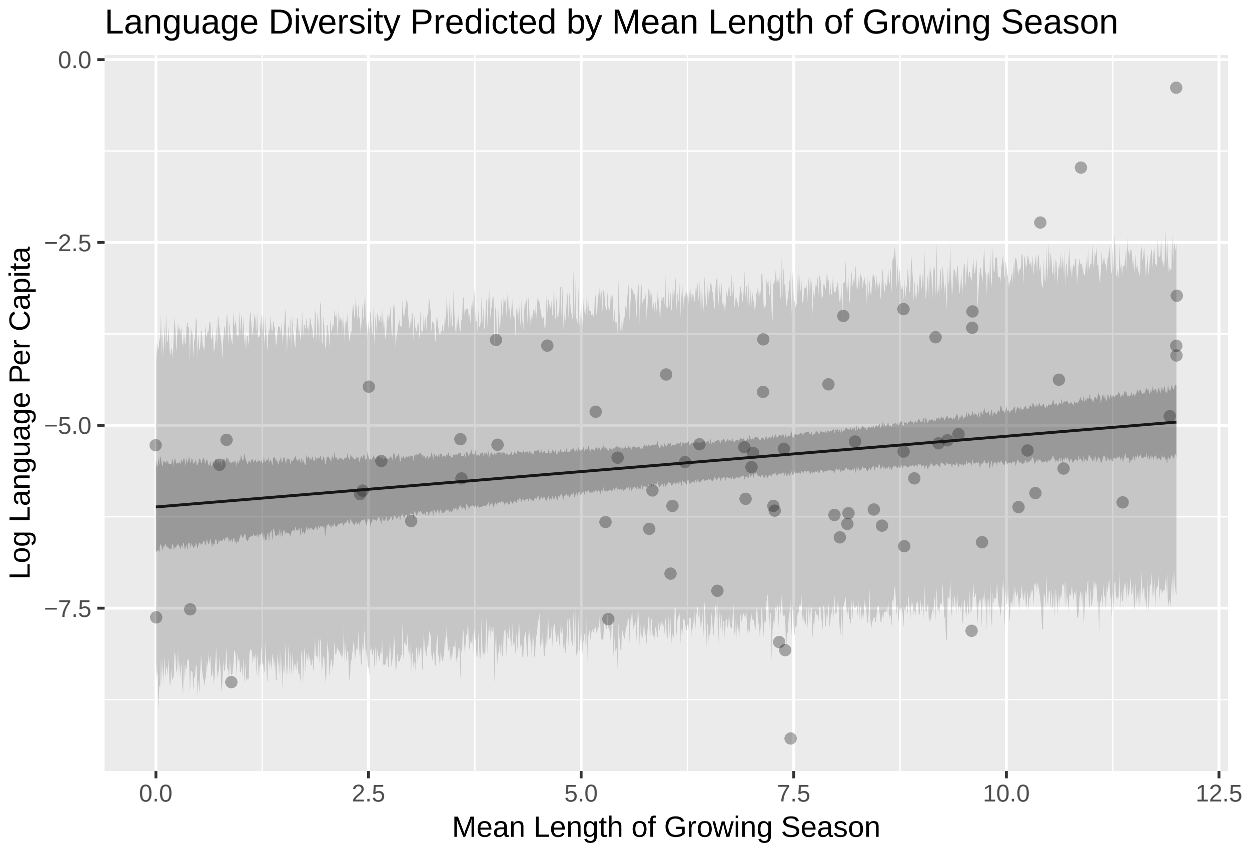
```
##       WAIC pWAIC dWAIC weight    SE  dSE
## h9    267.9   3.5   0.0   0.95 15.83   NA
## h9na 273.6   3.6   5.7   0.05 16.11 8.85
```

```
precis(h9)
```

```
##        Mean StdDev  5.5% 94.5%
## a     -0.80   0.90 -2.23  0.64
## bA    -0.41   0.07 -0.52 -0.30
## bM     0.10   0.05  0.02  0.18
## sigma  1.41   0.12  1.23  1.60
```

First, I evaluate the hypothesis that language diversity is positively associated with the average length of the growing season. I estimate two models for this, one which includes the area of a country as a control (h9) and one which does not (h9na). Since the WAIC and weight both indicate that the model with area performs better than the one which does not, I will use this model to investigate the hypothesis.

```
N <- 1e6 # sample size

# Get predictive means and data
preds <-
  as.tibble(MASS::mvrnorm(mu = h9@coef,
                          Sigma = h9@vcov , n = N )) %>%      # rather than extract.samples
  mutate(mean.growing.season =
           sample(seq(0 , 12, by = .01) , N, replace = TRUE) ,
         log_area = mean(net$log_area),
         predmean = a + bM * mean.growing.season + bA*log_area ,
         preddata = rnorm(N, a + bM * mean.growing.season + bA*log_area, sigma )) %>%
  group_by(mean.growing.season) %>%
  mutate(lb_mu = rethinking::HPDI(predmean, prob = .89)[1],
         ub_mu = rethinking::HPDI(predmean, prob = .89)[2],
         lb_y = rethinking::HPDI(preddata, prob = .89)[1],
         ub_y = rethinking::HPDI(preddata, prob = .89)[2]) %>%
  slice(1) %>%
  mutate(yhat = h9@coef["a"] + h9@coef["bM"] * mean.growing.season + h9@coef["bA"]*log_area) %>%
  select(mean.growing.season, yhat, lb_mu, ub_mu, lb_y, ub_y)

ggplot(net, aes(x = mean.growing.season)) +
  geom_jitter(aes(y = log_lpc), alpha = .3) +
  geom_line(data = preds, aes(y = yhat)) +
  geom_ribbon(data = preds, aes(ymin = lb_mu, ymax = ub_mu), alpha = .3) +
  geom_ribbon(data = preds, aes(ymin = lb_y, ymax = ub_y), alpha = .2) +
  labs(x = "Mean Length of Growing Season",
       y = "Log Language Per Capita",
       title = "Language Diversity Predicted by Mean Length of Growing Season")
```



```r
precis(h9)
```

```
##        Mean StdDev  5.5% 94.5%
## a     -0.80   0.90 -2.23  0.64
## bA    -0.41   0.07 -0.52 -0.30
## bM     0.10   0.05  0.02  0.18
## sigma  1.41   0.12  1.23  1.60
```

Interestingly, the coefficient for the mean length of growing season is slightly positive, indicating that for every 1 unit increase in the mean length of the growing season, there should be a corresponding 0.096741 increase in the log language per capita. Additionally, the 89% credible interval does not contain zero, suggesting that the relationship between the average length of a growing season and language diversity is slightly positive. However, the plot shows that there are a few outlier data points on the extreme ends of the range of mean lengths of growing seasons. I think that given the 89% credible range and the relatively small amount of data not included in the model predictions, I conclude that there is support for the hypothesis that language diversity is positively associated with the average length of the growing season.

b) A second hypothesis to evaluate is whether language diversity is negatively associated with the standard deviation of length of growing season. The logic behind this hypothesis is that uncertainty in growing season generates the need for more reliance on others and thus a common language.

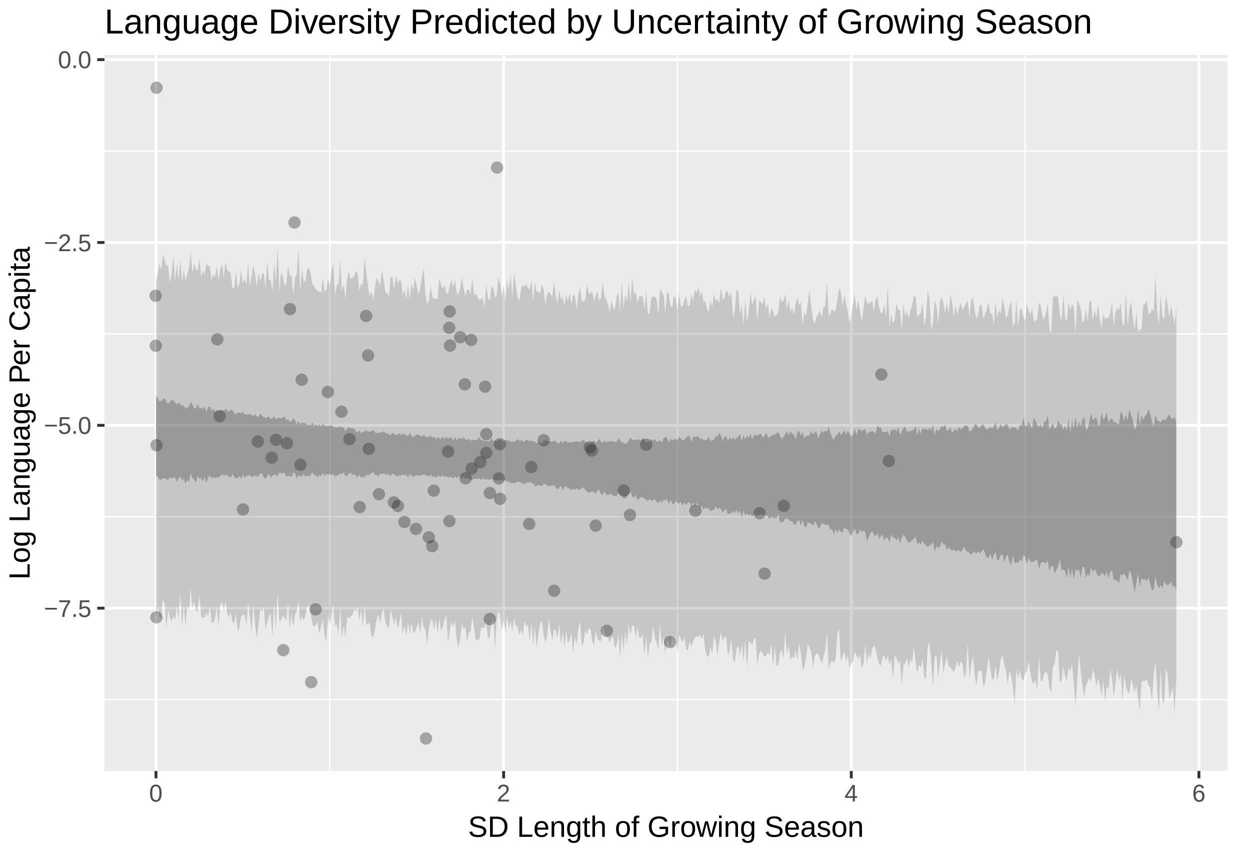
```r
h10 <- rethinking::map(
      alist(
      log_lpc ~ dnorm(mu, sigma),
      mu <- a + bA*log_area + bS*sd.growing.season,
      a ~ dnorm(0, 1),
      bA ~ dnorm(0, 1),
      bS ~ dnorm(0, 1),
      sigma ~ dunif(0, 100)
      ),
      data = net)

h10na <- rethinking::map(
      alist(
      log_lpc ~ dnorm(mu, sigma),
      mu <- a + bS*sd.growing.season,
      a ~ dnorm(0, 1),
      bS ~ dnorm(0, 1),
      sigma ~ dunif(0, 100)
      ),
      data = net)

compare(h10, h10na)
```

```
##        WAIC pWAIC dWAIC weight    SE  dSE
## h10   272.1   4.0   0.0   0.83 16.30   NA
## h10na 275.2   3.9   3.1   0.17 16.45 6.03
```

Again, comparing the model with log area as a covariate and without, the WAIC and weight indicate that the model including log area as a covariate performs better. Thus, this model will be used to evaluate the overarching hypothesis.

```
N <- 1e6 # sample size

# Get predictive means and data
preds <-
  as.tibble(MASS::mvrnorm(mu = h10@coef,
                          Sigma = h10@vcov , n = N )) %>%      # rather than extract.samples
  mutate(sd.growing.season =
           sample(seq(0 , 5.87, by = .01) , N, replace = TRUE) ,
         log_area = mean(net$log_area),
         predmean = a + bS * sd.growing.season + bA*log_area ,
         preddata = rnorm(N, a + bS * sd.growing.season + bA*log_area, sigma )) %>%
  group_by(sd.growing.season) %>%
  mutate(lb_mu = rethinking::HPDI(predmean, prob = .89)[1],
         ub_mu = rethinking::HPDI(predmean, prob = .89)[2],
         lb_y = rethinking::HPDI(preddata, prob = .89)[1],
         ub_y = rethinking::HPDI(preddata, prob = .89)[2]) %>%
  slice(1) %>%
  mutate(yhat = h9@coef["a"] + h9@coef["bS"] * sd.growing.season + h9@coef["bA"]*log_area) %>%
  select(sd.growing.season, yhat, lb_mu, ub_mu, lb_y, ub_y)

ggplot(net, aes(x = sd.growing.season)) +
  geom_jitter(aes(y = log_lpc), alpha = .3) +
  geom_line(data = preds, aes(y = yhat)) +
  geom_ribbon(data = preds, aes(ymin = lb_mu, ymax = ub_mu), alpha = .3) +
  geom_ribbon(data = preds, aes(ymin = lb_y, ymax = ub_y), alpha = .2) +
  labs(x = "SD Length of Growing Season",
       y = "Log Language Per Capita",
       title = "Language Diversity Predicted by Uncertainty of Growing Season")
```

```
## Warning: Removed 588 rows containing missing values (geom_path).
```



```
precis(h10)
```

```
##        Mean StdDev  5.5% 94.5%
## a     -0.46   0.88 -1.87  0.95
## bA    -0.37   0.08 -0.49 -0.24
## bS    -0.14   0.17 -0.42  0.13
## sigma  1.45   0.12  1.25  1.64
```

```
compare(h9, h10)
```

```
##      WAIC pWAIC dWAIC weight    SE  dSE
## h9  269.1   4.2   0.0   0.83 16.26   NA
## h10 272.3   4.1   3.2   0.17 16.25 4.45
```

The model predicts that for every one unit increase in standard deviation of the length of the growing season, there should be a corresponding -0.1446369 point decrease in log language per capita. However, the 89% credible interval does contain 0, suggesting that there may not be a genuinely negative relationship between the standard deviation of the length of the growing season and language diversity. The plot shows this uncertainty in the data points at the low end of the range of standard deviations that fall outside of the 89% credible interval and the lack of data inside of the 89% credible interval at the high end of the range of standard deviation of the growing season length.

When comparing this model to the one previously, it does not hold up well. This model has a higher WAIC and much lower weight than the model with only the mean length of the growing season, indicating that this model does not perform very well. Along with the 89% credible interval containing zero, I believe this indicates that the hypothesis that language diversity is negatively associated with the standard deviation of length of growing season is not supported by the data.

c) One possibility is that the mean and standard deviation of the growing season length interact in a synergistic way, such that the effect of each is stronger when the other is particularly high. In other words, the negative effect of a high standard deviation in growing length is stronger when the average growing length is long, and vice versa. To investigate this hypothesis, a model including an interaction term is estimated.

```
h11 <- rethinking::map(
      alist(
      log_lpc ~ dnorm(mu, sigma),
      mu <- a + bA*log_area + bS*sd.growing.season + bM*mean.growing.season +
            bMS*sd.growing.season*mean.growing.season,
      a ~ dnorm(0, 1),
      bA ~ dnorm(0, 1),
      bS ~ dnorm(0, 1),
      bM ~ dnorm(0, 1),
      bMS ~ dnorm(0, 1),
      sigma ~ dunif(0, 100)
      ),
      data = net)

h11na <- rethinking::map(
      alist(
      log_lpc ~ dnorm(mu, sigma),
      mu <- a + bS*sd.growing.season + bM*mean.growing.season +
            bMS*sd.growing.season*mean.growing.season,
      a ~ dnorm(0, 1),
      bS ~ dnorm(0, 1),
      bM ~ dnorm(0, 1),
      bMS ~ dnorm(0, 1),
      sigma ~ dunif(0, 100)
      ),
      data = net)

compare(h11, h11na)
```

```
##        WAIC pWAIC dWAIC weight    SE  dSE
## h11   268.1   5.8   0.0   0.85 17.19   NA
## h11na 271.6   6.3   3.4   0.15 16.55 7.89
```

Once again, the model including area as a covariate has a lower WAIC and higher weight than the model without, so I use that one to evaluate the hypothesis.

```
precis(h11)
```

```
##        Mean StdDev  5.5% 94.5%
## a     -1.17   0.92 -2.65  0.31
## bA    -0.41   0.08 -0.53 -0.28
## bS     0.39   0.37 -0.20  0.99
## bM     0.18   0.07  0.07  0.29
## bMS   -0.08   0.05 -0.15  0.00
## sigma  1.37   0.11  1.19  1.55
```

The model estimates that the interaction effect of mean length of growing season and the standard deviation of the growing season is -0.0760555. However, the 94.5% is anchored at 0, so there is a chance that the effect is not existent. Interestingly, including the interaction effect increased the coefficient for mean length of growing season and the coefficient for the standard deviation of the growing season became positive, such that for every one unit increase in standard deviation.

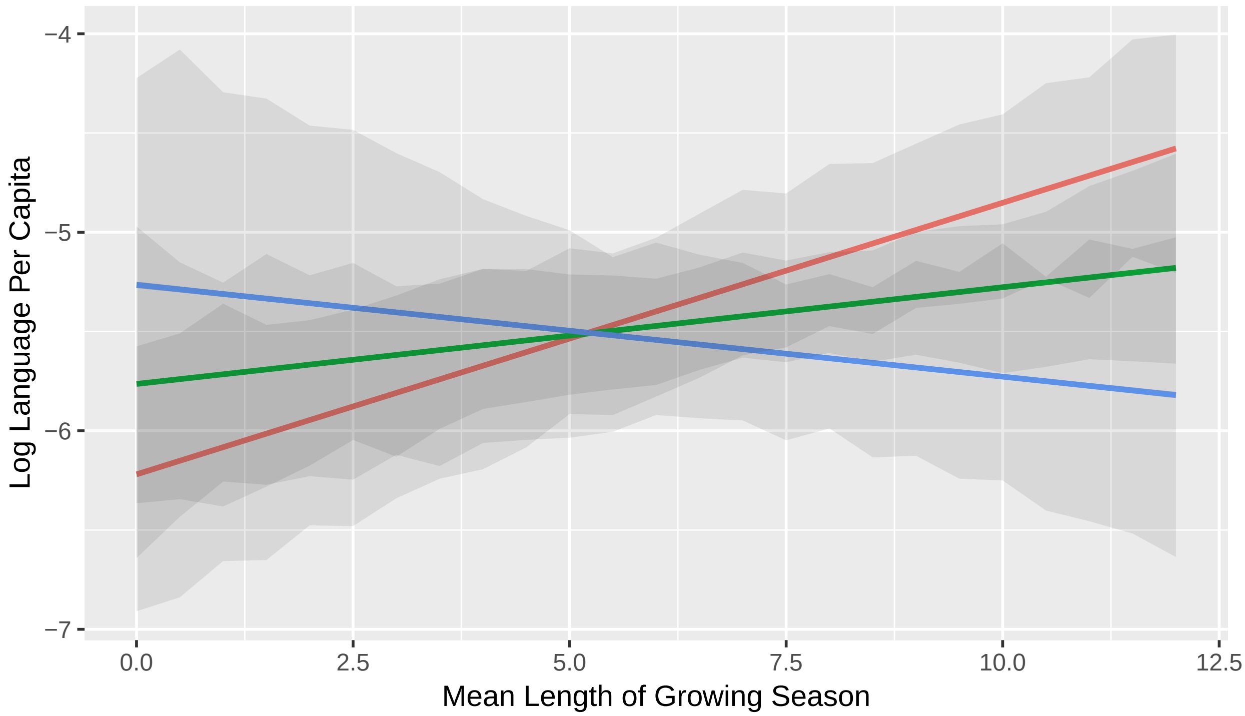
To truly understand the interaction effect, we should plot predictions for this model.

```
# construct fake data
log_area <- mean(net$log_area)
sd_gs <- quantile(net$sd.growing.season, c(.1, .5, .9))
m_gs <- seq(0, 12, by = 0.5 )
predvals <- as.tibble(expand.grid(sd_gs, m_gs, log_area))   # get all combinations with expand grid
colnames(predvals) <- c("sdgs", "meangs", "log_area")      # make sure cols are labeled right
predvals$combo <- seq(1:nrow(predvals))                       # index each combo of the vars

# set values for simulation
dimx <- nrow(predvals)                                        # dim of prediction matrix
nsamps <- 500                                                 # how many samples per combo

# draw samples for all combos
samples <-
  MASS::mvrnorm( mu = h11@coef ,
                          Sigma = h11@vcov ,
                 n = dimx*nsamps ) %>%                        # length is number of combos X sims per c
  as.tibble %>%
  mutate(combo = rep(1:dimx , nsamps ))                       # index for joining

# merge together and create plotting values
samples <-
  full_join( samples, predvals, by = "combo") %>%             # merge in predvals then get yhat (below)
  mutate( yhat = a + bA*log_area + bS*sdgs + bM*meangs +
            bMS*sdgs*meangs) %>%          # yhat for each draw
  group_by(sdgs, meangs) %>%                              # group by unique combos for calcs
  mutate( mmu = mean(yhat) ,                                  # mean of estimate
          lbmu = HPDI(yhat , prob = .89)[1] ,                 # LB of estimate
          ubmu = HPDI(yhat , prob = .89)[2] ) %>%             # UB of estimate
  slice(1)

# plot
meanx <- ggplot(samples, aes(x = meangs , group = factor(sdgs) ) ) +
         geom_smooth(aes(y = mmu , color = factor(sdgs) ) , method = "lm" ) +
         geom_ribbon(aes(ymin = lbmu , ymax = ubmu) , alpha = .1)  +
         theme(legend.position = "none") +
         labs(title = "Language Diversity by Mean Growing Season" ,
             subtitle = "at the 10th, 50th, and 90th percentiles of SD of Growing Season" ,
             x = "Mean Length of Growing Season" ,
             y = "Log Language Per Capita" ,
             caption = "89% Confidence Intervals; other values held at their means")

#SYMMETRIC RELATIONSHIP

# construct fake data
log_area <- mean(net$log_area)
m_gs <- quantile(net$mean.growing.season, c(.1, .5, .9))
sd_gs <- seq(0.527, 2.918,length.out = 20)
predvals <- as.tibble(expand.grid(sd_gs, m_gs, log_area))   # get all combinations with expand grid
colnames(predvals) <- c("sdgs", "meangs", "log_area")      # make sure cols are labeled right
predvals$combo <- seq(1:nrow(predvals))                       # index each combo of the vars
```

```
# set values for simulation
dimx <- nrow(predvals)                                        # dim of prediction matrix
nsamps <- 500                                                 # how many samples per combo

# draw samples for all combos
samples <-
  MASS::mvrnorm( mu = h11@coef ,
                          Sigma = h11@vcov ,
                 n = dimx*nsamps ) %>%                        # length is number of combos X sims per c
  as.tibble %>%
  mutate(combo = rep(1:dimx , nsamps ))                       # index for joining

# merge together and create plotting values
samples <-
  full_join( samples, predvals, by = "combo") %>%             # merge in predvals then get yhat (below)
  mutate( yhat = a + bA*log_area + bS*sdgs + bM*meangs +
            bMS*sdgs*meangs) %>%         # yhat for each draw
  group_by(meangs, sdgs) %>%                              # group by unique combos for calcs
  mutate( mmu = mean(yhat) ,                                  # mean of estimate
          lbmu = HPDI(yhat , prob = .89)[1] ,                 # LB of estimate
          ubmu = HPDI(yhat , prob = .89)[2] ) %>%             # UB of estimate
  slice(1)

# plot
sdx <- ggplot(samples, aes(x = sdgs , group = factor(meangs) ) ) +
       geom_smooth(aes(y = mmu , color = factor(meangs) ) , method = "lm" ) +
       geom_ribbon(aes(ymin = lbmu , ymax = ubmu) , alpha = .1)  +
       theme(legend.position = "none") +
       labs(title = "Language Diversity by SD Growing Season" ,
           subtitle = "at the 10th, 50th, and 90th percentiles of Mean of Growing Season" ,
           x = "SD Length of Growing Season" ,
           y = "Log Language Per Capita" ,
           caption = "89% Confidence Intervals; other values held at their means")

meanx
```

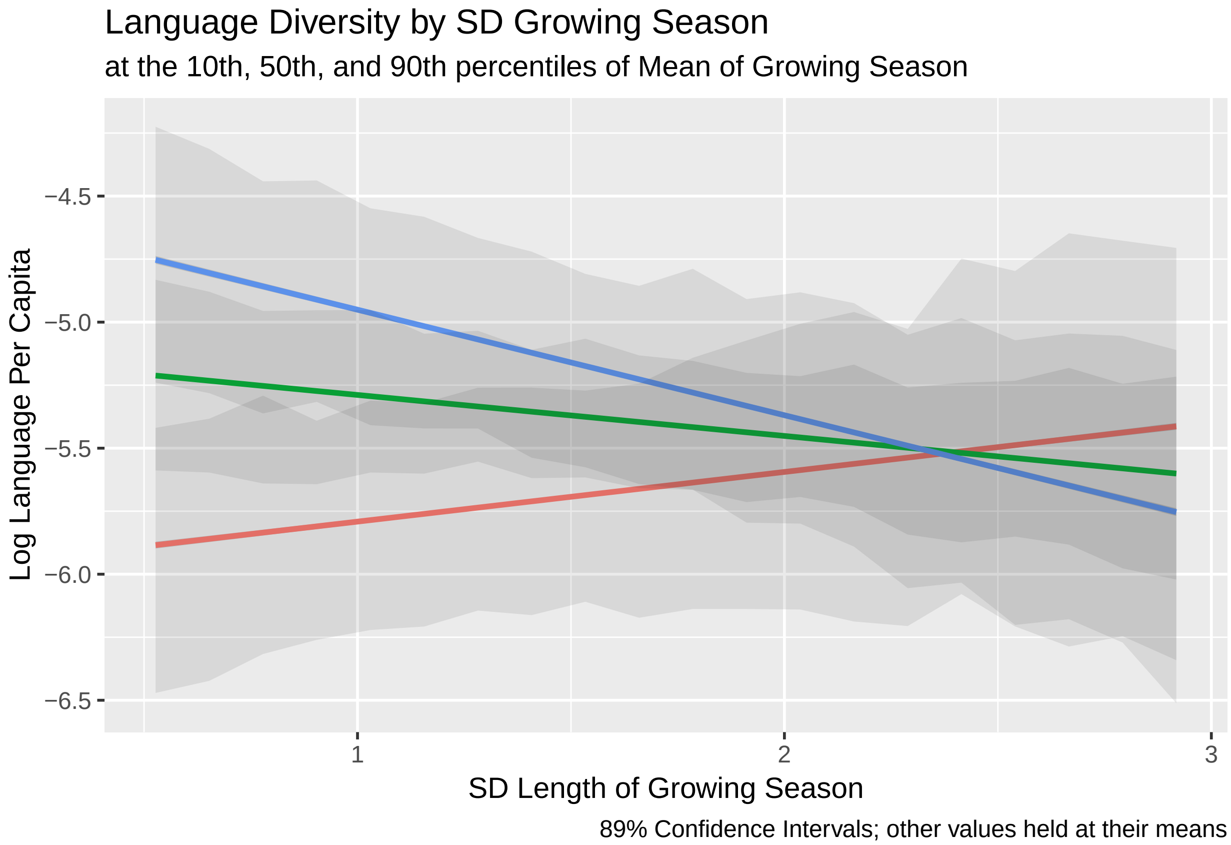
Language Diversity by Mean Growing Season

at the 10th, 50th, and 90th percentiles of SD of Growing Season



89% Confidence Intervals; other values held at their means

```
sdx
```



These two plots depict the interaction effect in symmetric ways. The first shows how the relationship between mean length of the growing season and the log language per capita depends on the standard deviation of the length of the growing season. The second shows how the relationship between the standard deviation of the length of the growing season and the log language per capita depends on the mean length of the growing season. These plots show that when the mean and the standard deviation of the length of the growing season are both high, language diversity is more likely to be low. This does support the hypothesis being tested, although we need to check to see if the model has a lower WAIC and higher weight when compared to the other two models estimated.

```
compare(h9, h10, h11)
```

```
##      WAIC pWAIC dWAIC weight    SE  dSE
## h11 268.2   5.8   0.0   0.56 16.86   NA
## h9  269.1   4.2   0.9   0.36 16.11 3.93
## h10 272.0   3.9   3.8   0.09 16.19 5.05
```

```
compare(h9, h11)
```

```
##      WAIC pWAIC dWAIC weight    SE  dSE
## h11 268.7   6.2   0.0   0.52 17.31   NA
## h9  268.9   4.1   0.2   0.48 16.03 4.16
```

Comparing this model to the previous two indicates that this model performs a little less well than the model with only mean length of the growing season, but both of those fit better than the model that predicts log language per capita with just the standard deviation of the growing season. Interestingly, when comparing only the two models that had a reasonable amount of weight out of the original three, the model with the interaction effect has a slightly lower WAIC and a higher weight. However, the difference in WAIC is quite small and the model with only the mean growing season still has a fairly high weight, suggesting that we cannot be fully certain that the interaction effect is truly existent. Thus, along with the inclusion of 0 in the 89% credible interval of the interaction, I do not think that the data fully supports the hypothesis that there is a synergistic interaction between the average length of a growing season and the standard deviation of the length of the growing season.