Assignment 2 - Methods 4

Laurits Lyngbaek

2025-03-18

Second assignment

The second assignment uses chapter 3, 5 and 6. The focus of the assignment is getting an understanding of causality.

Chapter 3: Causal Confussion

Reminder: We are tying to estimate the probability of giving birth to a boy I have pasted a working solution to questions 6.1-6.3 so you can continue from here:)

3H3 Use rbinom to simulate 10,000 replicates of 200 births. You should end up with 10,000 numbers, each one a count of boys out of 200 births. Compare the distribution of predicted numbers of boys to the actual count in the data (111 boys out of 200 births).

```
# 3H1
# Find the posterior probability of giving birth to a boy:
pacman::p_load(rethinking)
data(homeworkch3)
set.seed(1)
W <- sum(birth1) + sum(birth2)
N <- length(birth1) + length(birth2)
p_grid <-seq(from =0, to = 1, len =1000)
prob_p <- rep(1,1000)
prob_data <- dbinom(W,N,prob=p_grid)
posterior <-prob_data * prob_p
posterior <- posterior / sum(posterior)

# 3H2
# Sample probabilities from posterior distribution:
samples <- sample (p_grid, prob = posterior, size =1e4, replace =TRUE)

# 3H3
# Simulate births using sampled probabilities as simulation input, and check if they allign w
ith real value.
simulated_births <- rbinom(n = 1e4, size = N, prob = samples)
rethinking::dens(simulated_births,show.PI = 0.95)
```

```
## Warning in plot.window(...): "show.PI" is not a graphical parameter
```

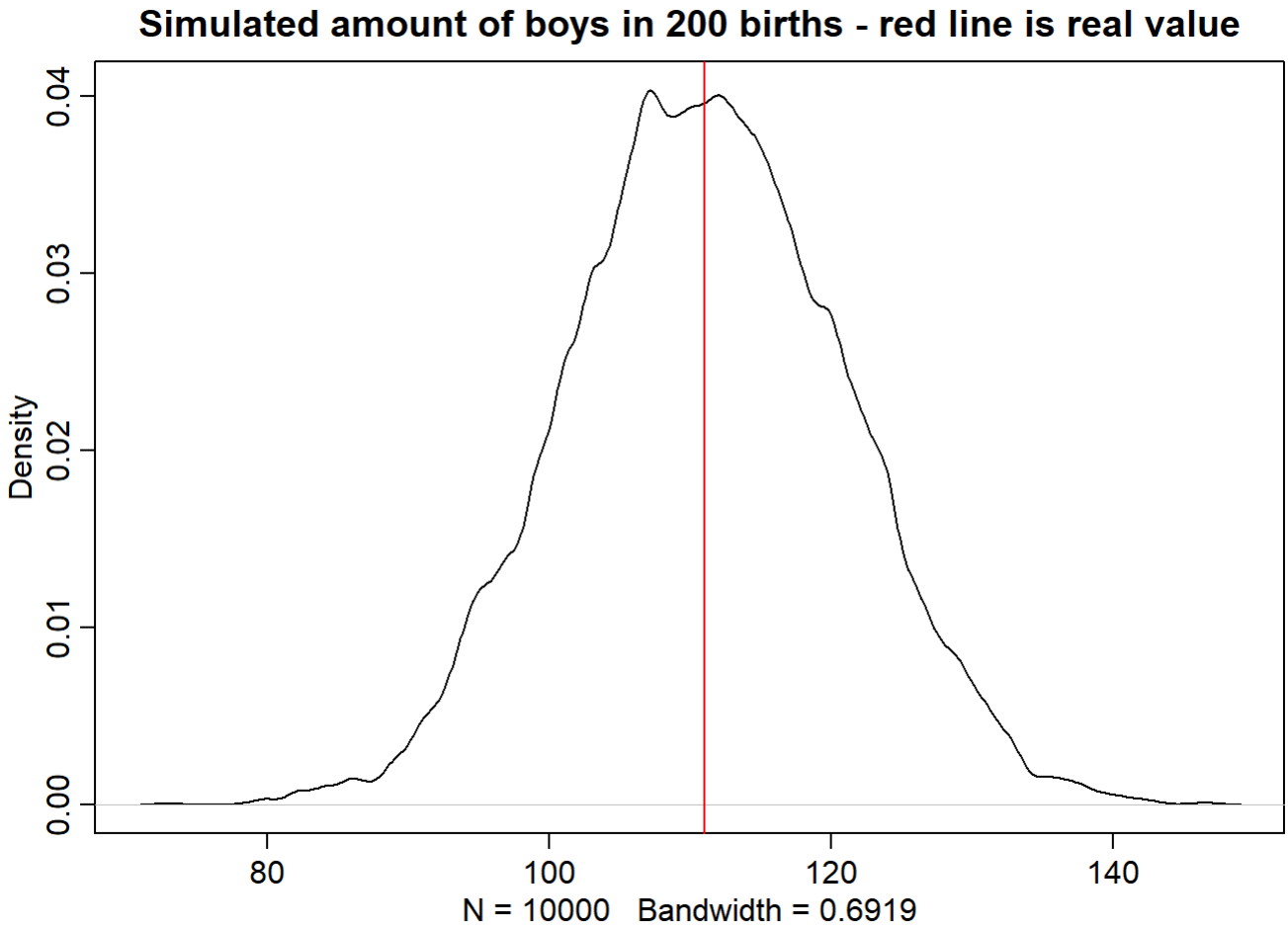
```
## Warning in plot.xy(xy, type, ...): "show.PI" is not a graphical parameter
```

```
## Warning in axis(side = side, at = at, labels = labels, ...): "show.PI" is not a
## graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "show.PI" is not a
## graphical parameter
```

```
## Warning in box(...): "show.PI" is not a graphical parameter
```

```
## Warning in title(...): "show.PI" is not a graphical parameter
```

```
abline(v=W, col="red")
title("Simulated amount of boys in 200 births - red line is real value")
```



3H4. Now compare 10,000 counts of boys from 100 simulated firstborns only to the number of boys in the first births, birth1. How does the model look in this light?

```
simulated_first_births <- rbinom(n = 1e4, size = length(birth1), prob = samples)
W1 <- sum(birth1)

rethinking::dens(simulated_first_births,show.PI = 0.95)
```

```
## Warning in plot.window(...): "show.PI" is not a graphical parameter
```

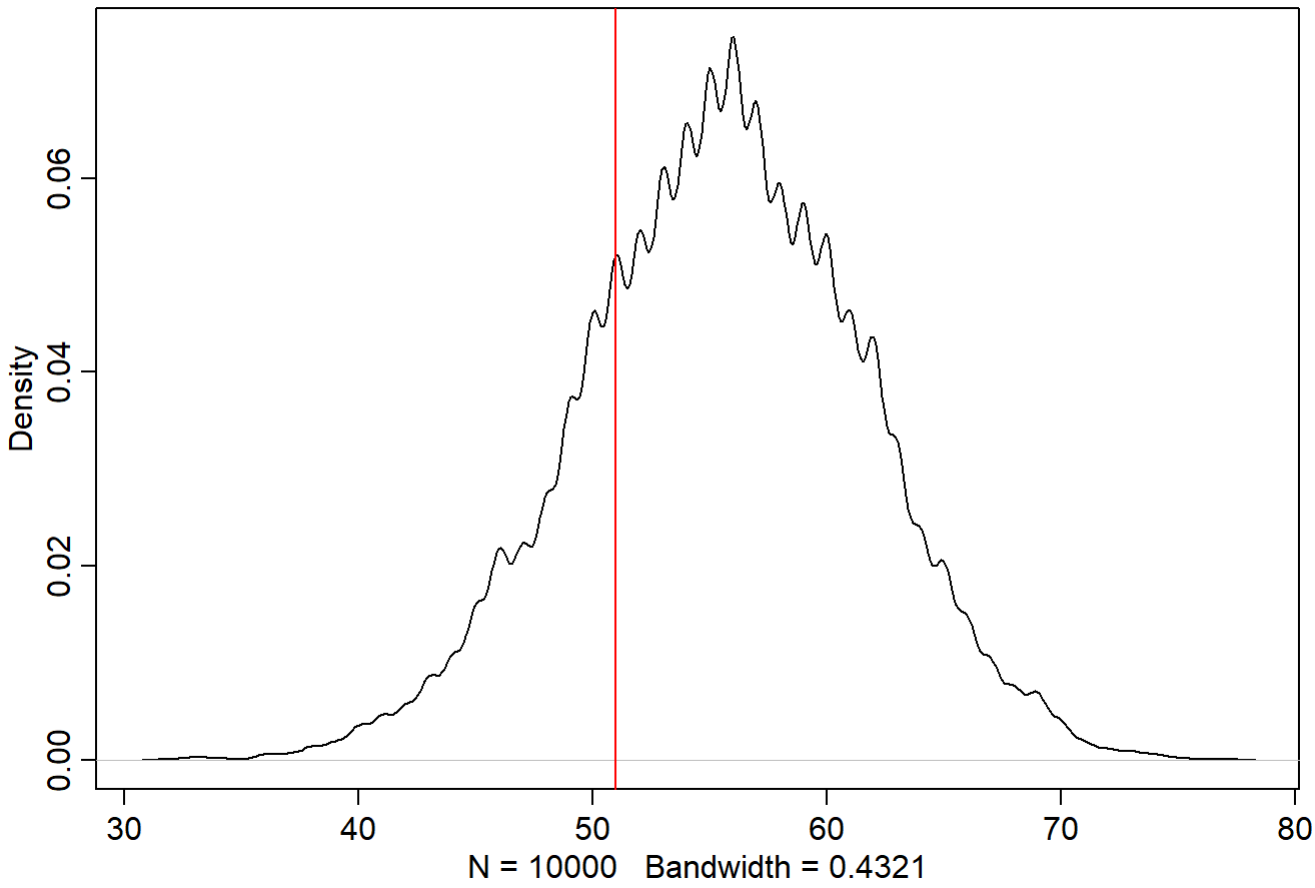
```
## Warning in plot.xy(xy, type, ...): "show.PI" is not a graphical parameter
```

```
## Warning in axis(side = side, at = at, labels = labels, ...): "show.PI" is not a
## graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "show.PI" is not a
## graphical parameter
```

```
## Warning in box(...): "show.PI" is not a graphical parameter
```

```
## Warning in title(...): "show.PI" is not a graphical parameter
```

```
abline(v=W1, col="red")
```



The model seems to overestimate the number of firstborn boys.

3H5. The model assumes that sex of first and second births are independent. To check this assumption, focus now on second births that followed female first borns. Compare 10,000 simulated counts of boys to only those second births that followed girls. To do this correctly, you need to cound the number of first borns who were girls and simulate that many births, 10,000 times. Compare the counts of boys in your simulations to the actual observed count of boys following girls. How does the model look in this light? Any guesses what is going on in these data?

```
n_first_born_girls <- -sum(birth1-1)

simulated_birth_after_girl <- rbinom(1e4, size = n_first_born_girls, prob = samples) # Simula
te boys born after a girl
W2 <- sum(birth2[birth1 == 0]) # Get number of boys born after a girl

rethinking::dens(simulated_birth_after_girl,show.PI = 0.95)
```

```
## Warning in plot.window(...): "show.PI" is not a graphical parameter
```

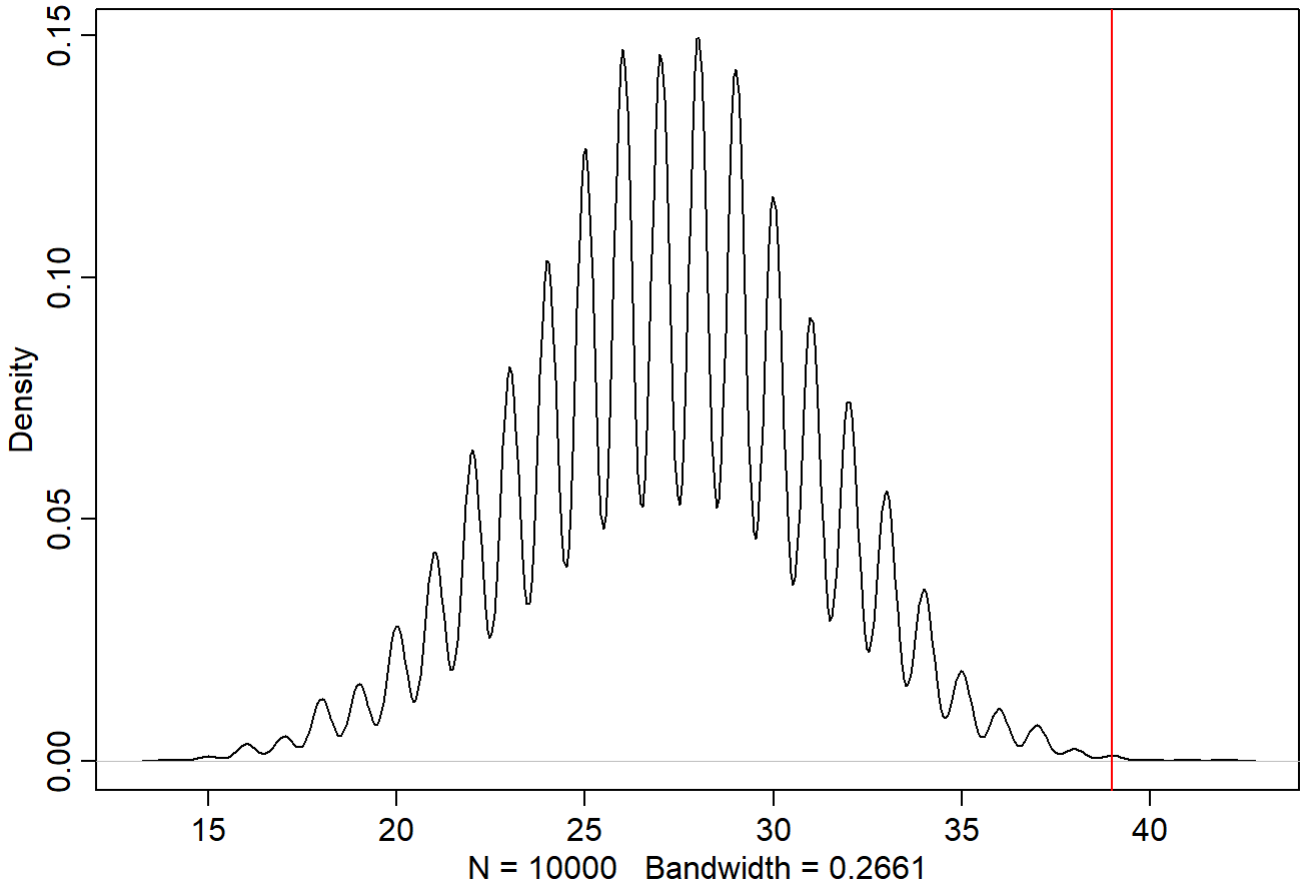
```
## Warning in plot.xy(xy, type, ...): "show.PI" is not a graphical parameter
```

```
## Warning in axis(side = side, at = at, labels = labels, ...): "show.PI" is not a
## graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "show.PI" is not a
## graphical parameter
```

```
## Warning in box(...): "show.PI" is not a graphical parameter
```

```
## Warning in title(...): "show.PI" is not a graphical parameter
```

```
abline(v=W2, col="red")
```



We know that there was 49 firstborn girls. We then simulate n=49 births 10,000 times, i.e. we simulate the 49 second-births, this simulation is based on all the data and does not care whether the birth was a first or a second birth. By comparing to the known second-births we can see the model is underestimating the number of boys born after a firstborn girl.

Chapter 5: Spurrious Correlations

Start of by checking out all the spurious correlations that exists in the world. Some of these can be seen on this wonderful website: https://www.tylervigen.com/spurious/random (https://www.tylervigen.com/spurious/random) All the medium questions are only asking you to explain a solution with words, but feel free to simulate the data and prove the concepts.

5M1. Invent your own example of a spurious correlation. An outcome variable should be correlated with both predictor variables. But when both predictors are entered in the same model, the correlation between the outcome and one of the predictors should mostly vanish (or at least be greatly reduced).

```
# Number of students
n <- 1000

# Simulate students
lecture_attendance <- rnorm(n)
happiness <- rnorm(n, lecture_attendance)
grade <- rnorm(n, lecture_attendance)

# Put the data in a data frame
d <- data.frame(lecture_attendance, happiness, grade)

# Model 1: grade ~ happiness
m5.1.1 <- quap(
  alist(
    grade ~ dnorm(mu, sigma),
    mu <- a + b_h * happiness,
    a ~ dnorm(0, 0.5),
    b_h ~ dnorm(0, 0.5),
    sigma ~ dexp(1)
  ), data = d
)

# Model 2: grade ~ lecture_attendance
m5.1.2 <- quap(
  alist(
    grade ~ dnorm(mu, sigma),
    mu <- a + b_la * lecture_attendance,
    a ~ dnorm(0, 0.5),
    b_la ~ dnorm(0, 0.5),
    sigma ~ dexp(1)
  ), data = d
)

# Model 3: grade ~ happiness + lecture_attendance
m5.1.3 <- quap(
  alist(
    grade ~ dnorm(mu, sigma),
    mu <- a + b_h * happiness + b_la * lecture_attendance,
    a ~ dnorm(0, 0.5),
    b_h ~ dnorm(0, 0.5),
    b_la ~ dnorm(0, 0.5),
    sigma ~ dexp(1)
  ), data = d
)

# Look at the parameter estimates
precis(m5.1.1)
```

```
##             mean         sd        5.5%     94.5%
## a     0.03217262 0.03890111 -0.02999886 0.0943441
## b_h   0.53411511 0.02749958  0.49016546 0.5780648
## sigma 1.23338935 0.02755396  1.18935279 1.2774259
```

```
precis(m5.1.2)
```

```
##             mean         sd        5.5%      94.5%
## a     0.01260304 0.03183859 -0.03828117 0.06348726
## b_la  1.02036153 0.03138762  0.97019804 1.07052501
## sigma 1.00887001 0.02254223  0.97284317 1.04489685
```

```
precis(m5.1.3)
```

```
##             mean         sd        5.5%      94.5%
## a     0.01511390 0.03179838 -0.03570606 0.06593386
## b_h   0.06480978 0.03077415  0.01562674 0.11399281
## b_la  0.95858721 0.04292080  0.88999149 1.02718293
## sigma 1.00687975 0.02249800  0.97092361 1.04283590
```

In this simulation lecture attendance is the common cause of both happiness and grade. When you analyze grade ~ happiness, you'll see a positive correlation When you analyze grade ~ lecture_attendance, you'll see a positive correlation When you analyze grade ~ happiness + lecture_attendance, the effect of happiness disappears (except for a small amount of noise)

5M2. Invent your own example of a masked relationship. An outcome variable should be correlated with both predictor variables, but in opposite directions. And the two predictor variables should be correlated with one another.

```
n <- 1000

time_at_party <- rnorm(n) #
beers_drunk <- rnorm(n, time_at_party)
people_talked_to <- rnorm(n, time_at_party)  # Negative correlation with study_hours

chance_of_getting_friends <- rnorm(n, (people_talked_to - beers_drunk) / 2)

d <- data.frame(chance_of_getting_friends, beers_drunk, people_talked_to)

# Predict chance_of_getting_friends ~ beers_drunk
m5.2.1 <- quap(
  alist(
    chance_of_getting_friends ~ dnorm(mu, sigma),
    mu <- a + b_b * beers_drunk,
    a ~ dnorm(0, 0.5),
    b_b ~ dnorm(0, 0.5),
    sigma ~ dexp(1)
  ), data = d
)

# Predict chance_of_getting_friends ~ people_talked_to
m5.2.2 <- quap(
  alist(
    chance_of_getting_friends ~ dnorm(mu, sigma),
    mu <- a + b_p * people_talked_to,
    a ~ dnorm(0, 0.5),
    b_p ~ dnorm(0, 0.5),
    sigma ~ dexp(1)
  ), data = d
)

# Predict chance_of_getting_friends ~ beers_drunk + people_talked_to
m5.2.3 <- quap(
  alist(
    chance_of_getting_friends ~ dnorm(mu, sigma),
    mu <- a + b_b * beers_drunk + b_p * people_talked_to,
    a ~ dnorm(0, 0.5),
    b_b ~ dnorm(0, 0.5),
    b_p ~ dnorm(0, 0.5),
    sigma ~ dexp(1)
  ), data = d
)

precis(m5.2.1)
```

```
##              mean         sd        5.5%       94.5%
## a     -0.03439582 0.03543903 -0.09103424  0.02224259
## b_b   -0.24495846 0.02527347 -0.28535035 -0.20456657
## sigma  1.12324973 0.02509691  1.08314002  1.16335945
```

```
precis(m5.2.2)
```

```
##              mean         sd        5.5%      94.5%
## a     -0.03992076 0.03559096 -0.09680199 0.01696046
## b_p    0.23638368 0.02577754  0.19518620 0.27758116
## sigma  1.12833065 0.02520888  1.08804200 1.16861930
```

```
precis(m5.2.3)
```

```
##              mean         sd        5.5%       94.5%
## a     -0.02432584 0.03067434 -0.07334936  0.02469767
## b_b   -0.46559603 0.02495402 -0.50547737 -0.42571469
## b_p    0.46439760 0.02533765  0.42390314  0.50489206
## sigma  0.97146012 0.02170728  0.93676768  1.00615255
```

In this simulation a person spends some amount of time at a party, during that time they both drink beers and talk to people, the more people they talk to the more likely they find new friends, however, when they become more drunk they also decrease the likelihood of making new friends. When you analyze chance_of_getting_friends ~ beers_drunk, you'll see a negative correlation When you analyze chance_of_getting_friends ~ people_talked_to, you'll see a positive correlation When you analyze chance_of_getting_friends ~ beers_drunk + people_talked_to, their individual effects are exaggerated (i.e. overestimated).

5M3. It is sometimes observed that the best predictor of fire risk is the presence of firefighters— States and localities with many firefighters also have more fires. Presumably firefighters do not cause fires. Nevertheless, this is not a spurious correlation. Instead fires cause firefighters. Consider the same reversal of causal inference in the context of the divorce and marriage data. How might a high divorce rate cause a higher marriage rate? Can you think of a way to evaluate this relationship, using multiple regression

People getting divorced would increase the number of single people. With more single people the likelihood of two people finding each other and getting married would increase. The implication is that people get married more than once, thus it could be evaluated by tracking the amount of people getting married for the second time. If that parameter is added the coefficient for divorce should be closer to 0, than when the regression isn't: `marriage ~ a*divorce` -> a is higher `marriage ~ a*divorce + b*rate_of_remarriage*` -> a is closer to 0

5M5. One way to reason through multiple causation hypotheses is to imagine detailed mechanisms through which predictor variables may influence outcomes. For example, it is sometimes argued that the price of gasoline (predictor variable) is positively associated with lower obesity rates (outcome variable). However, there are at least two important mechanisms by which the price of gas could reduce obesity. First, it could lead to less driving and therefore more exercise. Second, it could lead to less driving, which leads to less eating out, which leads to less consumption of huge restaurant meals. Can you outline one or more multiple regressions that address these two mechanisms? Assume you can have any predictor data you need.

Variables: - G = gas price - O = obesity rate - D = driving - E = exercise - R = eating at restaurants

Mutual relationships: As G increases D decreases

Model 1 (exercise): As D decreases E increases As E increases O decreases

Model 2 (restaurants): As D decreases R decreases As R decreases O decreases

This could be modeled as a set of regressions with varying predictors and outcomes i.e. the effect by mediation through each of the routes. The models would be as follows: - D ~ G - E ~ D - R ~ D - O ~ E - O ~ R - O ~ G

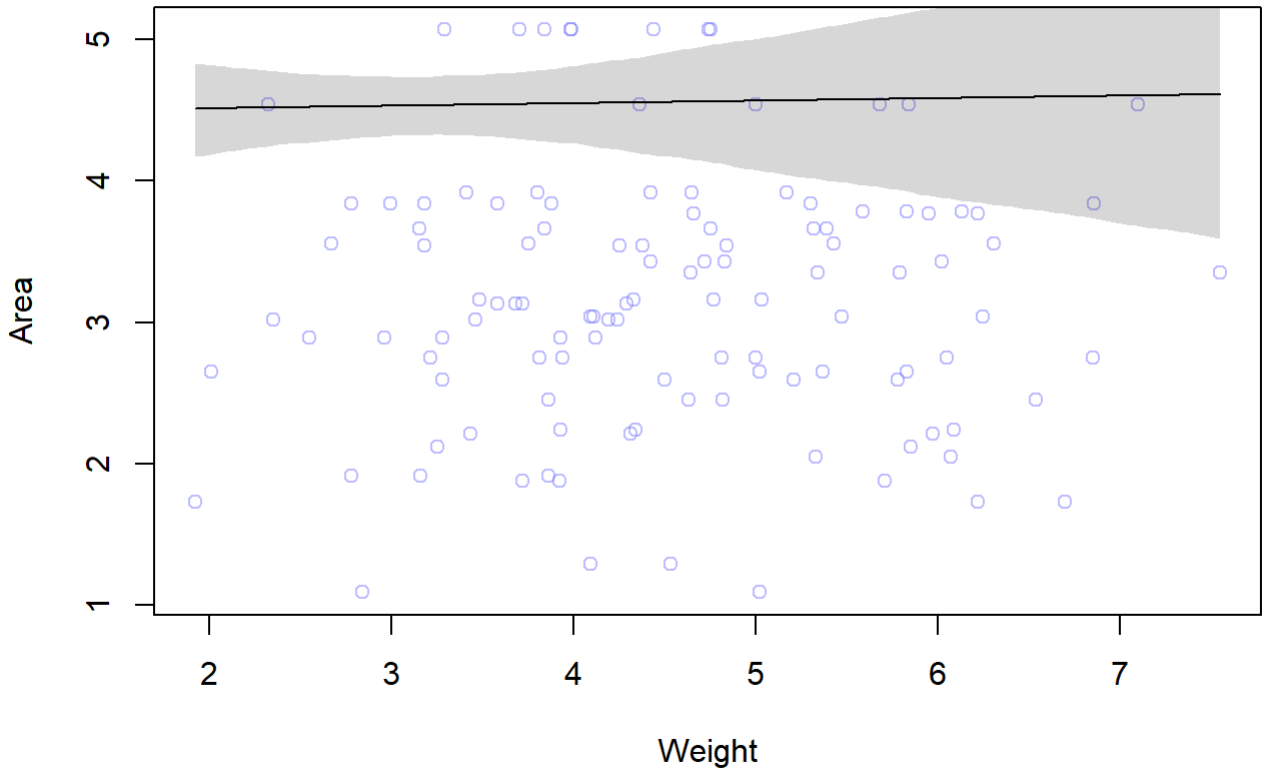
Chapter 5: Foxes and Pack Sizes

All five exercises below use the same data, data(foxes) (part of rethinking).84 The urban fox (Vulpes vulpes) is a successful exploiter of human habitat. Since urban foxes move in packs and defend territories, data on habitat quality and population density is also included. The data frame has five columns: (1) group: Number of the social group the individual fox belongs to (2) avgfood: The average amount of food available in the territory (3) groupsize: The number of foxes in the social group (4) area: Size of the territory (5) weight: Body weight of the individual fox

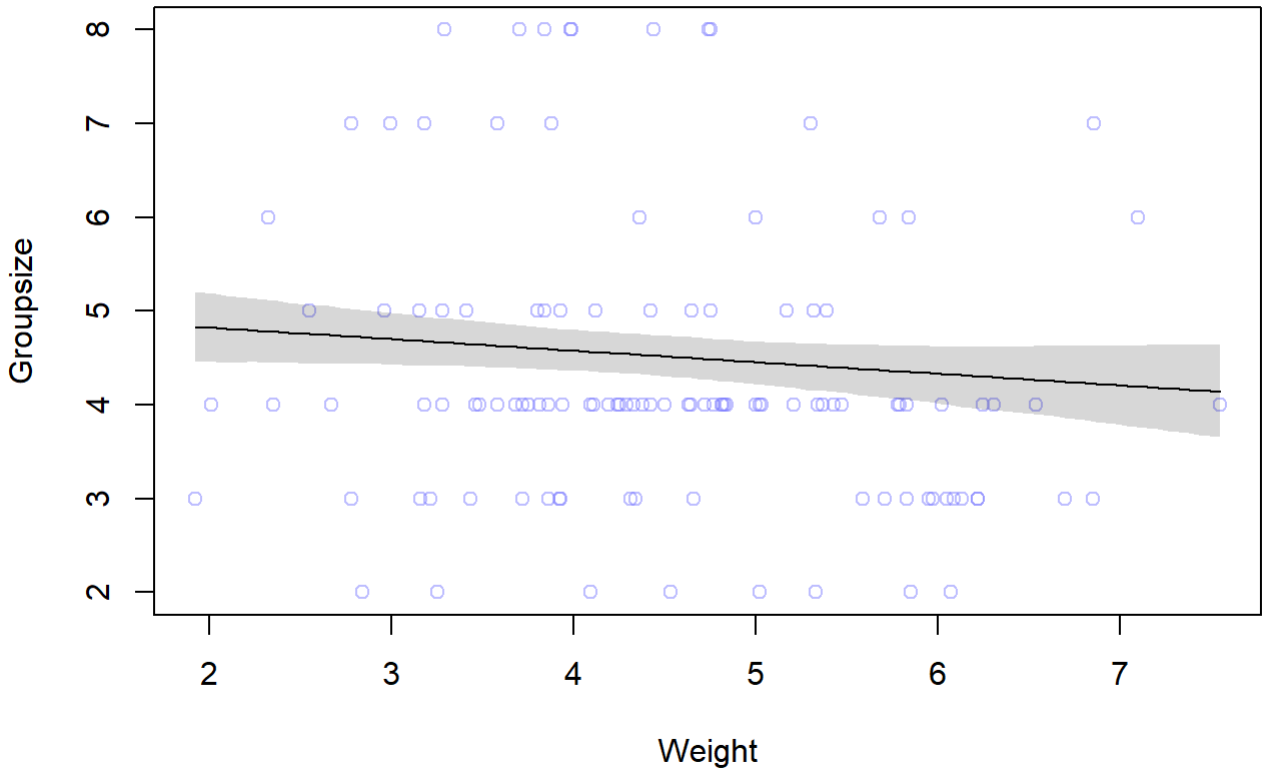
5H1. Fit two bivariate Gaussian regressions, using quap: (1) body weight as a linear function of territory size (area), and (2) body weight as a linear function of groupsize. Plot the results of these regressions, displaying the MAP regression line and the 95% interval of the mean. Is either variable important for predicting fox body weight?

```
data(foxes)

# Model 1: weight ~ area
m5H1.1 <- quap(
  alist(
    weight ~ dnorm(mu, sigma),
    mu <- a + b_a * area,
    a ~ dnorm(7, 3), # Google says foxes weigh 2-14 kg
    b_a ~ dnorm(3, 3), # No idea about area but the data shows it is somewhere between 1 and
5
    sigma ~ dexp(1)
  ), data = foxes
)

m5H1.2 <- quap(
  alist(
    weight ~ dnorm(mu, sigma),
    mu <- a + b_g * groupsize,
    a ~ dnorm(7, 3), # Google says foxes weigh 2-14 kg
    b_g ~ dnorm(6, 3), # No idea about area but the data shows it is somewhere between 2-8
    sigma ~ dexp(1)
  ), data = foxes
)
```

```
# Plot the results
weight.seq <- seq(from = min(foxes$weight), to = max(foxes$weight), length.out = 100)
mu <- link(m5H1.1, data = list(area = weight.seq))
mu.mean <- apply(mu, 2, mean)
mu.PI <- apply(mu, 2, PI, prob = 0.95)

plot(foxes$weight, foxes$area, col = col.alpha(rangi2, 0.5), xlab = "Weight", ylab = "Area")
lines(weight.seq, mu.mean)
shade(mu.PI, weight.seq)
```




```
mu2 <- link(m5H1.2, data = list(groupsize = weight.seq))
mu2.mean <- apply(mu2, 2, mean)
mu2.PI <- apply(mu2, 2, PI, prob = 0.95)

plot(foxes$weight, foxes$groupsize, col = col.alpha(rangi2, 0.5), xlab = "Weight", ylab = "Gr
oupsize")
lines(weight.seq, mu2.mean)
shade(mu2.PI, weight.seq)
```



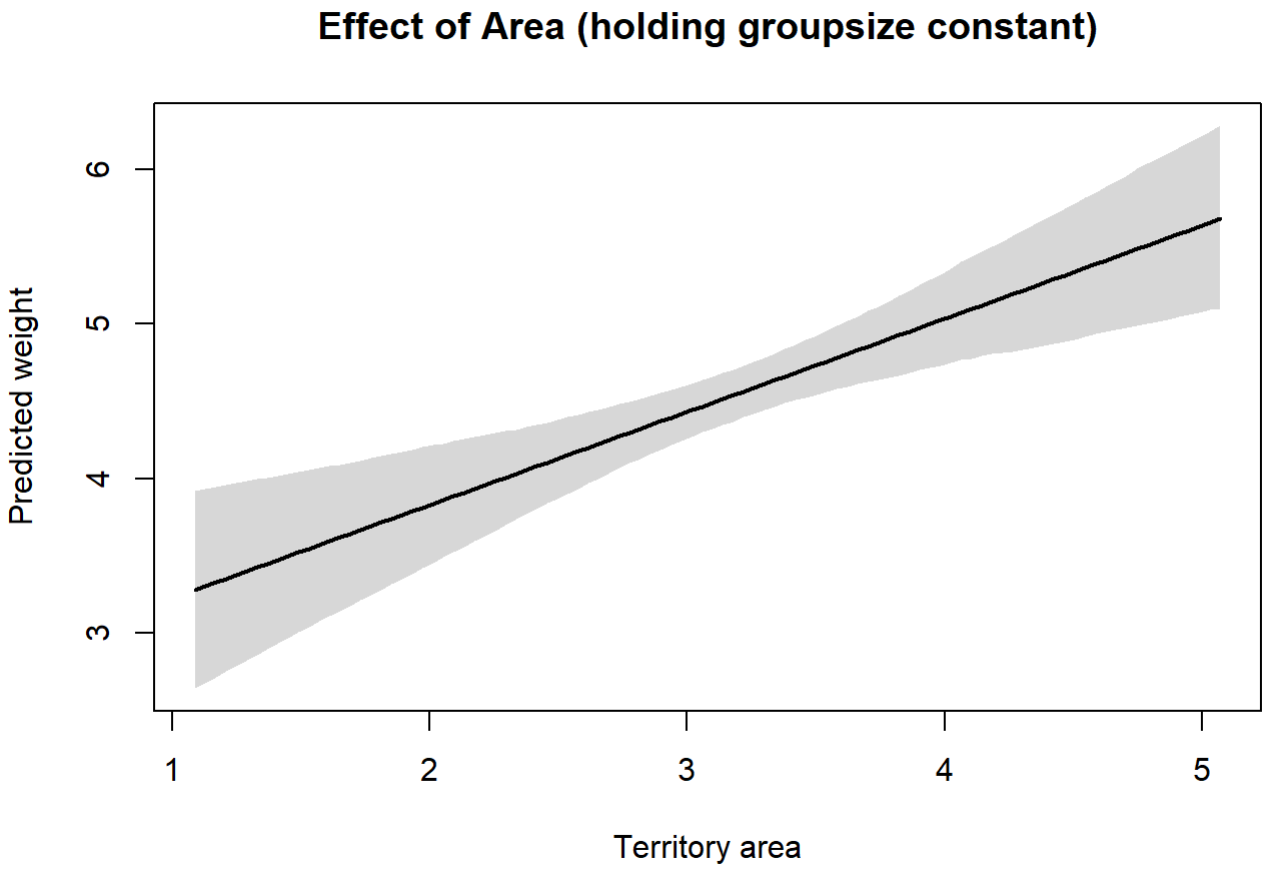
Both regression lines are near horizontal, indicating that neither area nor groupsize are important for predicting fox body weight.

5H2. Now fit a multiple linear regression with weight as the outcome and both area and groupsize as predictor variables. Plot the predictions of the model for each predictor, holding the other predictor constant at its mean. What does this model say about the importance of each variable? Why do you get different results than you got in the exercise just above?

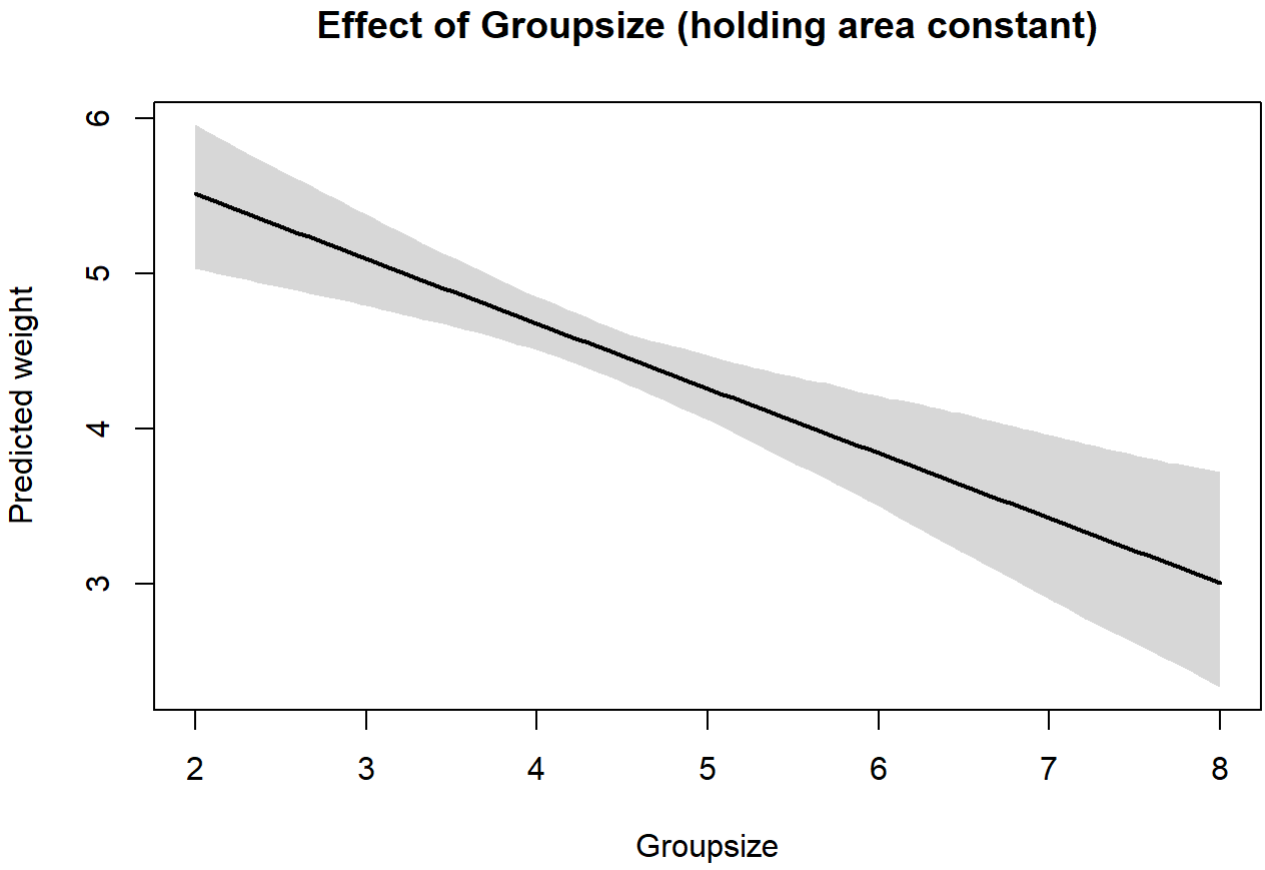
```
m5H2 <- quap(
  alist(
    weight ~ dnorm(mu, sigma),
    mu <- a + b_a * area + b_g * groupsize,
    a ~ dnorm(7, 3), # Google says foxes weigh 2-14 kg
    b_a ~ dnorm(3, 2), # No idea about area but the data shows it is somewhere between 1-5
    b_g ~ dnorm(6, 2), # No idea about area but the data shows it is somewhere between 2-8
    sigma ~ dexp(1)
  ), data = foxes
)

precis(m5H2)
```

```
##             mean         sd       5.5%      94.5%
## a      4.4647943 0.36584850  3.8800977  5.0494908
## b_a    0.5989925 0.19756364  0.2832477  0.9147374
## b_g   -0.4212873 0.11958764 -0.6124114 -0.2301632
## sigma  1.1131729 0.07256927  0.9971932  1.2291527
```

```r
weight.seq <- seq(from = min(foxes$weight), to = max(foxes$weight), length.out = 100)

area_seq <- seq(from = min(foxes$area), to = max(foxes$area), length.out = 100)
groupsize_seq <- seq(from = min(foxes$groupsize), to = max(foxes$groupsize), length.out = 10
0)

pred_area_data <- data.frame(area = area_seq, groupsize = mean(foxes$groupsize))
pred_groupsize_data <- data.frame(area = mean(foxes$area), groupsize = groupsize_seq)

pred_area <- link(m5H2, data = pred_area_data)
pred_groupsize <- link(m5H2, data = pred_groupsize_data)

# Compute mean and intervals from the matrices
pred_area_mu <- apply(pred_area, 2, mean)
pred_area_PI <- apply(pred_area, 2, PI)

pred_groupsize_mu <- apply(pred_groupsize, 2, mean)
pred_groupsize_PI <- apply(pred_groupsize, 2, PI)

plot(NULL, xlim = range(area_seq), ylim = range(pred_area_PI),
     xlab = "Territory area", ylab = "Predicted weight",
     main = "Effect of Area (holding groupsize constant)")
lines(area_seq, pred_area_mu, lwd = 2)
shade(pred_area_PI, area_seq)
```



```
plot(NULL, xlim = range(groupsize_seq), ylim = range(pred_groupsize_PI),
     xlab = "Groupsize", ylab = "Predicted weight",
     main = "Effect of Groupsize (holding area constant)")
lines(groupsize_seq, pred_groupsize_mu, lwd = 2)
shade(pred_groupsize_PI, groupsize_seq)
```



The model shows that the two variables are nearly equally important for predicting fox body weight, however they have inverse effects. The two variables are correlated, so when we counter-factually hold one variable constant (i.e. by stratifying by it), then the other variables influence is not masked by the other variable.

5H3. Finally, consider the avgfood variable. Fit two more multiple regressions: (1) body weight as an additive function of avgfood and groupsize, and (2) body weight as an additive function of all three variables, avgfood and groupsize and area. Compare the results of these models to the previous models you've fit, in the first two exercises. (a) Is avgfood or area a better predictor of body weight? If you had to choose one or the other to include in a model, which would it be? Support your assessment with any tables or plots you choose. (b) When both avgfood or area are in the same model, their effects are reduced (closer to zero) and their standard errors are larger than when they are included in separate models. Can you explain this result?

```
m5H3.1 <- quap(
  alist(
    weight ~ dnorm(mu, sigma),
    mu <- a + b_a * area + b_f * avgfood,
    a ~ dnorm(7, 3), # Google says foxes weigh 2-14 kg
    b_a ~ dnorm(3, 2), # No idea about area but the data shows it is somewhere between 1-5
    b_f ~ dnorm(1, 1), # No idea about area but the data shows it is somewhere between 0-2
    sigma ~ dexp(1)
  ), data = foxes
)

m5H3.2 <- quap(
  alist(
    weight ~ dnorm(mu, sigma),
    mu <- a + b_a * area + b_g * groupsize + b_f * avgfood,
    a ~ dnorm(7, 3), # Google says foxes weigh 2-14 kg
    b_a ~ dnorm(3, 2), # No idea about area but the data shows it is somewhere between 1-5
    b_g ~ dnorm(6, 2), # No idea about area but the data shows it is somewhere between 2-8
    b_f ~ dnorm(1, 1), # No idea about area but the data shows it is somewhere between 0-2
    sigma ~ dexp(1)
  ), data = foxes
)

precis(m5H3.1)
```

```
##               mean         sd       5.5%     94.5%
## a      4.462027223 0.40233516  3.8190179 5.1050365
## b_a    0.024198052 0.18517254 -0.2717434 0.3201395
## b_f   -0.007603531 0.76162255 -1.2248235 1.2096164
## sigma  1.172715229 0.07667106  1.0501801 1.2952504
```

```
precis(m5H3.2)
```

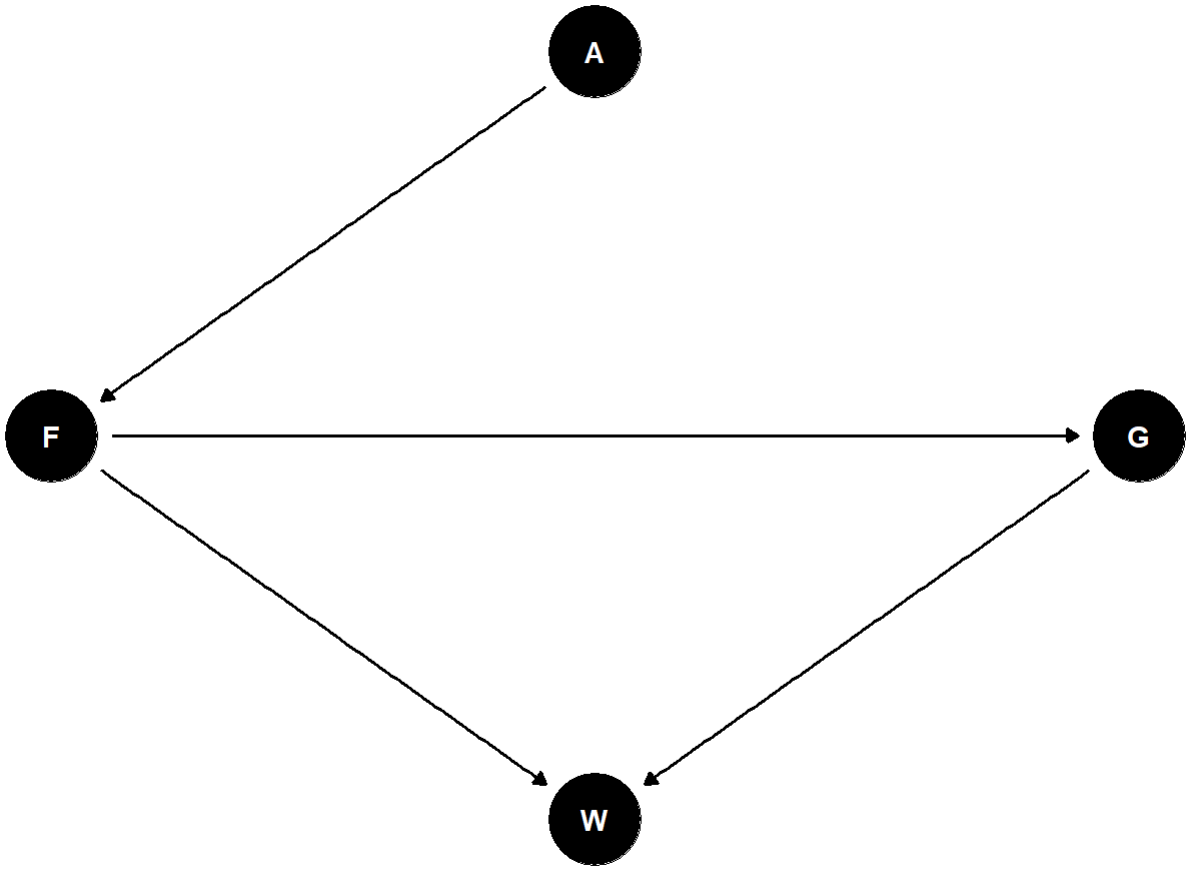
```
##             mean         sd        5.5%      94.5%
## a      4.2603541 0.38352538  3.64740645  4.8733017
## b_a    0.4749473 0.21004709  0.13925153  0.8106432
## b_g   -0.5151520 0.13173697 -0.72569308 -0.3046108
## b_f    1.3376743 0.82305620  0.02227151  2.6530770
## sigma  1.1022623 0.07199118  0.98720645  1.2173181
```

a. We see that in the model weight ~ area + avgfood, both coefficients are roughly 0. This suggests that there is a high correlation between area and avgfood that is causing a masked relationship. The other model weight ~ area + groupsize + avgfood shows shows area and groupsize have coefficient sizes of roughly 0.5 and -0.5 respectively, while avgfood has a coefficient closer to 1.3. The larger magnitude suggest that avgfood is a better predictor of body weight than area. This also assumes that they are on similar scales, as they are not standardized. However an argument could be made that area is a more certain predictor of body weight as its range of estimates are more narrow, so it depends on the interpretation of “better” (i.e. does better mean more certain or larger effect).

b. The standard errors increase because they are correlated and thus have shared variance. When we add a variable that is correlated with another variable, the model has to estimate the effect of both variables at the same time. This leads to larger standard errors because the model is less certain about the individual effects of each variable.

Defining our theory with explicit DAGs Assume this DAG as an causal explanation of fox weight:

```
pacman::p_load(dagitty,
               ggdag)
dag <- dagitty('dag {
A[pos="1.000,0.500"]
F[pos="0.000,0.000"]
G[pos="2.000,0.000"]
W[pos="1.000,-0.500"]
A -> F
F -> G
F -> W
G -> W
}')

# Plot the DAG
ggdag(dag, layout = "circle")+
  theme_dag()
```



where A is area, F is avgfood, G is groupsize, and W is weight.

Using what you know about DAGs from chapter 5 and 6, solve the following three questions:

1. Estimate the total causal influence of A on F. What effect would increasing the area of a territory have on the amount of food inside of it?

```
# Fit a model to estimate the effect of area on avgfood
m5H4.1 <- quap(
  alist(
    avgfood ~ dnorm(mu, sigma),
    mu <- a + b_a * area,
    a ~ dnorm(1, 1), # We know there is likely food where foxes live, so let's bias towards p
ositive.
    b_a ~ dnorm(0, 1),  # We don't really know the effect of area on food.
    sigma ~ dexp(1)
  ), data = foxes
)
precis(m5H4.1)
```

```
##             mean          sd       5.5%     94.5%
## a     0.15471962 0.030691243 0.10566908 0.2037702
## b_a   0.18840282 0.009297335 0.17354388 0.2032618
## sigma 0.09260314 0.006072551 0.08289803 0.1023082
```

Increasing the area of a territory by 1 unit would increase the amount of food by 0.19 units on average (there is almost no uncertainty for this parameter).

2. Infer the **total** causal effect of adding food F to a territory on the weight W of foxes. Can you calculate the causal effect by simulating an intervention on food?

```
# Fit a model to estimate the effect of avgfood on weight
m5H4.2 <- quap(
  alist(
    weight ~ dnorm(mu, sigma),
    mu <- a + b_f * avgfood,
    a ~ dnorm(7, 3), # Google says foxes weigh 2-14 kg
    b_f ~ dnorm(1, 1), # No idea about area but the data shows it is somewhere between 0-2
    sigma ~ dexp(1)
  ), data = foxes
)

precis(m5H4.2)
```

```
##             mean         sd       5.5%     94.5%
## a     4.48059946 0.37648650  3.8789013 5.0822976
## b_f   0.06967324 0.48008841 -0.6976008 0.8369472
## sigma 1.17341861 0.07653886  1.0510947 1.2957425
```

The total effect is uncertain as the MAP is close to zero (0.07), but the estimates vary wildly between -0.7 and 0.84 (at 5.5% and 94% quantiles of estimates). This suggests that the total effect of food on weight is not very strong.

3. Infer the **direct** causal effect of adding food F to a territory on the weight W of foxes. In light of your estimates from this problem and the previous one, what do you think is going on with these foxes?

```
# Fit a model to estimate the direct effect of avgfood on weight
# We stratify by gropusize.

m5H4.3 <- quap(
  alist(
    weight ~ dnorm(mu, sigma),
    mu <- a + b_f * avgfood + b_g * groupsize,
    a ~ dnorm(7, 3), # Google says foxes weigh 2-14 kg
    b_f ~ dnorm(1, 1), # No idea about area but the data shows it is somewhere between 0-2
    b_g ~ dnorm(6, 2), # No idea about area but the data shows it is somewhere between 2-8
    sigma ~ dexp(1)
  ), data = foxes
)
precis(m5H4.3)
```

```
##             mean        sd       5.5%      94.5%
## a      4.5720441 0.3649129  3.9888428  5.1552454
## b_f    1.9870629 0.7768986  0.7454288  3.2286970
## b_g   -0.3528758 0.1130949 -0.5336232 -0.1721283
## sigma  1.1222174 0.0737221  1.0043952  1.2400395
```

The direct effect of food on weight (i.e. stratified by groupsize) is MAP 1.99 (5.5% quantile 0.75, 94% quantile 3.23). This suggests that the direct effect of food on weight is positive.

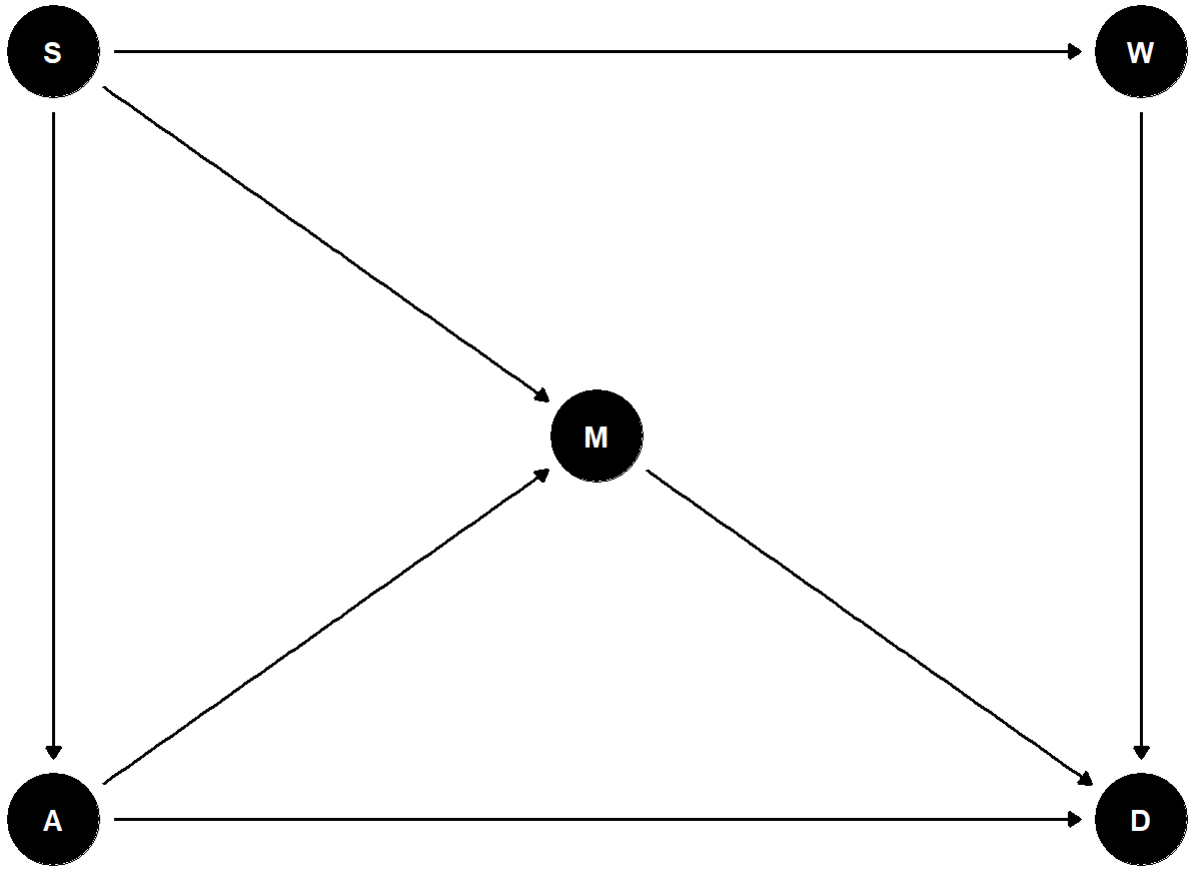
What is likely happening: - More food in a territory directly increases the weight of foxes. - More food also attracts more foxes to the territory (increases groupsize) - Larger groupsize increases competition for food, which decreases the individual weight of foxes. - This nets a near-zero total effect of food on weight.

This relationship is hidden by post-treatment bias, where the groupsize can be considered a post-treatment variable.

Chapter 6: Investigating the Waffles and Divorces

6H1. Use the Waffle House data, data(WaffleDivorce), to find the total causal influence of number of Waffle Houses on divorce rate. Justify your model or models with a causal graph.

```
# Load data and standardize
data(WaffleDivorce)

d <- WaffleDivorce

# Standardize the data
d$D <- standardize(d$Divorce)
d$W <- standardize(d$WaffleHouses)
d$S <- d$South
d$M <- standardize(d$Marriage)
d$A <- standardize(d$MedianAgeMarriage)

# The chapter provides a DAG for this data:
waffle_dag <- dagitty("dag {
  A -> D
  A -> M -> D
  A <- S -> M
  S -> W -> D
}")

# Fix coordinates
coordinates(waffle_dag) <- list(x = c(A = 1, S = 1, M = 2, W = 3, D = 3),
                                y = c(A = 1, S = 3, M = 2, W = 3, D = 1))
# Plot the DAG
ggdag(waffle_dag, layout = "circle") +
  theme_dag()
```



```
# Find the adjustment sets
adjustmentSets(waffle_dag, exposure = "W", outcome = "D")
```

```
## { A, M }
## { S }
```

```
# Model D ~ W + S

m6H1 <- quap(
  alist(
    D ~ dnorm(mu, sigma),
    mu <- a + b_w * W + b_s * S,
    a ~ dnorm(0, 0.2), # Everything is standardized
    b_w ~ dnorm(0, 0.5), # Everything is standardized
    b_s ~ dnorm(0, 10), # Southernness is 0 or 1
    sigma ~ dexp(1) # sd must be positive
  ), data = d
)

precis(m6H1)
```

```
##              mean         sd        5.5%      94.5%
## a     -0.11641964 0.12976423 -0.32380794 0.09096866
## b_w    0.06547832 0.16373439 -0.19620086 0.32715751
## b_s    0.59268706 0.35341487  0.02786184 1.15751228
## sigma  0.92245406 0.09123211  0.77664754 1.06826059
```

WaffleHouses has MAP 0.07 (-0.20 to 0.33 at 5.5% and 94.5% quantiles). This suggests that the total effect of WaffleHouses on Divorce is roughly zero, however with uncertainty to both positive and negative sides. Most of the variance is taken up by being in the south (MAP 0.59, 0.03 to 0.1.16 at 5.5% and 94.5% quantiles).

6H2. Build a series of models to test the implied conditional independencies of the causal graph you used in the previous problem. If any of the tests fail, how do you think the graph needs to be amended? Does the graph need more or fewer arrows? Feel free to nominate variables that aren't in the data.

```
# Get the implied conditional independencies
impliedConditionalIndependencies(waffle_dag)
```

```
## A _||_ W | S
## D _||_ S | A, M, W
## M _||_ W | S
```

```
# Test A _||_ W | S
# A ~ S
m6H2.1 <- quap(
  alist(
    A ~ dnorm(mu, sigma),
    mu <- a + b_s * S,
    a ~ dnorm(0, 0.2), # Everything is standardized
    b_s ~ dnorm(0, 10), # Southernness is 0 or 1
    sigma ~ dexp(1) # sd must be positive
  ), data = d
)

# A ~ S + W
m6H2.2 <- quap(
  alist(
    A ~ dnorm(mu, sigma),
    mu <- a + b_s * S + b_w * W,
    a ~ dnorm(0, 0.2), # Everything is standardized
    b_s ~ dnorm(0, 10), # Southernness is 0 or 1
    b_w ~ dnorm(0, 0.5), # Everything is standardized
    sigma ~ dexp(1) # sd must be positive
  ), data = d
)

precis(m6H2.1)
```

```
##              mean         sd       5.5%       94.5%
## a      0.09402414 0.12444367 -0.1048609  0.29290917
## b_s   -0.48763510 0.28296387 -0.9398660 -0.03540418
## sigma  0.95132046 0.09395967  0.8011548  1.10148615
```

```
precis(m6H2.2)
```

```
##              mean         sd       5.5%     94.5%
## a      0.10740133 0.13190941 -0.1034154 0.3182180
## b_s   -0.55656645 0.36220669 -1.1354427 0.0223098
## b_w    0.05117944 0.16785458 -0.2170846 0.3194435
## sigma  0.94973022 0.09389454  0.7996686 1.0997918
```

We see that including W in the model slightly lowers the MAP of S from -0.49 to -0.56 while the MAP of W is 0.05 but with a large uncertainty (-0.22 to 0.32 at 5.5% and 94.5% quantiles). This suggests that the model is not very sensitive to the inclusion of W, and thus the independence assumption mostly holds.

```
# Test D _||_ S | A, M, W

# D ~ A + M + W
m6H2.3 <- quap(
  alist(
    D ~ dnorm(mu, sigma),
    mu <- a + b_a * A + b_m * M + b_w * W,
    a ~ dnorm(0, 0.2), # Everything is standardized
    b_a ~ dnorm(0, 0.5), # Everything is standardized
    b_m ~ dnorm(0, 0.5), # Everything is standardized
    b_w ~ dnorm(0, 0.5), # Everything is standardized
    sigma ~ dexp(1) # sd must be positive
  ), data = d
)

# D ~ A + M + W + S
m6H2.4 <- quap(
  alist(
    D ~ dnorm(mu, sigma),
    mu <- a + b_a * A + b_m * M + b_w * W + b_s * S,
    a ~ dnorm(0, 0.2), # Everything is standardized
    b_a ~ dnorm(0, 0.5), # Everything is standardized
    b_m ~ dnorm(0, 0.5), # Everything is standardized
    b_w ~ dnorm(0, 0.5), # Everything is standardized
    b_s ~ dnorm(0, 10), # Southernness is 0 or 1
    sigma ~ dexp(1) # sd must be positive
  ), data = d
)


precis(m6H2.3)
```

```
##                mean         sd         5.5%      94.5%
## a     -1.518027e-07 0.09516373 -0.152090165  0.1520899
## b_a   -5.843604e-01 0.14882364 -0.822209349 -0.3465115
## b_m   -5.008810e-02 0.14775355 -0.286226800  0.1860506
## b_w    1.782119e-01 0.10774095  0.006021048  0.3504027
## sigma  7.650664e-01 0.07584178  0.643856625  0.8862763
```

```
precis(m6H2.4)
```

```
##              mean        sd       5.5%      94.5%
## a     -0.06491037 0.1158628 -0.2500815  0.1202608
## b_a   -0.55349482 0.1513294 -0.7953485 -0.3116412
## b_m   -0.03723533 0.1473700 -0.2727611  0.1982905
## b_w    0.09274326 0.1385417 -0.1286731  0.3141596
## b_s    0.29863863 0.3075471 -0.1928810  0.7901582
## sigma  0.75907132 0.0752610  0.6387897  0.8793529
```

We see that including S in the model lowers the MAP of W from 0.18 to 0.09. The rest other estimates hardly change. The MAP of S is 0.30 but with a large uncertainty (-0.19 to 0.79 at 5.5% and 94.5% quantiles). This suggests that the model is sensitive to the inclusion of S, and thus the independence assumption does not hold.

```
# Test M _||_ W | S

# M ~ S
m6H2.5 <- quap(
  alist(
    M ~ dnorm(mu, sigma),
    mu <- a + b_s * S,
    a ~ dnorm(0, 0.2), # Everything is standardized
    b_s ~ dnorm(0, 10), # Southernness is 0 or 1
    sigma ~ dexp(1) # sd must be positive
  ), data = d
)

# M ~ S + W
m6H2.6 <- quap(
  alist(
    M ~ dnorm(mu, sigma),
    mu <- a + b_s * S + b_w * W,
    a ~ dnorm(0, 0.2), # Everything is standardized
    b_s ~ dnorm(0, 10), # Southernness is 0 or 1
    b_w ~ dnorm(0, 0.5), # Everything is standardized
    sigma ~ dexp(1) # sd must be positive
  ), data = d
)

precis(m6H2.5)
```

```
##              mean        sd       5.5%     94.5%
## a     -0.03040034 0.1263055 -0.2322609 0.1714602
## b_s    0.15984272 0.2899998 -0.3036329 0.6233183
## sigma  0.97724357 0.0963346  0.8232823 1.1312049
```

```
precis(m6H2.6)
```

```
##              mean         sd       5.5%     94.5%
## a     -0.03910306 0.13364939 -0.2527006 0.1744945
## b_s    0.20627124 0.37046632 -0.3858055 0.7983480
## b_w   -0.03439813 0.17176997 -0.3089197 0.2401235
## sigma  0.97670649 0.09630719  0.8227890 1.1306240
```

We see that including W in the model leads to a small increase on the MAP of S from 0.16 to 0.21. The MAP for W is -0.03, however for both S and W the uncertainty is quite large (-0.39 to 0.80 and -0.31 to 0.24 respectively at 5.5% and 94.5% quantiles). This suggests that the model is not very sensitive to the inclusion of W, and thus the independence assumption mostly holds.

These three tests of conditional independencies suggest that we should reconsider the DAG. Especially the independence assumption of D || S | A, M, W. This suggests there are other confounding variables between divorce and being in the Southern USA. One major unobserved confounder could be religion, which is prevalent in that region of the USA, while also having specific values regarding marriage/divorce tied in. Generally adding nodes and thus more arrows to a DAG is likely to make it more realistic with regard to the causal relationships (if done with some insight, not blindly/naively). This risk sacrificing practicality and explainability to gain realism.