chapter5_solutions

Easy

5E1. (2) and (4) are MLRs

5E2. Animal Diversity ~ alpha + beta x plant_diversity + beta2 x (latitude | plant_diversity)

5E3. Time To Degree ~ alpha + beta x funding + beta2 x lab_size. Beta and Beta2 > 0, because the problem specifies that they are *both positively associated with time to degree.*

5E4. models 1 and 3 are inferentially equivalent.

Medium.

5M1. Invent your own example of spurious correlation, where an outcome is correlated with both predictor variables, but when both predictor variables are included to the same model, the correlation between the outcome and one of the predictors should mostly vanish.

Suppose you are a basketball coach interested in evaluating player performance. Suppose your outcome variable of interest is a players points-per-game (PPG). It has been shown that a player's PPG is influenced both by their shots-per-game (SPG) as well as by their minutes-per-game (MPG). Players who get a lot of minutes tend also to take a lot of shots, and thereby score a lot of points. Alternatively, players who take a lot of shots are more likely to score more points. Hence both SPG and MPG are positively correlated with PPG. However, it is unlikely that a player could have high SPG numbers with low MPG numbers – this would be like a player who shoots it literally every time they touch the ball. So the effect of MPG on PPG would likely fall out when you also know a player's SPG stats.

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.

Consider rowing data of individual height, weight, and rowing speeds. Taller, lighter rowers are faster than shorter, heavier ones. But it's important to consider weight relative to height, since taller individuals are heavier than shorter individuals.

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?

With more people getting divorced, you will have more single people who may then get re-married. You could evaluate this by fitting a model to predict *marriage rate* using *re-marriage rate and divorce rate.* If divorce rate truly caused a higher marriage rate by increasing the rate of re-marriage, then the multiple linear regression model would tell us that re-marriage rates are more strongly associated with the marriage rate than divorce rate.

5M4. In the divorce data, States with high numbers of Mormons (members of The Church of Jesus Christ of Latter-day Saints, LDS) have much lower divorce rates than the regression models expected. Find a list of LDS population by State and use those numbers as a predictor variable, predicting divorce rate using

marriage rate, median age at marriage, and percent LDS population (possibly standardized). You may want to consider transformations of the raw percent LDS variable.

```
# might skip this for now...
```

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.

Gas = beta x obesity_rate + alpha

However higher gas prices could lead to more walking and less eating out. You could do a MLR to model the effect of exercise and eating out on obesity (to show that obesity is influenced by both). Then you could add in the gasoline variable to see if there's any difference in the ability to predict obesity by knowing gas prices, after you already know the trends in exercise and eating out.

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?

Some weird things going on with the code here... it wasn't working when I had "a" instead of "alpha" but when I changed the variable name to "a" I not longer got this error: `Error in eval(parse(text = lm), envir = e) : object 'a' not found`... literally no idea.

```
library(rethinking)
```

```
## Loading required package: rstan
```

```
## Loading required package: StanHeaders
```

```
## Loading required package: ggplot2
```

```
## rstan (Version 2.21.2, GitRev: 2e1f913d3ca3)
```

```
## For execution on a local, multicore CPU with excess RAM we recommend calling
## options(mc.cores = parallel::detectCores()).
## To avoid recompilation of unchanged Stan programs, we recommend calling
## rstan_options(auto_write = TRUE)
```

```
## Loading required package: parallel
```

```
## rethinking (Version 2.13)
```

```
##
## Attaching package: 'rethinking'
```

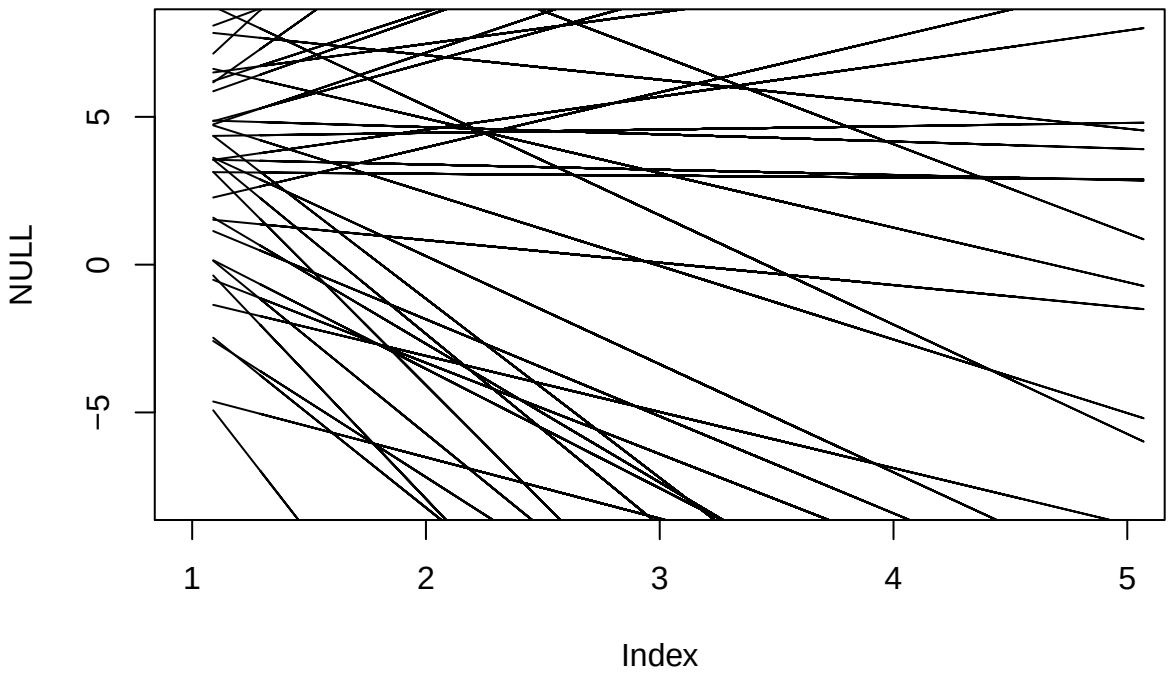
```
## The following object is masked from 'package:stats':
##
##     rstudent
```

```
data(foxes)
d <- foxes

#d$a <- scale(d$area)
#d$g_scale <- scale(d$groupsize)
```

```
# body weight as a linear function of area
flist <- alist(weight ~ dnorm(mu, sigma),
               mu <- alpha + b*area,
               alpha ~ dnorm(5, 5),
               b ~ dnorm(0, 5),
               sigma ~ dunif(0, 5))

area_model <- quap(flist, d)

# body weight as a linear function of groupsize
flist <- alist(weight ~ dnorm(mu, sigma),
               mu <-  alpha + b*groupsize,
               alpha ~ dnorm(2, 1),
               b ~ dnorm(2, 1),
               sigma ~ dnorm(1, 1))

size_model <- quap(flist, d)
```

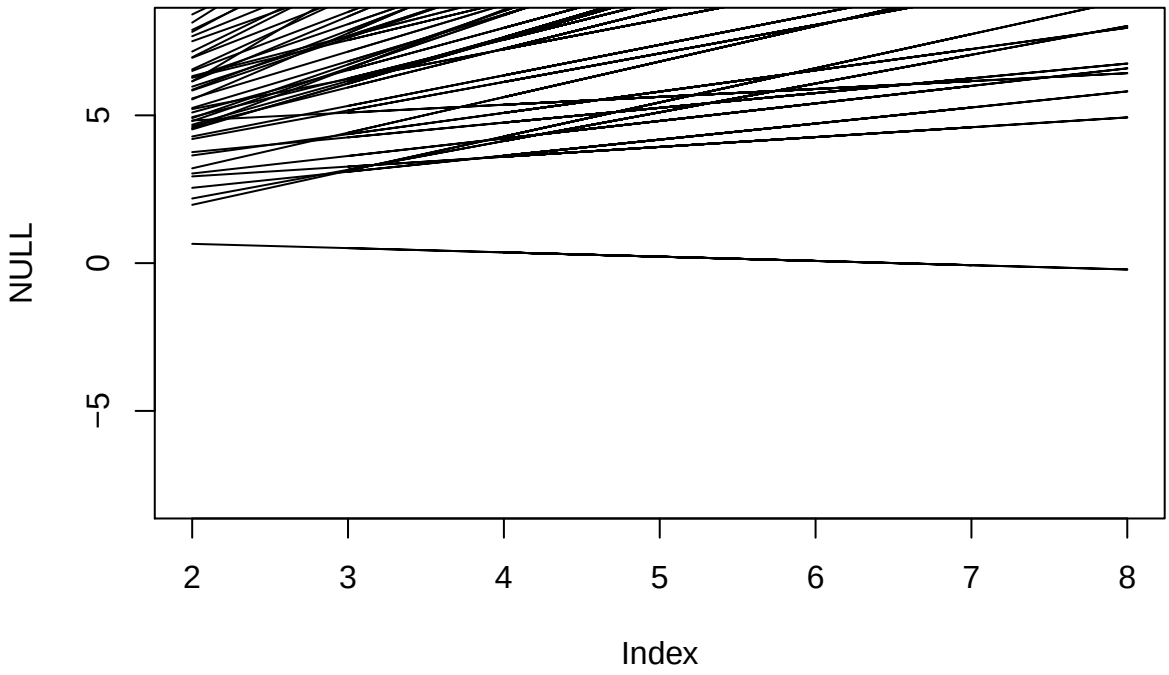
Run a quick prior predictive check for these arbitrarily and hastily chosen priors:

```
prior <- extract.prior(area_model)
xseq <- c(1, 5)
mu <- link(area_model, post = prior)
plot(NULL, xlim = xseq, ylim = c(-8, 8))
for (i in 1:50) lines(d$area, mu[i,])
```



The priors I chose are kind of crazy..

```
prior <- extract.prior(size_model)
xseq <- c(2, 8)
mu <- link(size_model, post = prior)
plot(NULL, xlim = xseq, ylim = c(-8, 8))
for (i in 1:50) lines(d$groupsize, mu[i,])
```

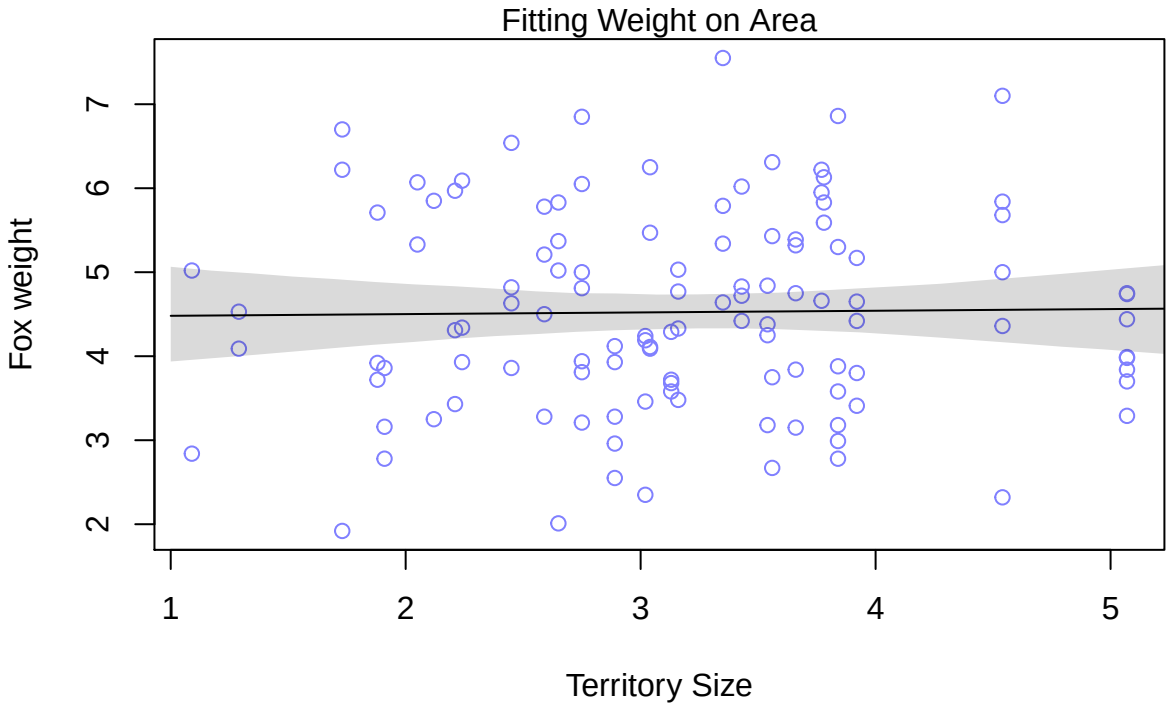


Yeah these priors are complete madness.

```
# area model
territory_size = seq(from = 1, to = 6, length.out = 30)
bunch_of_weights = link(area_model, data = list(area = territory_size)) # map regression line

average_predicted_weight = apply(bunch_of_weights, 2, mean) #map regression line (mean weight)
weight_pi = apply(bunch_of_weights, 2, PI, prob = .95) # 95% interval of mean
```

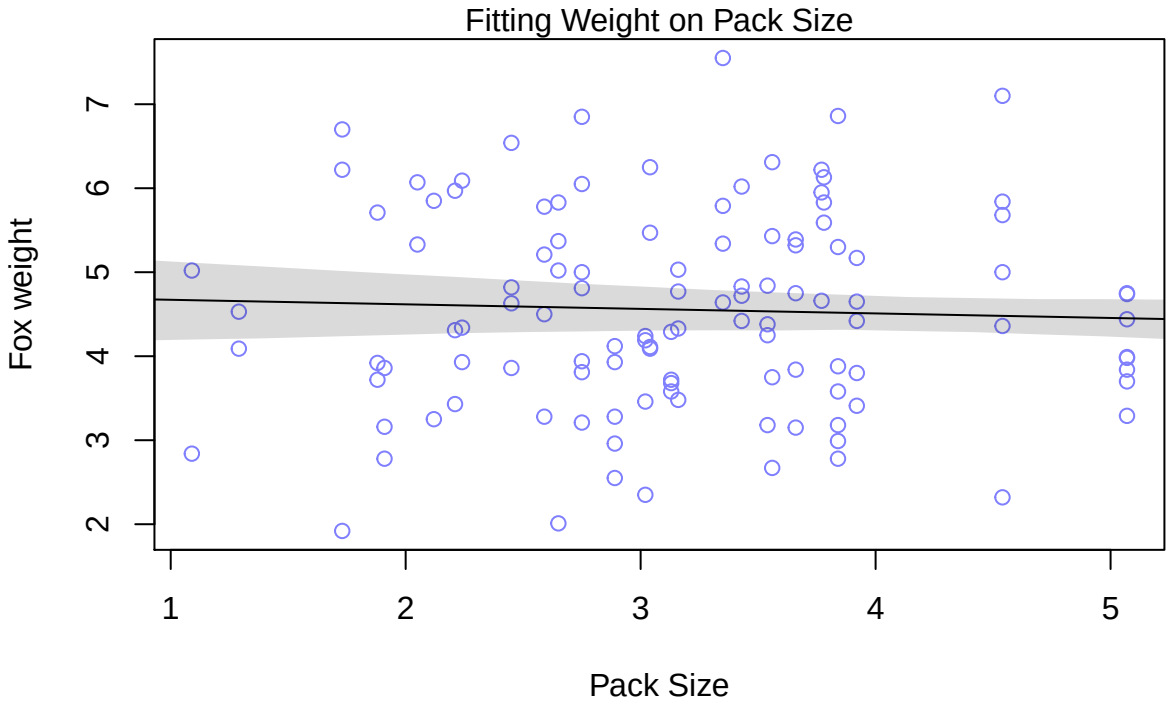
Visualize.

```
# display raw data and sample size
plot( d$area , d$weight ,
xlim=range(d$area) , ylim=range(d$weight) ,
col=rangi2 , xlab="Territory Size" , ylab="Fox weight" )
mtext('Fitting Weight on Area')

lines(territory_size, average_predicted_weight, type = 'l')
shade(weight_pi, territory_size)
```




```
# size model
pack_size = seq(from = 0, to = 8, length.out = 30)
bunch_of_weights = link(size_model, data = list(groupsize = pack_size)) # map regression line

average_predicted_weight = apply(bunch_of_weights, 2, mean) #map regression line (mean weight)
weight_pi = apply(bunch_of_weights, 2, PI, prob = .95) # 95% interval of mean
```

```
# display raw data and sample size
plot( d$area , d$weight ,
xlim=range(d$area) , ylim=range(d$weight) ,
col=rangi2 , xlab="Pack Size" , ylab="Fox weight" )
mtext('Fitting Weight on Pack Size')

lines(pack_size, average_predicted_weight, type = 'l', ylim = c(0, 8))
shade(weight_pi, pack_size)
```



Neither area nor size seem to be all that strongly correlated with a fox's 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?

```
flist <- alist(weight ~ dnorm(mu, sigma),
               mu ~ alpha + r*area + g*groupsize,
               alpha ~ dnorm(1,1),
               r ~ dnorm(1,1),
               g ~ dnorm(1,1),
               sigma ~ dunif(0,5))

mlr <- quap(flist, d)
precis(mlr)
```

```
##              mean         sd       5.5%      94.5%
## alpha  4.0188629 0.35133613  3.4573599  4.5803659
## r      0.7229732 0.19333328  0.4139893  1.0319572
## g     -0.4173610 0.11901654 -0.6075724 -0.2271495
## sigma  1.1250536 0.07462954  1.0057812  1.2443260
```

```
# coefficient for area increases more than 3x
# coefficient for size increases more than 20x....
```

```
precis(area_model)
```

```
##              mean         sd       5.5%     94.5%
## alpha 4.45430644 0.38955764  3.8317181 5.0768948
## b     0.02385827 0.11803093 -0.1647779 0.2124945
## sigma 1.17868542 0.07738436  1.0550103 1.3023606
```

```
precis(size_model)
```

```
##              mean        sd       5.5%      94.5%
## alpha  4.73264323 0.3119680  4.2340581 5.23122840
## b     -0.05411473 0.0680455 -0.1628646 0.05463512
## sigma  1.16788589 0.0769852  1.0448487 1.29092310
```

If you're like me, then "plot the prediction of each predictor, holding the other predictor constant at its mean" doesn't make a whole lot of sense initially... But judging from the content of the chapter it seems like it is asking for us to create a counterfactural plot. As a reminder, the steps to build a counterfactural are:

- pick a variable to manipulate, the intervention variable
- define the range of values to set the intervention variable to
- for each value of the intervention, use the model to simulate the values of the other variables (inlcuding outcome)
- We'll need to figure out how to "hold the other predictor constant at its mean"

```
# let's choose area as the intervention, leaving groupsize and weight as the others

# since area ranges from 1.09 to 5.07
area_range <- seq(from = 1, to = 5, length.out = 30)

# we probably should calculate the mean of groupsize now
group_mean = mean(d$groupsize)
```

I think what the problem is asking for is: what does the model predict for weight if we use the raw area data and control for groupsize. This means we need to predict weight using the model defined as is.

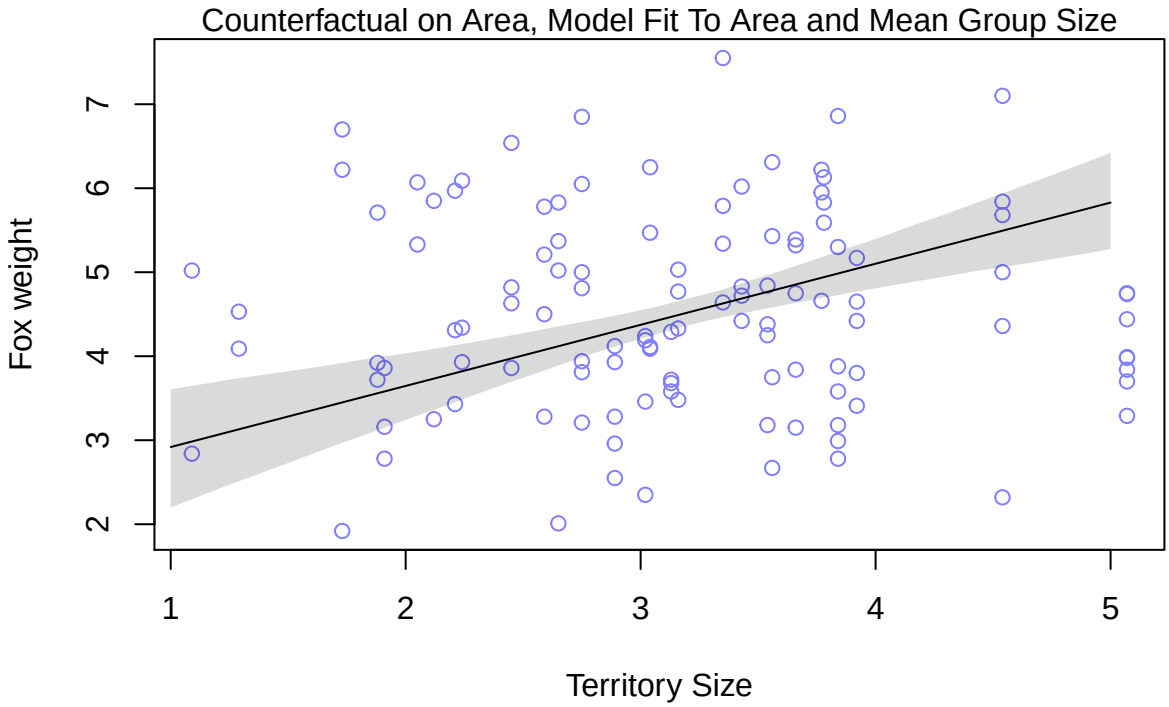
```
# create new dataframe for predicting weight
library(data.table)
area_seq = seq(1, 5, length.out = 30)
new_df = data.table(area = area_seq, groupsize = group_mean)

# predict weights using our model and this new data
weight_preds = link(mlr, data = new_df)
```

Now we have a list of 1,000 predicted weights for each unique value for area. To get the prediction for each area we just have to now average those 1,000 weight preds across area values.

```
mu = apply(weight_preds, 2, mean)

# PI around mean if we want it
mu_pi = apply(weight_preds, 2, PI)
```

Visualize these predictions:

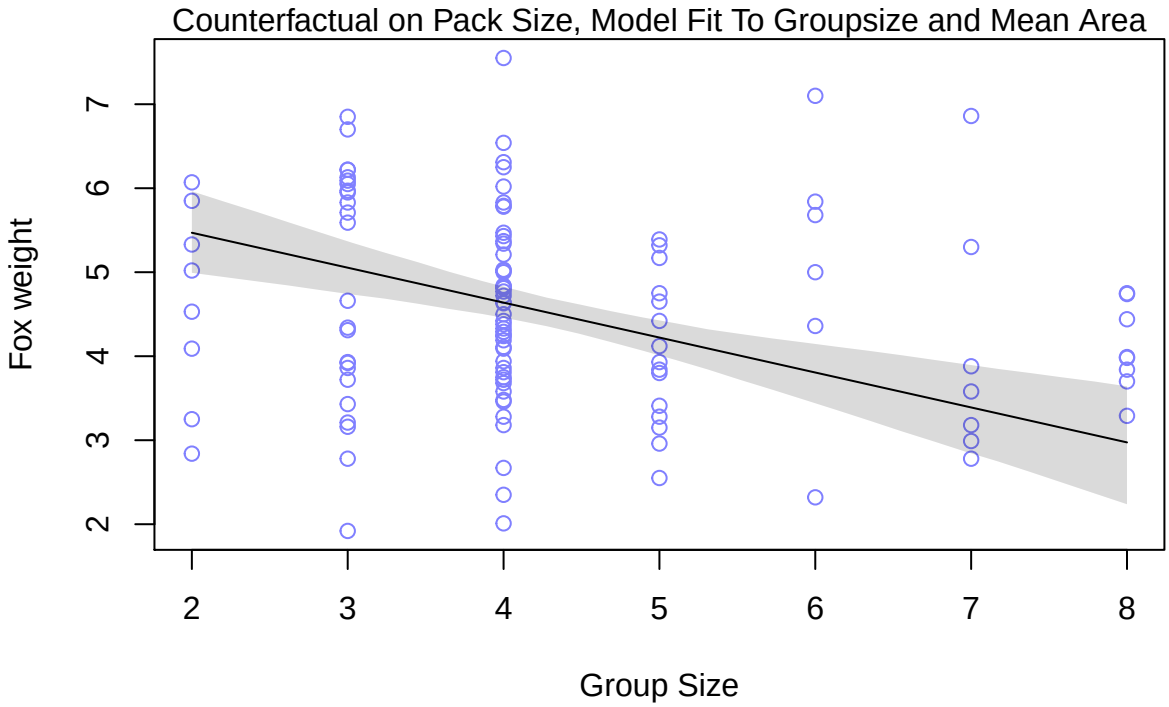
```
# display raw data and sample size
plot( d$area , d$weight ,
xlim=range(d$area) , ylim=range(d$weight) ,
col=rangi2 , xlab="Territory Size" , ylab="Fox weight" )
mtext('Counterfactual on Area, Model Fit To Area and Mean Group Size')

lines(area_seq, mu, type = 'l', ylim = c(0, 8))
shade(mu_pi, area_seq)
```



Using same approach, we can build a counterfactual for groupsize.

```
group_seq = seq(2, 8, length.out = 30)
new_df = data.table(groupsize = group_seq, area = mean(d$area))

# predict weights using our model and this new data
weight_preds = link(mlr, data = new_df)


mu = apply(weight_preds, 2, mean)

# PI around mean if we want it
mu_pi = apply(weight_preds, 2, PI)
```

```
# display raw data and sample size
plot( d$groupsize , d$weight ,
xlim=range(d$groupsize) , ylim=range(d$weight) ,
col=rangi2 , xlab="Group Size" , ylab="Fox weight" )
mtext('Counterfactual on Pack Size, Model Fit To Groupsize and Mean Area')


lines(group_seq, mu, type = 'l', ylim = c(0, 8))
shade(mu_pi, group_seq)
```



This result tells us that bodyweight is negatively correlated with groupsize, and positively correlated with area.

```
cor(d$area, d$groupsize)
```

```
## [1] 0.8275945
```

Because area and groupsize are positively correlated, and have opposite influences on weight... I'm struggling to explain this one.

#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.

```
flist <- alist(weight ~ dnorm(mu, sigma),
               mu <- alpha + food * avgfood + grp * groupsize,
               alpha ~ dnorm(1,1),
               food ~ dnorm(0,5),
               grp ~ dnorm(0,5),
               sigma ~ dnorm(1,.4))

food_pack <- quap(flist, d)


flist <- alist(weight ~ dnorm(mu, sigma),
               mu <- alpha + territory * area + grp * groupsize,
               alpha ~ dnorm(1,1),
               territory ~ dnorm(1,1),
               grp ~ dnorm(1,1),
               sigma ~ dnorm(1,.4))

area_pack <- quap(flist, d)
```

```
flist <- alist(weight ~ dnorm(mu, sigma),
               mu <- alpha + food * avgfood + grp * groupsize + territory * area,
               alpha ~ dnorm(1,1),
               food ~ dnorm(0, 5),
               grp ~ dnorm(0,5),
               territory ~ dnorm(1,1),
               sigma ~ dnorm(1,.4))

food_pack_area <- quap(flist, d)
```

```
precis(food_pack)
```

```
##             mean         sd       5.5%      94.5%
## alpha  3.6866866 0.39536842  3.0548115  4.3185617
## food   4.5337677 1.13856611  2.7141192  6.3534163
## grp   -0.5970555 0.15176994 -0.8396132 -0.3544978
## sigma  1.1176722 0.07232798  1.0020781  1.2332662
```

```
precis(area_pack)
```

```
##                 mean         sd       5.5%      94.5%
## alpha      4.0216417 0.35002763  3.4622299  4.5810535
## territory  0.7223367 0.19265367  0.4144390  1.0302345
## grp       -0.4174842 0.11858978 -0.6070136 -0.2279549
## sigma      1.1208621 0.07270214  1.0046700  1.2370541
```

Investigate Model 1: avg food + groupsize

Average food seems strongly positively correlated with weight. The effect of groupsize in this model (without area) is smaller than in the MLR (-.34 compared to -.05). So avg food is almost as effect as area at removing the masked relationship of groupsize.

```
precis(mlr)
```

```
##             mean         sd       5.5%      94.5%
## alpha  4.0188629 0.35133613  3.4573599  4.5803659
## r      0.7229732 0.19333328  0.4139893  1.0319572
## g     -0.4173610 0.11901654 -0.6075724 -0.2271495
## sigma  1.1250536 0.07462954  1.0057812  1.2443260
```

```
precis(size_model)
```

```
##              mean        sd       5.5%      94.5%
## alpha  4.73264323 0.3119680  4.2340581 5.23122840
## b     -0.05411473 0.0680455 -0.1628646 0.05463512
## sigma  1.16788589 0.0769852  1.0448487 1.29092310
```

Investigate Model 2: avg food + groupsize + area

```
precis(food_pack_area)
```

```
##                 mean         sd        5.5%      94.5%
## alpha      3.6153982 0.39426616  2.98528472  4.2455117
## food       2.9984620 1.35863609  0.82709916  5.1698249
## grp       -0.6521188 0.15286940 -0.89643363 -0.4078040
## territory  0.4625780 0.23037595  0.09439271  0.8307632
```

```
## sigma      1.1063555 0.07172096  0.99173154  1.2209794
```

The effect of groupsize increases by .22 when we add area, but the effect of area decreases .16 compared to the model on group + area alone. This suggests that avg food contains some of the information that area has. The coefficient for avg food also dropped from 2.5 to 1.67.

(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.

Essentially this is saying is a model fit on weight ~ food + group better or worse than a model fit on weight ~ area + group.

###(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?

Since both of these variables are positively correlated