Queller-Phil-ADA-homework-3

Phil Queller

4/26/2020

# Challenge 1

#read in data  
  
k <-read.csv("~/Desktop/KamilarDATA.csv")  
head(k)

## Scientific\_Name Family Genus Species  
## 1 Allenopithecus\_nigroviridis Cercopithecidae Allenopithecus nigroviridis  
## 2 Allocebus\_trichotis Cercopithecidae Allocebus trichotis  
## 3 Alouatta\_belzebul Atelidae Alouatta belzebul  
## 4 Alouatta\_caraya Atelidae Alouatta caraya  
## 5 Alouatta\_guariba Atelidae Alouatta guariba  
## 6 Alouatta\_palliata Atelidae Alouatta palliata  
## Brain\_Size\_Species\_Mean Brain\_Size\_Female\_Mean Brain\_size\_Ref  
## 1 58.02 53.70 Isler et al 2008  
## 2 NA NA   
## 3 52.84 51.19 Isler et al 2008  
## 4 52.63 47.80 Isler et al 2008  
## 5 51.70 49.08 Isler et al 2008  
## 6 49.88 48.04 Isler et al 2008  
## Body\_mass\_male\_mean Body\_mass\_female\_mean Mass\_Dimorphism  
## 1 6130 3180 1.928  
## 2 92 84 1.095  
## 3 7270 5520 1.317  
## 4 6525 4240 1.539  
## 5 5800 4550 1.275  
## 6 7150 5350 1.336  
## Mass\_Ref MeanGroupSize AdultMales AdultFemale AdultSexRatio  
## 1 Isler et al 2008 NA NA NA NA  
## 2 Smith and Jungers 1997 1.00 1.00 1.0 NA  
## 3 Isler et al 2008 7.00 1.00 1.0 1.00  
## 4 Isler et al 2008 8.00 2.30 3.3 1.43  
## 5 Isler et al 2008 6.53 1.37 2.2 1.61  
## 6 Isler et al 2008 12.00 2.90 6.3 2.17  
## Social\_Organization\_Ref  
## 1   
## 2 Kappeler 1997  
## 3 Campbell et al 2007  
## 4 van Schaik et al. 1999; Kappeler and Pereira 2003; Nunn & van Schaik 2000  
## 5 Campbell et al 2007  
## 6 van Schaik et al. 1999; Kappeler and Pereira 2003; Nunn & van Schaik 2000  
## InterbirthInterval\_d Gestation WeaningAge\_d MaxLongevity\_m LitterSz  
## 1 NA NA 106.15 276.0 1.01  
## 2 NA NA NA NA 1.00  
## 3 NA NA NA NA NA  
## 4 337.62 187 323.16 243.6 1.01  
## 5 NA NA NA NA NA  
## 6 684.37 186 495.60 300.0 1.02  
## Life\_History\_Ref GR\_MidRangeLat\_dd Precip\_Mean\_mm Temp\_Mean\_degC AET\_Mean\_mm  
## 1 Jones et al. 2009 -0.17 1574.0 25.2 1517.8  
## 2 -16.59 1902.3 20.3 1388.2  
## 3 -6.80 1643.5 24.9 1286.6  
## 4 Jones et al. 2009 -20.34 1166.4 22.9 1193.1  
## 5 -21.13 1332.3 19.6 1225.7  
## 6 Jones et al. 2009 6.95 1852.6 23.7 1300.0  
## PET\_Mean\_mm Climate\_Ref HomeRange\_km2 HomeRangeRef DayLength\_km  
## 1 1589.4 Jones et al. 2009 NA NA  
## 2 1653.7 Jones et al. 2009 NA NA  
## 3 1549.8 Jones et al. 2009 NA NA  
## 4 1404.9 Jones et al. 2009 NA 0.40  
## 5 1332.2 Jones et al. 2009 0.03 Jones et al. 2009 NA  
## 6 1633.9 Jones et al. 2009 0.19 Jones et al. 2009 0.32  
## DayLengthRef Territoriality Fruit Leaves Fauna DietRef1  
## 1 NA NA   
## 2 NA NA   
## 3 NA 57.3 19.1 0.0 Campbell et al. 2007  
## 4 Nunn et al. 2003 NA 23.8 67.7 0.0 Campbell et al. 2007  
## 5 NA 5.2 73.0 0.0 Campbell et al. 2007  
## 6 Nunn et al. 2003 0.6506 33.1 56.4 0.0 Campbell et al. 2007  
## Canine\_Dimorphism Canine\_Dimorphism\_Ref Feed Move Rest Social  
## 1 2.210 Plavcan & Ruff 2008 NA NA NA NA  
## 2 NA NA NA NA NA  
## 3 1.811 Plavcan & Ruff 2008 13.75 18.75 57.30 10.00  
## 4 1.542 Plavcan & Ruff 2008 15.90 17.60 61.60 4.90  
## 5 1.783 Plavcan & Ruff 2008 18.33 14.33 64.37 3.00  
## 6 1.703 Plavcan & Ruff 2008 17.94 12.32 66.14 3.64  
## Activity\_Budget\_Ref  
## 1   
## 2   
## 3 Campbell et al. 2007  
## 4 Campbell et al. 2007  
## 5 Campbell et al. 2007  
## 6 Campbell et al. 2007

For this exercise, the end aim is to fit a simple linear regression model to predict longevity (MaxLongevity\_m) measured in months from species’ brain size (Brain\_Size\_Species\_Mean) measured in grams. Do the following for both longevity~brain size and log(longevity)~log(brain size). Fit the regression model and, using {ggplot2}, produce a scatterplot with the fitted line superimposed upon the data. Append the the fitted model equation to your plot.

r <- lm(MaxLongevity\_m ~ Brain\_Size\_Species\_Mean, data = k, level = 0.90)

## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :  
## extra argument 'level' will be disregarded

tidy(r)

## # A tibble: 2 x 5  
## term estimate std.error statistic p.value  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 (Intercept) 249. 11.1 22.4 8.91e-46  
## 2 Brain\_Size\_Species\_Mean 1.22 0.110 11.1 2.69e-20

Raw values:

alpha <- 0.1  
ci <- predict(r,  
 newdata = data.frame(Brain\_Size\_Species\_Mean = seq(min(k$Brain\_Size\_Species\_Mean, na.rm = TRUE), max(k$Brain\_Size\_Species\_Mean, na.rm = TRUE), by = 1)),  
 interval = "confidence", level = 1 - alpha  
)  
ci <- data.frame(ci)  
ci <- cbind(seq(min(k$Brain\_Size\_Species\_Mean, na.rm = TRUE), max(k$Brain\_Size\_Species\_Mean, na.rm = TRUE), by = 1), ci)  
  
  
pi <- predict(r,  
 newdata = data.frame(Brain\_Size\_Species\_Mean = seq(min(k$Brain\_Size\_Species\_Mean, na.rm = TRUE), max(k$Brain\_Size\_Species\_Mean, na.rm = TRUE), by = 1)),  
 interval = "prediction", level = 1 - alpha  
)  
pi <- data.frame(pi)  
head(pi)

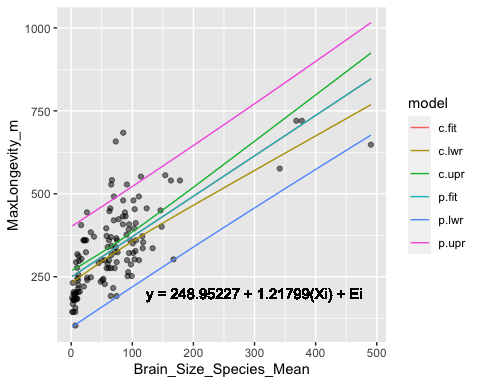
## fit lwr upr  
## 1 250.9376 99.19856 402.6766  
## 2 252.1556 100.43137 403.8798  
## 3 253.3736 101.66396 405.0832  
## 4 254.5916 102.89633 406.2868  
## 5 255.8096 104.12849 407.4906  
## 6 257.0275 105.36043 408.6947

i <- cbind(ci, pi)  
head(i)

## seq(min(k$Brain\_Size\_Species\_Mean, na.rm = TRUE), max(k$Brain\_Size\_Species\_Mean,   
## 1 1.63  
## 2 2.63  
## 3 3.63  
## 4 4.63  
## 5 5.63  
## 6 6.63  
## fit lwr upr fit lwr upr  
## 1 250.9376 232.7302 269.1450 250.9376 99.19856 402.6766  
## 2 252.1556 234.0721 270.2391 252.1556 100.43137 403.8798  
## 3 253.3736 235.4130 271.3341 253.3736 101.66396 405.0832  
## 4 254.5916 236.7529 272.4302 254.5916 102.89633 406.2868  
## 5 255.8096 238.0918 273.5273 255.8096 104.12849 407.4906  
## 6 257.0275 239.4296 274.6255 257.0275 105.36043 408.6947

names(i) <- c("weight", "c.fit", "c.lwr", "c.upr", "p.fit", "p.lwr", "p.upr")  
  
long <- pivot\_longer(i, c.fit:p.upr, names\_to = "model")  
  
g <- ggplot(data = k, aes(x = Brain\_Size\_Species\_Mean, y = MaxLongevity\_m)) +  
 geom\_point(alpha = 0.5) +  
 geom\_line(data = long, aes(x = weight, y = value, group = model, color = model)) +  
 geom\_text(x = 300, y = 200, label = "y = 248.95227 + 1.21799(Xi) + Ei")  
  
  
g

## Warning: Removed 85 rows containing missing values (geom\_point).

 Some summary stats from the model:

t <- tidy(r)  
print(t)

## # A tibble: 2 x 5  
## term estimate std.error statistic p.value  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 (Intercept) 249. 11.1 22.4 8.91e-46  
## 2 Brain\_Size\_Species\_Mean 1.22 0.110 11.1 2.69e-20

c <- confint(r, level = 0.90)  
print(c)

## 5 % 95 %  
## (Intercept) 230.540738 267.36379  
## Brain\_Size\_Species\_Mean 1.035571 1.40041

beta1 = 1.21799

beta0 = 248.95227

beta1 does not equal zero so the null hypothesis is rejected, which indicates that species’ longevity increases as a function of brain size.

90% confidence intervals for the slope are 1.035571 and 1.40041

Produce a point estimate and associated 90% prediction interval for the longevity of a species whose brain weight is 750 gm.

m.summary <- tidy(r)   
beta0 <- m.summary %>%  
 filter(term == "(Intercept)") %>%  
 pull(estimate)  
beta1 <- m.summary %>%  
 filter(term == "Brain\_Size\_Species\_Mean") %>%  
 pull(estimate)  
(h.hat <- beta1 \* 750 + beta0)

## [1] 1162.445

Do you trust the model to predict observations accurately for this value of the explanatory variable? Why or why not?

I do not trust this prediction very much because the the value of the explanatory value falls among those values of x that have the least trustworthy confidence intervals (the upper right portion of the graph). This is because the 4 points in this region are so different than the others, its hard to tell if they are pulling the line artificially.

log values:

log(HomeRange\_km2) ~ log(Body\_mass\_female\_mean)

## log(HomeRange\_km2) ~ log(Body\_mass\_female\_mean)

#make new columns in the dataframe for log transformed values:  
  
log\_df <- k %>% mutate(  
 log\_MaxLongevity\_m = log(MaxLongevity\_m),  
 log\_Brain\_Size\_Species\_Mean = log(Brain\_Size\_Species\_Mean),  
 log\_HomeRange\_km2 = log(HomeRange\_km2),  
 log\_Body\_mass\_female\_mean = log(Body\_mass\_female\_mean)  
)  
  
head(log\_df)

## Scientific\_Name Family Genus Species  
## 1 Allenopithecus\_nigroviridis Cercopithecidae Allenopithecus nigroviridis  
## 2 Allocebus\_trichotis Cercopithecidae Allocebus trichotis  
## 3 Alouatta\_belzebul Atelidae Alouatta belzebul  
## 4 Alouatta\_caraya Atelidae Alouatta caraya  
## 5 Alouatta\_guariba Atelidae Alouatta guariba  
## 6 Alouatta\_palliata Atelidae Alouatta palliata  
## Brain\_Size\_Species\_Mean Brain\_Size\_Female\_Mean Brain\_size\_Ref  
## 1 58.02 53.70 Isler et al 2008  
## 2 NA NA   
## 3 52.84 51.19 Isler et al 2008  
## 4 52.63 47.80 Isler et al 2008  
## 5 51.70 49.08 Isler et al 2008  
## 6 49.88 48.04 Isler et al 2008  
## Body\_mass\_male\_mean Body\_mass\_female\_mean Mass\_Dimorphism  
## 1 6130 3180 1.928  
## 2 92 84 1.095  
## 3 7270 5520 1.317  
## 4 6525 4240 1.539  
## 5 5800 4550 1.275  
## 6 7150 5350 1.336  
## Mass\_Ref MeanGroupSize AdultMales AdultFemale AdultSexRatio  
## 1 Isler et al 2008 NA NA NA NA  
## 2 Smith and Jungers 1997 1.00 1.00 1.0 NA  
## 3 Isler et al 2008 7.00 1.00 1.0 1.00  
## 4 Isler et al 2008 8.00 2.30 3.3 1.43  
## 5 Isler et al 2008 6.53 1.37 2.2 1.61  
## 6 Isler et al 2008 12.00 2.90 6.3 2.17  
## Social\_Organization\_Ref  
## 1   
## 2 Kappeler 1997  
## 3 Campbell et al 2007  
## 4 van Schaik et al. 1999; Kappeler and Pereira 2003; Nunn & van Schaik 2000  
## 5 Campbell et al 2007  
## 6 van Schaik et al. 1999; Kappeler and Pereira 2003; Nunn & van Schaik 2000  
## InterbirthInterval\_d Gestation WeaningAge\_d MaxLongevity\_m LitterSz  
## 1 NA NA 106.15 276.0 1.01  
## 2 NA NA NA NA 1.00  
## 3 NA NA NA NA NA  
## 4 337.62 187 323.16 243.6 1.01  
## 5 NA NA NA NA NA  
## 6 684.37 186 495.60 300.0 1.02  
## Life\_History\_Ref GR\_MidRangeLat\_dd Precip\_Mean\_mm Temp\_Mean\_degC AET\_Mean\_mm  
## 1 Jones et al. 2009 -0.17 1574.0 25.2 1517.8  
## 2 -16.59 1902.3 20.3 1388.2  
## 3 -6.80 1643.5 24.9 1286.6  
## 4 Jones et al. 2009 -20.34 1166.4 22.9 1193.1  
## 5 -21.13 1332.3 19.6 1225.7  
## 6 Jones et al. 2009 6.95 1852.6 23.7 1300.0  
## PET\_Mean\_mm Climate\_Ref HomeRange\_km2 HomeRangeRef DayLength\_km  
## 1 1589.4 Jones et al. 2009 NA NA  
## 2 1653.7 Jones et al. 2009 NA NA  
## 3 1549.8 Jones et al. 2009 NA NA  
## 4 1404.9 Jones et al. 2009 NA 0.40  
## 5 1332.2 Jones et al. 2009 0.03 Jones et al. 2009 NA  
## 6 1633.9 Jones et al. 2009 0.19 Jones et al. 2009 0.32  
## DayLengthRef Territoriality Fruit Leaves Fauna DietRef1  
## 1 NA NA   
## 2 NA NA   
## 3 NA 57.3 19.1 0.0 Campbell et al. 2007  
## 4 Nunn et al. 2003 NA 23.8 67.7 0.0 Campbell et al. 2007  
## 5 NA 5.2 73.0 0.0 Campbell et al. 2007  
## 6 Nunn et al. 2003 0.6506 33.1 56.4 0.0 Campbell et al. 2007  
## Canine\_Dimorphism Canine\_Dimorphism\_Ref Feed Move Rest Social  
## 1 2.210 Plavcan & Ruff 2008 NA NA NA NA  
## 2 NA NA NA NA NA  
## 3 1.811 Plavcan & Ruff 2008 13.75 18.75 57.30 10.00  
## 4 1.542 Plavcan & Ruff 2008 15.90 17.60 61.60 4.90  
## 5 1.783 Plavcan & Ruff 2008 18.33 14.33 64.37 3.00  
## 6 1.703 Plavcan & Ruff 2008 17.94 12.32 66.14 3.64  
## Activity\_Budget\_Ref log\_MaxLongevity\_m log\_Brain\_Size\_Species\_Mean  
## 1 5.620401 4.060788  
## 2 NA NA  
## 3 Campbell et al. 2007 NA 3.967268  
## 4 Campbell et al. 2007 5.495528 3.963286  
## 5 Campbell et al. 2007 NA 3.945458  
## 6 Campbell et al. 2007 5.703782 3.909620  
## log\_HomeRange\_km2 log\_Body\_mass\_female\_mean  
## 1 NA 8.064636  
## 2 NA 4.430817  
## 3 NA 8.616133  
## 4 NA 8.352319  
## 5 -3.506558 8.422883  
## 6 -1.660731 8.584852

log\_fit <- lm(log\_MaxLongevity\_m ~ log\_Brain\_Size\_Species\_Mean, data = log\_df, level = 0.90)

## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :  
## extra argument 'level' will be disregarded

tidy(log\_fit)

## # A tibble: 2 x 5  
## term estimate std.error statistic p.value  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 (Intercept) 4.88 0.0691 70.6 3.32e-103  
## 2 log\_Brain\_Size\_Species\_Mean 0.234 0.0178 13.1 2.17e- 25

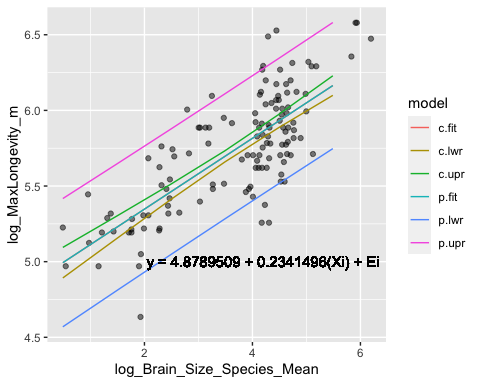
alpha <- 0.1  
ci <- predict(log\_fit,  
 newdata = data.frame(log\_Brain\_Size\_Species\_Mean = seq(min(log\_df$log\_Brain\_Size\_Species\_Mean, na.rm = TRUE), max(log\_df$log\_Brain\_Size\_Species\_Mean, na.rm = TRUE), by = 1)),  
 interval = "confidence", level = 1 - alpha  
)  
ci <- data.frame(ci)  
ci <- cbind(seq(min(log\_df$log\_Brain\_Size\_Species\_Mean, na.rm = TRUE), max(log\_df$log\_Brain\_Size\_Species\_Mean, na.rm = TRUE), by = 1), ci)  
  
  
pi <- predict(log\_fit,  
 newdata = data.frame(log\_Brain\_Size\_Species\_Mean = seq(min(log\_df$log\_Brain\_Size\_Species\_Mean, na.rm = TRUE), max(log\_df$log\_Brain\_Size\_Species\_Mean, na.rm = TRUE), by = 1)),  
 interval = "prediction", level = 1 - alpha  
)  
pi <- data.frame(pi)  
head(pi)

## fit lwr upr  
## 1 4.993352 4.569469 5.417234  
## 2 5.227501 4.809179 5.645823  
## 3 5.461651 5.046859 5.876443  
## 4 5.695800 5.282457 6.109144  
## 5 5.929950 5.515952 6.343948  
## 6 6.164100 5.747352 6.580847

i <- cbind(ci, pi)  
head(i)

## seq(min(log\_df$log\_Brain\_Size\_Species\_Mean, na.rm = TRUE), max(log\_df$log\_Brain\_Size\_Species\_Mean,   
## 1 0.48858  
## 2 1.48858  
## 3 2.48858  
## 4 3.48858  
## 5 4.48858  
## 6 5.48858  
## fit lwr upr fit lwr upr  
## 1 4.993352 4.892460 5.094244 4.993352 4.569469 5.417234  
## 2 5.227501 5.153365 5.301638 5.227501 4.809179 5.645823  
## 3 5.461651 5.411105 5.512197 5.461651 5.046859 5.876443  
## 4 5.695800 5.658987 5.732614 5.695800 5.282457 6.109144  
## 5 5.929950 5.886392 5.973508 5.929950 5.515952 6.343948  
## 6 6.164100 6.099439 6.228760 6.164100 5.747352 6.580847

names(i) <- c("weight", "c.fit", "c.lwr", "c.upr", "p.fit", "p.lwr", "p.upr")  
  
long <- pivot\_longer(i, c.fit:p.upr, names\_to = "model")  
  
g <- ggplot(data = log\_fit, aes(x = log\_Brain\_Size\_Species\_Mean, y = log\_MaxLongevity\_m)) +  
 geom\_point(alpha = 0.5) +  
 geom\_line(data = long, aes(x = weight, y = value, group = model, color = model)) +  
 geom\_text(x = 4.2, y = 5, label = "y = 4.8789509 + 0.2341496(Xi) + Ei")  
  
g

 Some summary stats from the model:

t <- tidy(log\_fit)  
print(t)

## # A tibble: 2 x 5  
## term estimate std.error statistic p.value  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 (Intercept) 4.88 0.0691 70.6 3.32e-103  
## 2 log\_Brain\_Size\_Species\_Mean 0.234 0.0178 13.1 2.17e- 25

c <- confint(log\_fit, level = 0.90)  
print(c)

## 5 % 95 %  
## (Intercept) 4.7644934 4.9934084  
## log\_Brain\_Size\_Species\_Mean 0.2046396 0.2636595

beta1 = 0.2341496

beta0 = 4.8789509

beta1 does not equal zero so the null hypothesis is rejected, which indicates that species’ longevity increases as a function of brain size.

90% confidence intervals for the slope are 0.2046396 and 0.2636595

Produce a point estimate and associated 90% prediction interval for the longevity of a species whose brain weight is 750 gm.

m.summary <- tidy(log\_fit)   
beta0 <- m.summary %>%  
 filter(term == "(Intercept)") %>%  
 pull(estimate)  
beta1 <- m.summary %>%  
 filter(term == "log\_Brain\_Size\_Species\_Mean") %>%  
 pull(estimate)  
(h.hat <- beta1 \* log(750) + beta0)

## [1] 6.429038

Do you trust the model to predict observations accurately for this value of the explanatory variable? Why or why not?

I trust this prediction because the model has narrow confidence intervals.

Looking at your two models (i.e., untransformed versus log-log transformed), which do you think is better? Why?

I think the log transformed model is more useful. The more narrow confidence intervals gives it more accurate predictions.

# Challenge 2

Using the “KamilarAndCooperData.csv” dataset, run a linear regression looking at log(HomeRange\_km2) in relation to log(Body\_mass\_female\_mean) and report your β coeffiecients (slope and intercept).

fit <- lm(log\_HomeRange\_km2 ~ log\_Body\_mass\_female\_mean, data = log\_df)  
results <- tidy(fit, conf.int = TRUE, conf.level = .95) %>%   
 select(term, estimate, std.error, conf.low, conf.high)  
head(results)

## # A tibble: 2 x 5  
## term estimate std.error conf.low conf.high  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 (Intercept) -9.44 0.673 -10.8 -8.11  
## 2 log\_Body\_mass\_female\_mean 1.04 0.0849 0.869 1.20

B0 = -9.44123 B1 = 1.03643

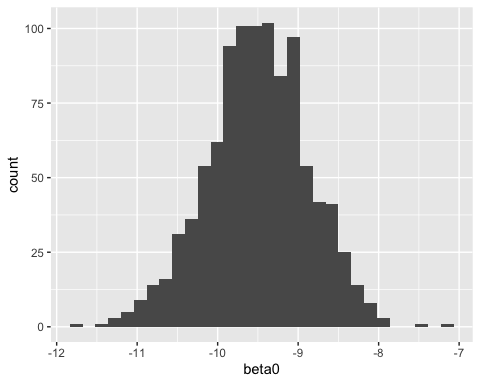
Then, use bootstrapping to sample from the dataset 1000 times with replacement, each time fitting the same model and calculating the appropriate coefficients. [The size of each sample should be equivalent to the total number of observations in the dataset.] This generates a bootstrap sampling distribution for each β coefficient. Plot a histogram of these sampling distributions for β0 and β1.

boot <- tibble()   
  
for (i in 1:1000) {  
 s <- sample\_n(log\_df, size = nrow(log\_df), replace = TRUE)  
 fit <- lm(s, formula = log\_HomeRange\_km2 ~ log\_Body\_mass\_female\_mean)  
 tidy <- tidy(fit) %>% pull(estimate) -> results  
 names(results) <- tidy(fit)$term  
 boot <- bind\_rows(boot, results)  
}  
  
  
names(boot) <- c("beta0", "beta1")  
boot

## # A tibble: 1,000 x 2  
## beta0 beta1  
## <dbl> <dbl>  
## 1 -9.33 1.03   
## 2 -9.04 0.985  
## 3 -9.70 1.10   
## 4 -10.2 1.16   
## 5 -9.37 1.01   
## 6 -9.47 1.01   
## 7 -9.08 0.996  
## 8 -9.85 1.10   
## 9 -11.1 1.25   
## 10 -9.07 0.997  
## # … with 990 more rows

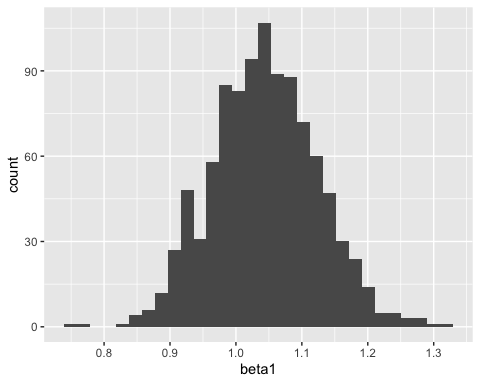
B0\_hist <- ggplot(boot, aes(x = beta0)) +  
 geom\_histogram()  
  
B0\_hist

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



B1\_hist <- ggplot(boot, aes(x = beta1)) +  
 geom\_histogram()  
  
B1\_hist

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

 Estimate the standard error for each of your β coefficients as the standard deviation of the sampling distribution from your bootstrap.

se\_beta0 <- sd(boot$beta0)  
se\_beta0

## [1] 0.6270647

se\_beta1 <- sd(boot$beta1)  
se\_beta1

## [1] 0.08001738

Also determine the 95% CI for each of your β coefficients based on the appropriate quantiles from your sampling distribution.

beta1\_upr\_ci <- quantile(boot$beta1, 0.975)  
beta1\_upr\_ci

## 97.5%   
## 1.199587

beta1\_lwr\_ci <- quantile(boot$beta1, 0.025)  
beta1\_lwr\_ci

## 2.5%   
## 0.8974725

beta0\_upr\_ci <- quantile(boot$beta0, 0.975)  
beta0\_upr\_ci

## 97.5%   
## -8.306264

beta0\_lwr\_ci <- quantile(boot$beta0, 0.025)  
beta0\_lwr\_ci

## 2.5%   
## -10.76872

How do the SEs estimated from the bootstrap sampling distribution compare to those estimated mathematically as part of lm() function?

bootstrap: se\_beta0 = 0.6165899 se\_beta1 = 0.07880122

lm() function

se\_beta0 = 0.67293459 se\_beta1 = 0.08487709

They standard erros calculated from the lm() function are very similar to those derived from bootstrapping.

How do you bootstrap CIs compare to those estimated mathematically as part of the lm() function?

bootstrap: beta1\_upr\_ci = 1.195263 beta1\_lwr\_ci = 0.8920778

lm(): beta1\_upr\_ci = 1.204292 beta1\_lwr\_ci = 0.8685707

bootstrap: beta0\_upr\_ci = -8.34653 beta0\_lwr\_ci = -10.75408

lm() beta0\_upr\_ci = -8.110374 beta0\_lwr\_ci = -10.7720889

The confidence intervals calculated from the lm() function are similar to those derived from bootstrapping.

# Challenge 3

Write your own function, called boot\_lm(), that takes as its arguments a dataframe (d=), a linear model (model=, written as a character string, e.g., “logHR ~ logBM”), a user-defined confidence interval level (conf.level=) with default “0.95”, and a number of bootstrap replicates (reps=, with default “1000”).

Your function should return a dataframe that includes: the β coefficient names; the value of the β coefficients, their standard errors, and their upper and lower CI limits for the linear model based on your original dataset; and the mean β coefficient estimates, SEs, and CI limits for those coefficients based on your bootstrap.

Use your function to run the following models on the “KamilarAndCooperData.csv” dataset:

log(HomeRange\_km2) ~ log(Body\_mass\_female\_mean)  
log(DayLength\_km) ~ log(Body\_mass\_female\_mean)  
log(HomeRange\_km2) ~ log(Body\_mass\_female\_mean) + MeanGroupSize

# ```{r}

boot\_lm <- function(d, model, conf.level = 0.95, reps = 1000) {

og <- lm(d, formula = model) lm\_results <- tidy(og, conf.int = TRUE, conf.level = .95) %>% select(term, estimate, std.error, conf.low, conf.high)

boot <- tibble()

for (i in 1:reps) { s <- sample\_n(d, size = nrow(d), replace = TRUE) fit <- lm(s, formula = model) tidy <- tidy(fit) %>% pull(estimate) -> results names(results) <- tidy(fit)$term boot <- bind\_rows(boot, results) }

names(boot) <- c(“beta0”, “beta1”)

se\_beta0 <- sd(bootbeta1)

beta1\_upr\_ci <- quantile(bootbeta1, 0.025)

beta0\_upr\_ci <- quantile(bootbeta0, 0.025)

boot\_results <- cbind(bootbeta1, se\_beta1, beta1\_lwr\_ci, beta1\_upr\_ci)

final <- cbind(lm\_results, boot\_results)

return(boot)

}

names(boot) <- c("beta0", "beta1")  
  
se\_beta0 <- sd(boot$beta0)  
  
  
se\_beta1 <- sd(boot$beta1)  
  
  
se(boot$beta1)  
  
percent\_ci <- 95  
alpha <- 0.05  
  
lower\_ci <- mean(boot$beta0) + qnorm(alpha / 2) \* se\_beta0  
  
  
upper\_ci <- mean(boot$beta1) + qnorm(alpha / 2) \* se\_beta1  
   
 tidy <- tidy(boot\_fit) %>% pull(estimate)  
 names(results) <- tidy(boot\_fit)$term  
 boot <- bind\_rows(boot, results)  
  
  
  
  
  
#```{r}  
  
m <- boot\_lm(k, log(MaxLongevity\_m) ~ log(Brain\_Size\_Species\_Mean))  
tidy(m)  
a <- log(HomeRange\_km2) ~ log(Body\_mass\_female\_mean)  
tidy(a)  
b <- log(DayLength\_km) ~ log(Body\_mass\_female\_mean)  
tidy(b)  
c <- log(HomeRange\_km2) ~ log(Body\_mass\_female\_mean) + MeanGroupSize  
tidy(c)