Ian Eggleston

Worked with Sarah Guitart

1. Sample size is 123  
   dat\_pen = subset(penguins, species == "Gentoo")  
   alpha = 0.05  
   n = sum(!is.na(dat\_pen$bill\_length\_mm))  
   n
2. Standard deviation = 3.082  
   dat\_sd = sd(dat\_pen$bill\_length\_mm, na.rm = TRUE)  
   dat\_sd
3. Critical t-values = 1.9796  
   t\_crit = abs(qt(alpha / 2, df = n - 1))
4. Sample standard error = 0.2779  
   sse = dat\_sd / sqrt(n)  
   sse
5. Confidence intervals: lower = 0.4418, upper = 1.5420  
   ci\_radius = sse \* t\_crit  
   gentoo\_bill\_length\_ci = c(  
    lower = mean(!is.na(dat\_pen$bill\_length\_mm)) - ci\_radius,  
    upper = mean(!is.na(dat\_pen$bill\_length\_mm)) + ci\_radius)   
     
   print(round(gentoo\_bill\_length\_ci, 4))
6. Confidence interval = 46.97 and 48.07
7. Code for questions 7-8

require(boot)

boot\_mean = function(x, i)

{

return(mean(x[i], na.rm = TRUE))

}

myboot =

boot(

data = dat\_pen$bill\_length\_mm,

statistic = boot\_mean,

R = 10000)

print(myboot)

str(myboot)

mean(dat\_pen$bill\_length\_mm)

myboot$t0

mean(myboot$t) - myboot$t0

sd(myboot$t)

quantile(

myboot$t,

c(0.025, 0.975))

rm(list = ls())

# Re-read my data:

moths = read.csv(here("data", "moths.csv"))

moth\_dat = moths[,-1]

rarefaction\_sampler = function(input\_dat, n\_iterations)

{

n\_input\_rows = nrow(input\_dat)

results\_out = matrix(

nrow = n\_iterations,

ncol = n\_input\_rows)

# The outer loop: runs once for each bootstrap iteration. index variable is i

for(i in 1:n\_iterations)

{

# The inner loop: simulates increasing sampling intensity

# Sampling intensity ranges from 1 site to the complete count of

# sites in the input data (n)

for(j in 1:n\_input\_rows)

{

# sample the input data row indices, with replacement

rows\_j = sample(n\_input\_rows, size = j, replace=TRUE)

# Creates a new data matrix

t1 = input\_dat[rows\_j, ]

# Calculates the column sums

t2 = apply(t1, 2, sum)

# Counts the number of columns in which any moths were observed

results\_out[i, j] = sum(t2 > 0)

}

}

return(results\_out)

}

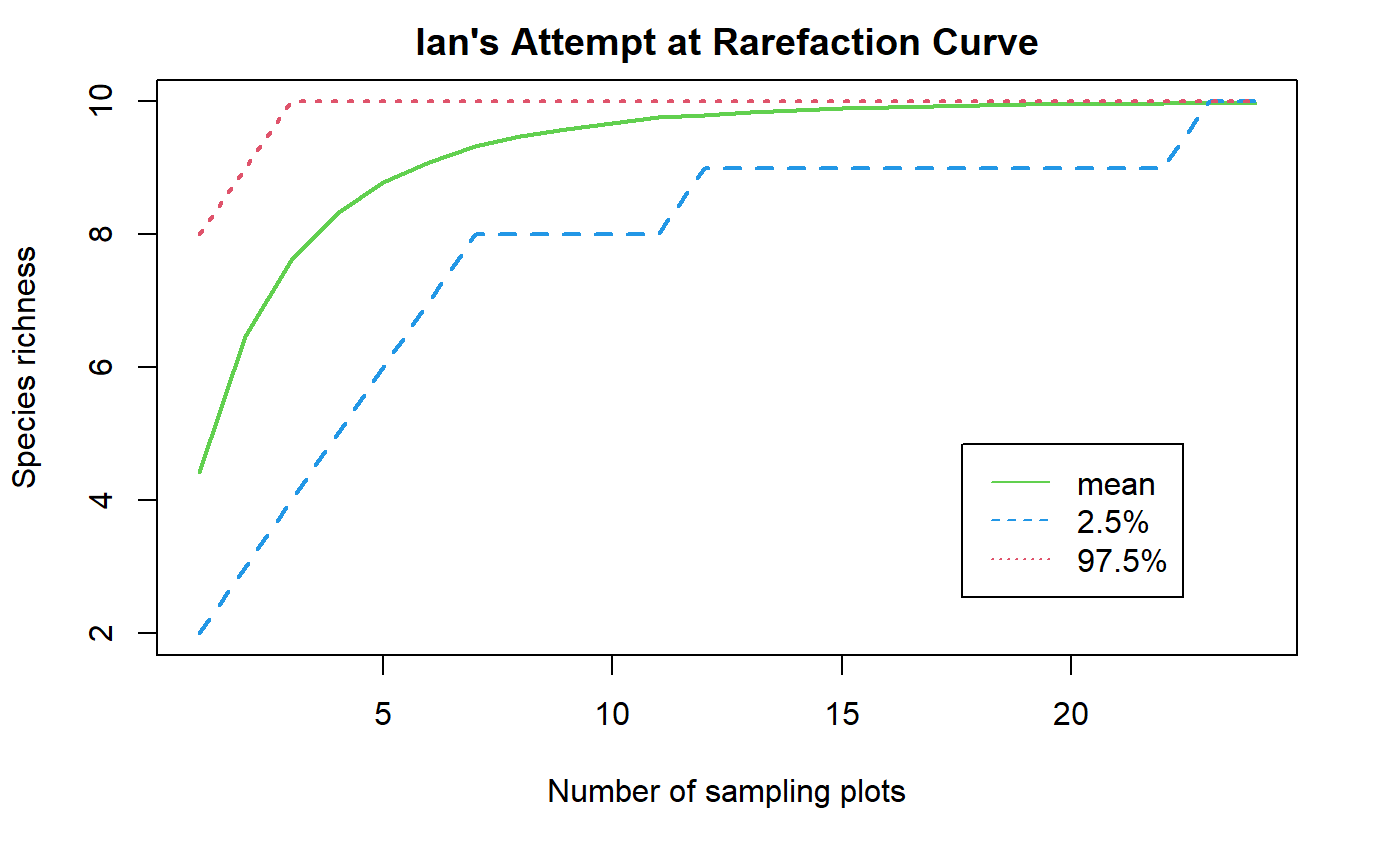
rarefact = rarefaction\_sampler(moth\_dat, 100)

Rarefact

1. The most difficult part was identifying the issue with the code, however R provides some help by describing the problem. After finding n was the wrong variable, it was easy to replace it with the correct one.

moths = read.csv(here("data", "moths.csv"))

rarefact = rarefaction\_sampler(moths[,-1], 10000)



rare\_mean = apply(rarefact, 2, mean)

rare\_quant = apply(rarefact, 2, quantile, probs=c(0.025, 0.975))

rare = t(rbind(rare\_mean, rare\_quant))

matplot(

rare,

type='l',

xlab='Number of sampling plots',

ylab='Species richness',

main="Ian's Attempt at Rarefaction Curve",

col = c(3,4,2), lwd = 2

)

legend(

'bottomright',

legend=c('mean','2.5%','97.5%'),

lty=c(1,2,3),col=c(3,4,2), inset=c(.1,.1))

1. Based on the plot, 15 sites would most likely allow you to see all of the moth species. At this number of sampling sites, they observed all 10 moth species on average. If you want to increase the odds of this, 20 sampling sites almost guarantees you will see the 10 species based on the confidence intervals. The graph shows on average how many species were seen at different numbers of sites. It shows the likelihood of seeing more species increases with sampling more sites.