Jessica Bonin
Analysis of Environmental Data
Lab 7 Report
October 31, 2021
Worked with Juliana Berube, Andrew Gordon, and Julia Vineyard

1. What is the sample size, n? Show the code you used for the calculation and remember to check for missing data.

```
require(palmerpenguins)
gentoo_penguins <- penguins[which(penguins$species=='Gentoo'), ]
gentoo_penguins_2 <-
gentoo_penguins[complete.cases(gentoo_penguins$bill_length_mm), ]
```

n = **123**

2. What is the sample standard deviation? Show the code you used for the calculation.

```
sd(gentoo_penguins_2$bill_length_mm)
```

Sd= **3.081857**

3. What are the critical t-values? Show the R code you used for the calculation.

```
qt(.975, 122, lower.tail = FALSE, log.p = FALSE)
qt(.975, 122, lower.tail = TRUE, log.p = FALSE)
```

**-1.9796** and **1.9796**

4. What is the sample standard error? Show the R code you used for the calculation.

```
sse_mean = function(x)
{
  sd(x, na.rm = TRUE)/(sqrt(length(x[!is.na(x)])))
}
sse_mean(gentoo_penguins_2$bill_length_mm)
```

**sse = 0.2778817**

5. Finally, construct the CI and show the R code you used for the calculation.

```
mean(gentoo_penguins_2$bill_length_mm)-
  (sse_mean(gentoo_penguins_2$bill_length_mm)*
     qt(.975, 122, lower.tail = TRUE, log.p = FALSE))

mean(gentoo_penguins_2$bill_length_mm)+
  (sse_mean(gentoo_penguins_2$bill_length_mm)*
     qt(.975, 122, lower.tail = TRUE, log.p = FALSE))
```

**46.95478, 48.05497**

6. What is the CI?

**46.97234, 48.06179**

7. Show the r code you used to call the boot() function.

```
require(boot)
boot_mean = function(x, i)
{
  return(mean(x[i], na.rm = TRUE))
}
myboot =
  boot(data = gentoo_penguins_2$bill_length_mm, statistic =
  boot_mean, R = 10000)
```

8. Show the r code you used to calculate the upper and lower 2.5% quantiles.

```
quantile(myboot$t, c(0.025, 0.975))
```

9. Show your completed rarefaction_sampler() function.

```
rarefaction_sampler = function(input_dat, n_iterations)
{
  n_input_rows = nrow(input_dat)
  moth_dat = moths[,-1]
  n = nrow(moth_dat)

  results_out = matrix(
    nrow = n_iterations,
    ncol = n_input_rows)

  for(i in 1:n_iterations)
  {
    for(j in 1:n)
    {
      rows_j = sample(n, size = j, replace=TRUE)
      t1 = input_dat[rows_j, ]
      t2 = apply(t1, 2, sum)
      results_out[i, j] = sum(t2>0)
    }
  }
  return(results_out)
}
```

10. What did you find most difficult about building the function?

**The most difficult thing was figuring out that you had to define the parameters inside the function in order to have it run when the environments are cleared.**

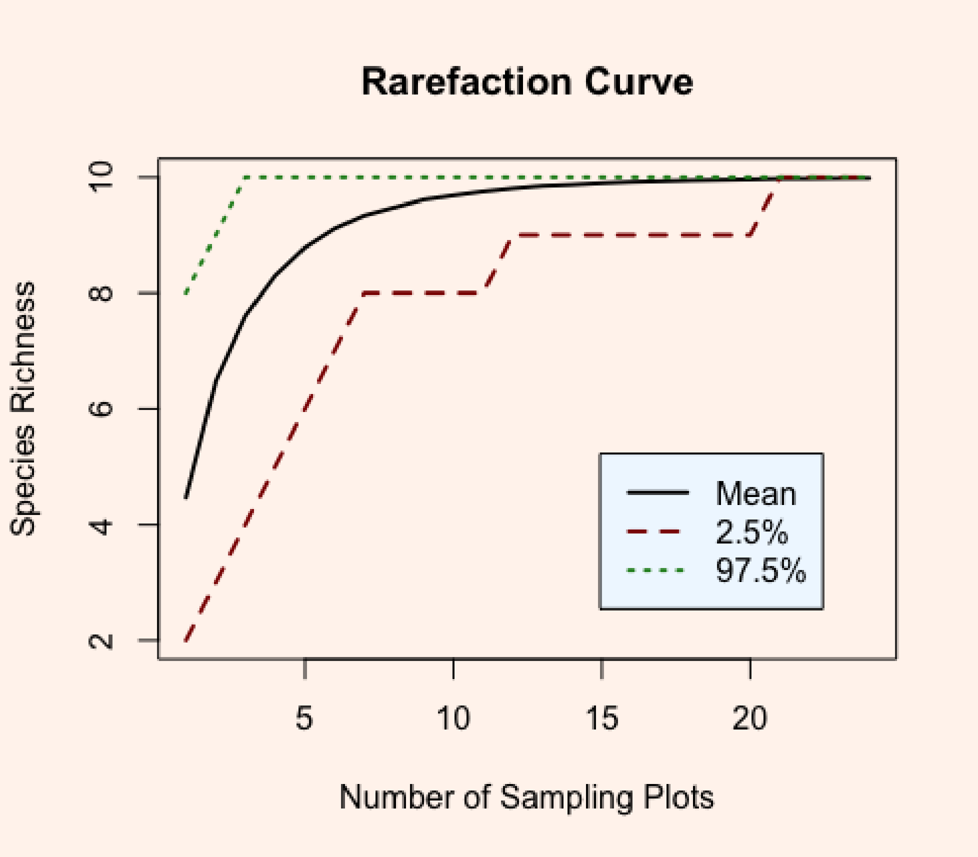
11. Show the code you used to perform the simulations and construct the curve.

```
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))
```

12. Include your rarefaction curve plot in your report. Show the R-code you used to create your plot.

```
par(bg= "seashell")
matplot(
  rare,
  type='l',
  xlab='Number of Sampling Plots',
  ylab='Species Richness',
  main='Rarefaction Curve',
  lty=c("solid", "dashed", "dotted"),
  col=c("black", "red4", "forestgreen"),
  lwd = c(2,2,2))

legend(
  'bottomright',
  legend=c('Mean','2.5%','97.5%'), bg = "aliceblue",
  lty=c("solid", "dashed", "dotted"),col=c("black", "red4",
"forestgreen"), lwd = c(2,2,2), inset=c(.1,.1))
```



13. About how many sites should you visit if you want to see all of the moth species? Explain your reasoning using your rarefaction curve figure.

**Over 20 (about 21). The reason I say this is because this is the number of plots sampled when all of the curves meet.**