Final Synthesis

Linda Dominguez

(1b) Construct the population projection matrix and calculate the growth rate

```
# Vital Rates
S1 <- 0.706  #1st year survival
S2 <- 0.65  #Juvenile survival
S3 <- 0.75  #Sub-adult survival
S4 <- 0.823  #Adult survival
M2 <- 0.398  #Fraction of juvenile survivors that become sub-adults
M3 <- 0.108  #Fraction of sub-adult survivors that become adults
E <- 2.5  #Per capita adult fecundity (number of eggs)

# Creates an 'empty' 4x4 matrix of zeros
A <- array(0, c(4, 4))
rownames(A) <- c("1stYear", "juveniles", "subadults", "adults")
colnames(A) <- rownames(A)
# # # Define matrix elements from vital rates below # # #

# Here is an example for the Egg to Juvenile Transition
A[2, 1] <- S1  #all surviving 1styears go to juveniles
# A Continue building the matrix
A[2, 2] <- S2 - (S2 * M2)  #survival minus juveniles going to adult
A[3, 2] <- M2 * S2  #juvenile to subadult
A[3, 3] <- S3 - (S3 * M3)  #survival minus subadults going to adult
# A
A[4, 3] <- M3 * S3  #subadult to adult
A[4, 4] <- S4
A[1, 4] <- E * S4  #adult to eggs
# A

# Calculate the population growth rate


getLambdaFunct <- function(x) {
    "lambda"
    return(as.numeric(eigen(x)$values[1]))
}
getLambdaFunct(A)
```

```
## [1] 0.9863854
```

```
originalLamb <- getLambdaFunct(A)

A
```

```
##           1stYear juveniles subadults adults
```

```
## 1stYear     0.000    0.0000     0.000 2.0575
## juveniles   0.706    0.3913     0.000 0.0000
## subadults   0.000    0.2587     0.669 0.0000
## adults      0.000    0.0000     0.081 0.8230
```

(1d) Add in the effect of poaching

```
# Be sure to set the poaching level to 0.05 (or 5%)
P <- 0.05  #poaching level


# Copy in the code from above and alter it to incorporate the effect of poaching
S1 <- 0.706  #1st year survival
S2 <- 0.65  #Juvenile survival
S3 <- 0.75  #Sub-adult survival
S4 <- 0.823  #Adult survival
M2 <- 0.398  #Fraction of juvenile survivors that become sub-adults
M3 <- 0.108  #Fraction of sub-adult survivors that become adults
E <- 2.5  #Per capita adult fecundity (number of eggs)
NewAdultSurvival <- S4 - (S4 * P)
# Creates an 'empty' 4x4 matrix of zeros
A <- array(0, c(4, 4))
rownames(A) <- c("1stYear", "juveniles", "subadults", "adults")
colnames(A) <- rownames(A)
# # # Define matrix elements from vital rates below # # #

# Here is an example for the Egg to Juvenile Transition
A[2, 1] <- S1  #all surviving 1styears go to juveniles
# A Continue building the matrix
A[2, 2] <- S2 - (S2 * M2)  #survival minus juveniles going to adult
A[3, 2] <- M2 * S2  #juvenile to subadult
A[3, 3] <- S3 - (S3 * M3)  #survival minus subadults going to adult
# A
A[4, 3] <- M3 * S3  #subadult to adult
A[4, 4] <- NewAdultSurvival

A[1, 4] <- NewAdultSurvival * E  #adult to eggs
A
```

```
##           1stYear juveniles subadults   adults
## 1stYear     0.000    0.0000     0.000 1.954625
## juveniles   0.706    0.3913     0.000 0.000000
## subadults   0.000    0.2587     0.669 0.000000
## adults      0.000    0.0000     0.081 0.781850
```

```
# Calculate the population growth rate
getLambdaFunct(A)
```

```
## [1] 0.9618169
```

```
newLambda <- getLambdaFunct(A)
```

(1e) What is the sensitivity of Lambda to poaching?

Note that you do not need to answer this question in R, but you can if you want to

```r
# Calculate the sensitivity value associated with poaching sensitivity

sensitivity <- (newLambda - originalLamb)/(P - 0)
sensitivity
```

```
## [1] -0.4913699
```

Projection 20 years

```r
# startingPop <- A
years <- 20  # want t=20

# popMatrixProjection <- matrix(0, nrow = years, ncol = length(startingPop))
# colnames(popMatrixProjection) <- rownames(A) rownames(popMatrixProjection) <-
# 1:years

# popMatrixProjection[1,] <- startingPop

# for (year in 2:years) { popMatrixProjection[year,] <- A %*%
# popMatrixProjection[year-1,] }

# #[plotting weplot.MM(popMatrixProjection, title='Population Projection of Sea
# Turtles')

deforestationRate <- 0.012  # 1.2% decline per year
fractHabitatRemaining <- (1 - deforestationRate)^years
fractHabitatRemaining
```

```
## [1] 0.7854868
```

```r
# question 1.7
after5 <- (1 - deforestationRate)^5
after5
```

```
## [1] 0.9414228
```

```r
after5Rate <- 0.035
fractRemain <- (1 - after5Rate)^15
fractRemain
```

```
## [1] 0.5860163
```

```r
# 0.5860163
```