BIOL 564 Lab 3: Structured Populations

Senay Yitbarek

Spring 2025

I. Overview

In this lab you will practice working with projection matrices. The goal is to give you a sense of how they can be used to understand age- and stage-structured populations. You'll also get a taste of programming.

Rather than running a pre-fabricated simulation, in this week's lab you will fill in some simple missing pieces of R code to generate population dynamics for a discrete-time stage-structured model. Simply answer the questions below where they are asked, and follow the instructions to make modifications to the R code.

You will likely find it helpful to run the code in the R console as you go, to test it out. To run a single line of code in your R markdown script, place your cursor on the desired line and press $Ctrl + Enter$ (on a mac, use $Command$ instead of $Ctrl$). To run an entire chunk of code, place your cursor anywhere in the chunk and press $Ctrl + Shift + Enter$. Alternatively, you can click on Run in the top right of the R Studio scripting window.

A few tips when using R:

- Addition, subtraction, multiplication, and division are supported by `+ - * /` respectively.

- Where you put spaces doesn't matter. Line breaks do matter.

- Matrices are used in two ways in this lab: (1) to represent populations over time, with different stages in rows and different time points in columns, and (2) as projection matrices (in Exercise 4).

- For today's lab, you should have these packages installed and loaded: `ggplot2`, `reshape2`, `primer`.

II. Exercises

Exercise 1

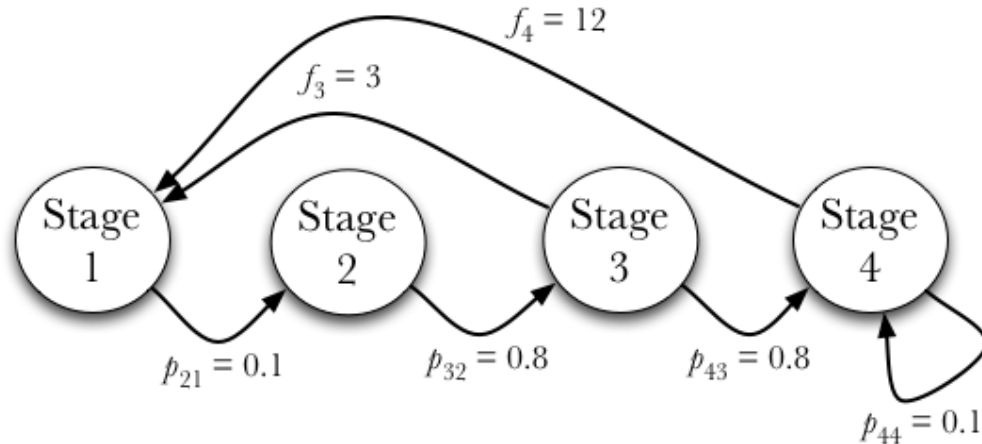Consider the following diagram of a stage-structured population:

Figure 1. A stage-structured population.

a. What do the different arrows and values represent? (2pts)

b. Write four linear equations relating the populations of each stage at time $t+1$ to populations at time t, using, for example, $N_1(t)$ and $N_1(t+1)$ to represent the population of stage 1 at times t and $t+1$. (1pt)

Fill in the equations here:

$N_1(t+1) =$

$N_2(t+1) =$

$N_3(t+1) =$

$N_4(t+1) =$

c. (0.5pt) Now you'll put the equations you wrote into R code to simulate the population from time 1 to time 2. Below is some code to get you started. It creates a matrix comprising N values over time: 4 rows, one for each stage, by 40 columns, one for each timestep. Initially $(t=1)$, the first stage has 10 individuals and second, third, and fourth stages each have 1 individual.

```
# Population sizes over time:
# N[i, t] is the population size of stage i at time t
# 4 stages, 10 timesteps
N <- matrix(0, nrow = 4, ncol = 40, dimnames = list(c('N1', 'N2', 'N3', 'N4'), c(1:40)))

# Initial population sizes: N1 = 10, N2 = 1, N3 = 1, N4 = 1
N[1, 1] <- 10
N[2, 1] <- 1
N[3, 1] <- 1
N[4, 1] <- 1
```

- Translate the four equations you provided above into R code to generate the numbers of individuals in each stage at time 2. Once you have done this, display what N looks like at time 2 by entering the command `N[, 2]`.

d. (0.5pt) Now, we'll repeat the process we used to get from time 1 to time 2, to get all the way to time 40, by using a "loop". The code starts at $t=2$, runs the body of the loop, sets $t=3$, runs the body of the loop again, and so on. You will need to fill in the equations for $N[,t+1]$ in terms of population sizes at previous timesteps $N[,t]$:

2

```
# MODIFY THIS CODE TO GENERATE YOUR ANSWER

for (t in 2:39) {
    N[1, t+1] <- 0
    N[2, t+1] <- 0
    N[3, t+1] <- 0
    N[4, t+1] <- 0
}
```

Use the following code to plot population sizes over time.

```
x <- melt(t(N))
colnames(x) <- c('t', 'Stage', 'PopSize')

# Remove the leading hashtags when you are ready to plot
#ggplot(x, aes(x = t, y = PopSize, color = Stage)) +
#  geom_line() + geom_point()
```

Exercise 2

Now let's calculate the total growth rate over time at each time step.

First, we need to calculate the changes in total population size over time. This is done by taking the column sums of our N matrix, which we'll also plot:

```
N_total <- colSums(N)

# Plotting
x <- data.frame(t = seq_along(N_total), N_total)

# Remove the leading hashtags when you are ready to plot
#ggplot(x, aes(x = t, y = N_total)) +
#  geom_point() + geom_line()
```

a. (1pt) Now, you will write a line of code to calculate the growth rate at each timestep. The following code loops through values of t as before: $t = 2, 3, 4, \ldots$ The vector `lambda` will have 39 entries, one for each transition in time. (Note, a vector is a sequence of terms, in this case λ values, which is also sometimes called an array. We save the vector to variable $lambda$). In other words, we want `lambda[t]` to represent the growth rate from time t to time $t + 1$, which will involve using `N_total[t]` and `N_total[t+1]`.

```
# MODIFY THIS CODE TO GENERATE YOUR ANSWER

lambda <- numeric(39)

for (t in 1:39) {
    lambda[t] <- 0
}
```

Plotted:

3

```
x <- data.frame(t = seq_along(lambda), lambda)

# Remove the leading hashtags when you are ready to plot
#ggplot(x, aes(x = t, y = lambda)) +
#  geom_point() + geom_line()
```

The actual values:

```
# Remove the leading hashtags when you are ready to plot
# data.frame(lambda)
```

Exercise 3

Let's take a look at how the stage distribution – the fraction of the total population occupied by different stages – changes over time. As before, the plot will only show dummy data until you have correctly completed the lab exercises up to this point.

```
x <- apply(N, 1, function(x) x / N_total)
x <- melt(x)
colnames(x) <- c('t', 'Stage', 'PopFraction')

# Remove the leading hashtags when you are ready to plot
#ggplot(x, aes(x = t, y = PopFraction, color = Stage)) +
#  geom_point() + geom_line() + ylim(c(0,1))
```

a. What do you observe in the population dynamics? (2pts)

Let's also take a look at the numerical values of the stage distribution at $t = 40$:

```
# Remove the leading hashtags when you are ready to plot
#melt(N[, 40])
```

To examine the *proportion* of individuals in each stage distribution at $t = 40$:

```
# Remove the leading hashtags when you are ready to plot
#melt(N[,40]/sum(N[,40]))
```

Exercise 4

Now, you'll convert the system of equations to matrix form. In the following code, make a separate line for each entry of the projection matrix that should not be zero. For example, to set the fecundity of stage 1 equal to 10, type `A[1, 1] <- 10`.

a. Create one line of code for each new entry into the projection matrix (0.5pt):

```
# First we make an empty matrix called 'A'. Then, populate it accordingly.
A <- matrix(0, nrow = 4, ncol = 4)
```

Check to make sure it looks right:

```

```
Remove the leading hashtags when you are ready to plot
#A
```

R can directly calculate the eigenvalues and eigenvectors of a matrix.

```
eigenValsVecs <- eigen(A)

# Remove the leading hashtags when you are ready to plot
#eigenValsVecs
```

b. What does the value of the largest eigenvalue (in absolute value) represent? What does it have to do with the results in Exercise 2? (2pts)

c. Recall that the columns (not rows) of the eigenvector matrix represent the eigenvectors corresponding to the different eigenvalues. What does the eigenvector corresponding to the largest eigenvalue represent? Before answering the question, it will be useful to scale the eigenvector by its sum, which you can do using the code below. How does your answer relate to your results in exercise 3? (2pts)

```
# Scale the first eigenvector in the eigenvector matrix, and remove any imaginary components

# Remove the leading hashtags when you are ready to plot
#Re(eigenValsVecs$vectors[, 1] / sum(eigenValsVecs$vectors[, 1]))
```

Exercise 5

Here is a real example from the literature. Crouse et al. (1987)* created a stage-structured model to look at a loggerhead sea turtle population, primarily motivated by conservation concerns. They identified seven stages: eggs/hatchlings, small juveniles, large juveniles, sub-adults, novice breeders, first year migrants and mature breeders. Based on their own work and values from the literature, they created the following Lefkovitch matrix:

$$
\mathbf{P} = \begin{bmatrix}
0 & 0 & 0 & 0 & 127 & 4 & 80 \\
0.6747 & 0.7370 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.0486 & 0.6610 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.0147 & 0.6907 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.0518 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.8091 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.8091 & 0.8089
\end{bmatrix}
$$

Figure 2. Leftkovitch matrix for a loggerhead sea turtle population, from Crouse *et al.* (1987).

a. First create an empty matrix called 'P'. Then populate the matrix to answer the exercise. Note: several values can be added simulaneously using the command `c()`. (0.5pt)

```
# MODIFY THIS CODE TO GENERATE YOUR ANSWER

P <- matrix(0, nrow = 7, ncol = 7)
```

```
P[1, 5:7] <- 0
P[2, 1:2] <- 0
P[3, 2:3] <- 0
P[4, 3:4] <- 0
P[5, 4] <- 0
P[6, 5] <- 0
P[7, 6:7] <- 0

# Remove the leading hashtags when you are ready to plot
# DemoInfo(P)
```

b. Use the `eigen()` function to calculate the eigenvalues and eigenvectors for **P**. Is the loggerhead population increasing or decreasing? (1pt)

c. Crouse *et al.* noted that one traditional conservation technique was to protect eggs until they hatch successfully on beaches. Which transition probability would this affect? Let's *assume* that conservation efforts could get this transition probability up to one, i.e., total egg protection and hatching success. Would the population then be increasing or decreasing? (2pts)

##Sensitivity & Elasticity

The analysis of how changes in transition probabilities or fecundities affect the growth/decline rate of the population (i.e., age/stage structured models) has obvious conservation implications. The approach also allows us to test how sensitive our predictions are to errors in our inputs. For example, say we do not know for sure the transition probability from sub-adult to novice breeder; how much do our modeling results change if we modify the transition probability up or down ten percent? This type of assessment is called sensitivity (or elasticity) analysis. Technically, the sensitivity of the ijth entry is defined as

$$s_{ij} = \frac{\partial \lambda}{\partial P_{ij}},$$

where λ is the growth rate of the population with a stable stage distribution, and P_{ij} is the ijth entry of **P**. (Remember: λ is a function of all the entries in **P** so the sensitivity is just how much λ changes as you change an element of **P**, which is why the partial derivative is used.)

Similarly, each entry of **P** has an elasticity, ϵ:

$$\epsilon_{ij} = \frac{\partial \ln(\lambda)}{\partial \ln(P_{ij})} = \frac{P_{ij}}{\lambda} \frac{\partial \lambda}{\partial P_{ij}},$$

The elasticity values are proportional to the sensitivities, and have the added benefit that they sum to one (so they are easier to compare across different models). The interpretation of the two values is similar: the larger the elasticity or sensitivity of an entry, the more λ will increase when that entry is increased and the more λ will decrease when that entry is decreased. Both measures can be useful - each one is better suited to certain problems than the other.

Exercise 6

The `DemoInfo()` function from the **primer** R package can be used to easily calculate λ, stable age/stage structure, reproductive value, sensitivity, and elasticity. Additionally, you can use the following code to visually display the elasticity matrix elements (the code assumes your matrix is saved to the variable P). The darker the color, the larger the elasticity. Use the output of the code below to answer the following questions.

```
# Calculate the elasticity matrix
# x <- DemoInfo(P)$Elasticities

# Remove the leading hashtags when you are ready to visualize the elasticity matrix
#image(t(x)[, 7:1])
```

a. What is the relative importance to overall population growth of (i) increasing fecundity, (ii) increasing transition probabilities from one life stage to the next, and (iii) increasing survival probabilities within life stages? (3pts)

b. Given limited time and resources, and based on your elasticity analysis, *i)* How would a conservationist best invest their time and resources into helping the loggerhead population recover? *ii)* What would be a wasteful investment of time and resources? (2pts)

---

*Crouse, Deborah T., Larry B. Crowder, and Hal Caswell. "A stage-based population model for loggerhead sea turtles and implications for conservation." Ecology 68.5 (1987): 1412-1423.