Ecological Dynamics
Lab 2: Structured population models

Background

- Key characteristics such as fertility and mortality often vary with age in complex organisms.
- Understanding such variation is often critical for managing or conserving natural populations.
- The effects of demographic and environmental stochasticity are ubiquitous and typically assessed via simulation.

Objectives

- Learn to construct a Leslie matrix from life table data.
- Use linear algebra to estimate key characteristics of structured population models.
- Test the sensitivity of structured population models to environmental stochasticity.

Instructions

- Launch RStudio, open a new script file and save it as `lab1-yourLastName.R`.
- Complete the activities below by adding the appropriate commands to your `R` script file.
- Embed your answers as comments in the `R` script file.
- At the end of your session, save your file onto a flash drive (files saved on lab computers are wiped when you log out).

Task 1: Modeling the dynamics of structured populations

1. The first step in modeling age-structured populations is often to load the data and compute fertility and survivorship schedules. To do so, begin by downloading the following dataset from the web:

```
life.table <- read.csv(file = "lifetable.csv")
```

   Once the data has been downloaded, compute the survivorship schedule $S(i) = \frac{S(i)}{S(0)}$, survivorship probabilities $P_i = \frac{l(i)}{l(i-1)}$ and the fertilities $F_i = b(i)P_i$ for each age class i. Note that subscripts i refer to age classes whereas (i) refer to ages (e.g., $b(i)$ refers to age i and age class $i+1$ whereas P_i refers to age class i and age $i-1$).

2. Now construct the Leslie matrix for this dataset. Remember that the Leslie matrix is a $k \times k$ square matrix, where k represents the number of age classes. The top row contains the fertilities F_i and the subdiagonal contains the survivorship probabilities P_i. All other entries are zeros.

3. Use function `eigen` to compute the dominant eigenvalue and the corresponding eigenvector of the Leslie matrix. What do the eigenvalues and the eigenvectors, respectively, mean for this population?

4. Now it's time to simulate the model to verify our theoretical predictions (eigen-analysis). To do so, assume that the initial number of individuals of each age class comes from a random uniform distribution (see function `runif`). Now simulate the dynamics of the model for `t=20` time steps. Note that you will need to perform matrix multiplication for this step via the `%*%` operator in `R`.
5. Plot the proportional abundance of each age as a function of time by using the `matplot` function (allows you to plot multiple lines at the same time). Do not forget to add axis labels and a legend to your figure. Also add the population structure predicted by the eigen-analysis as vertical dashed lines using function `abline`. Do your simulation results match your theoretical expectations?
6. To estimate the population growth observed in your simulations empirically, use function `lm` to regress the log of total population size $\log(N)$ against time. This will give you an estimate of the growth rate of the population. Does this result match your theoretical expectation based on eigen-analysis?

Task 2: Simulating the effects of demographic and environmental stochasticity

For this section, you will estimate the effects of demographic and/or environmental stochasticity on the behavior of age-structured population models. To do so, you will need to download and install the `popbio` package:

```
# Only install the package once:
install.packages("popbio")
```

Note that this should only be done once per computer. Once the package is installed you can simply load it by issuing the following command:

```
# Load the package every time you need it in your R session:
library(popbio)
```

1. Begin by using function `eigen.analysis` to determine sensitivity and elasticity of λ to each model parameter. Identify the most important parameter and justify your answer.
2. We are now going to conduct an experiment by making F_1 vary over time and across simulations. To do so, we will assume that F_1 comes from a normal distribution with a mean of 5 and a standard deviation that varies from 0 to 2 in 100-linearly spaced increments (see function `rnorm`). For each of the 100 standard deviation values, we will draw $t = 50$ random values of F_i. Each one of these values will be used for a single time step t of the simulation. Assume that all other Leslie matrix parameters remain the same. We will need to keep track of multiple metrics. First, we will compute the (i) arithmetic mean of λ, (ii) geometric mean λ, and (iii) regression-based estimate of λ obtained via eigen-analysis at each time step for each simulation. Second, we will need to record the final proportional abundance at the end of each simulation. Note that you will have to create a function to compute the geometric mean, which is defined as: $\left(\prod_{i=1}^{N} \lambda_i\right)^{1/N}$.
3. On a 2-row x 1-column figure, plot the three estimates of λ (y-axis) as a function of the standard deviation of F_1 in the top panel and the final proportional abundance (y-axis) as a function of the standard deviation of F_1 in the bottom panel. Did environmental stochasticity affect the behavior of the model and, if so, how?

4. To focus on the different estimates of λ, plot both the geometric mean of λ in red and the arithmetic mean of λ in blue against the regression-based estimate of λ (x-axis). Add the 1:1 line (dashes), a legend and the proper axis labels. Using this figure, compare the arithmetic and geometric mean estimates of λ to the empirical, regression-based estimate. Can you explain the patterns based on the model?

5. To confirm these graphical results, use function `lm` to regress the arithmetic and geometric means of λ against the regression-based estimate. Compare the slopes and the goodness-of-fit R^2 to determine which mean is a better approximation of the regression-based estimate.