HW 1: Primer on Leslie matrices

Leslie matrix models are discrete-time matrix population models that emphasize age class distributions. An $m \times m$ Leslie matrix has the following structure:

$$
L = \begin{bmatrix}
F_0 & F_1 & F_2 & \dots & F_{m-1} \\
P_0 & 0 & 0 & \dots & 0 \\
0 & P_1 & 0 & \dots & 0 \\
\vdots & \vdots & \vdots & & \vdots \\
0 & 0 & \dots & P_{m-2} & 0
\end{bmatrix}
$$

where each F_i is the fecundity rate of age class i and each P_i is the probability that an individual in class i survives to the next age class. An $m \times m$ matrix models m distinct age classes. The Leslie matrix model assumes that in a single time period (say, a year), all individuals either advance to the next class (with probability P_i) or perish, and that no individual survives past the last age class.

Consider a column vector $\mathbf{n}_t$ giving the number of individuals in each class at time t:

$$
\mathbf{n}_t = \begin{bmatrix}
n_0 \\
n_1 \\
n_2 \\
\vdots \\
n_{m-1}
\end{bmatrix}_t.
$$

To find the number of individuals in each class at time $t + 1$, you simply perform matrix-vector multiplication (np.dot in numpy):

$$
\mathbf{n}_{t+1} = L\mathbf{n}_t
$$

Note that several time steps can be taken at the same time by taking powers of L; for instance:

$$
\mathbf{n}_{t+2} = L(L\mathbf{n}_t) = L^2\mathbf{n}_t.
$$

One fact about Leslie matrices is that the eigenvalue with the largest magnitude, known as the dominant eigenvalue, can tell you whether or not the total population will grow or shrink in time. If the dominant eigenvalue has a magnitude greater than 1, the population as a whole will grow. Less than 1 and it will shrink.

Note that in Python, you can get both the eigenvalues and eigenvectors of a matrix using numpy.linalg.eig. See the documentation on this function for more information. Also note that in Python, $L ** 2$ will result in each element of L being squared, *not* the matrix-matrix multiplication you want here!

To complete your homework, follow the instructions in the starter code!