

BIO311: Population Ecology  
*Prac 8: Life tables & Population Matrices*

Timothée Bonnet &\* Koen van Benthem

`timothee.bonnet@ieu.uzh.ch`  
`koen.vanbenthem@ieu.uzh.ch`

Spring 2016

## Contents

<b>1</b>	<b>Hypothetical Dataset</b>	<b>2</b>
<b>2</b>	<b>Rotifer data</b>	<b>13</b>

---

\*This document was co-authored by Tina Cornioley

# 1 Hypothetical Dataset

## 1.1 Life table

In this practical, you will work with life tables. Read carefully the following text and do the exercise at the end of this section. The first part is the description of a hypothetical barn owl population and the second part is brief reminder of life table theory.



Figure 1: Barn owl, from wikipedia.

### 1.1.1 A barn owl population

The barn owl (Schleiereule in German) is a common nocturnal prey bird (see fig. 1). We focus on a cohort of female barn owls that were born in 2010 and who have all died by the beginning of 2014.

- In 2010, 10 female chicks were born and were followed over the years.
- In 2011, 8 of them were still alive. Only 4 females of this cohort reproduced in 2011: two produced 4 female chicks, one produced 3 female chicks and the last one produced 5 female chicks.
- In 2012 only 4 females were still alive. This time, they each produce 3 female chicks.
- In 2013, only one individual of the original cohort was still alive and had only one female chick.
- In 2014, all individuals were dead.

### 1.1.2 Standard cohort life table

Table 1 contains the standard life-table calculation. Below is a description of each column. We purposefully do not provide you with the equations to obtain each value, we left this for you to find in the exercise.

Table 1: An example of a life table.

$x$	$S(x)$	$b(x)$	$l(x)$	$g(x)$	Rep. rate
0	10	...	...	...	...
1	8	...	...	...	...
2	...	...	...	...	...
3	...	...	...	...	...
4	...	...	...	...	...

- *The first column  $x$*  is the age of the individuals. By convention, the newborn start at age 0 and are age 1 at their first birthday.
- *The second column  $S(x)$*  is the number of individuals of the cohort still alive at age  $x$ .
- *The third column  $b(x)$*  is the fecundity schedule (sometimes it is referred to as  $m(x)$ ,  $m$  for maternity). It is the **average number of female offspring born per unit of time to a female of a given age**.
- *The fourth column  $l(x)$*  is the survivorship schedule, not to be confused with the survival probability  $g(x)$ . It is defined as the **proportion of the original cohort that survives to a given age**. Or in other words it is the probability that an individual survives from birth to age  $x$ .
- *The fifth column  $g(x)$*  is the survival probability. It is different from the survivorship schedule in that it is the **probability that an individual of age  $x$  survives to age  $x + 1$** .
- *The sixth column Rep. rate* is what we call the reproductive rate: what is the mean number of offspring that each individual produce at a given age, regardless of whether the individual survives until that age. We are in fact interested in the **net reproductive rate**,  $R_0$  but to find it, we must first find the reproductive rate of each age. The net reproductive rate is defined as the **mean number of female offspring produced per female individual over their lifetime**. It is the reproductive potential of a female corrected for its mortality.

In addition to the elements of the table, there are a few more important properties you can extract from the life table. A general property of the table is,  $G$ , the generation time. A definition of the generation time is **the average age of mothers when they give birth**. From the life table, you can also calculate the instantaneous rate of increase,  $r$ . Remember from Madan's lecture on unstructured population that:

$$N_t = N_0 e^{rt} \quad (1)$$

The exact solution of  $r$  is given by solving the following equation adapted from the Lotka-Euler equation for  $r$ :

$$1 = \sum_{x=0}^k e^{-rx} l(x) b(x) \quad (2)$$

This equation can only be solved iteratively. Alternatively, an approximation of  $r$  is given by:

$$r \approx \frac{\ln(R_0)}{G} \quad (3)$$

### \* Exercise 1 Life Table

Write on paper a life table for the hypothetical barn owl population described above. Your life table should be organised like table 1 and contain the same columns. For each column, write the corresponding equation.

In addition, find  $R_0$ ,  $G$  and the approximation of  $r$ .

### 1.2 Age-structured Matrix Analysis

In the former section, we have followed the fate of a hypothetical cohort of barn owls. Let us now look at the transition matrix. We want to know the number of individuals in each age class. Imagine these 10 female barn owls are the founders of a new population and we do a postbreeding census, in which females are counted each year just after they have bred.

- In 2010, there are 10 females in age class 1.
- In 2011 there are 8 females in age class 2 and 16 females in age class 1.
- In 2012 there are 4 females in age class 3, 13 females in age class 2 and 12 females produced by the individuals in age class 2 and 26 females produced by the females in age class 1.
- In 2013 there is 1 female in age class 4, 7 in age class 3 and 30 in age class 2. There are in addition 81 new individuals in age class 1: 1 from the females in age class 3, 19 from the females in age class 2 and 61 from the females in age class 1. The survival rate of the age class 4 is 0.

### \* Exercise 2 Age-structured matrix

1. Let us work out the matrix of this barn owl population on paper.
  - a) First, draw the life cycle of the population of barn owls describe above.
  - b) Find the survival probability  $P_i$  for each age class  $i$ , that is the probability of an individual in age class  $i$  to age class  $i + 1$ .
  - c) Find the fertility  $F_i$ , the average number of offspring produced by an individual of age class  $i$ .

- d) Write the equations to get the number of individuals in each age class at time  $t + 1$ . Your equations should be of the form  $n_1(t + 1) = F_1 n_1(t)$ .
- e) What is the structure of the corresponding Leslie matrix? What is the dimension of your matrix?
- f) Build the corresponding Leslie matrix of this population.

### 1.2.1 Matrices in R

Now that you have found the matrix of this population, let us implement it in R. Set the initial population vector to the founder population (10 individuals in age class 1).

```
# Build the Leslie Matrix for the barn owl population

F1<- #insert your value here
F2<- #insert your value here
F3<- #insert your value here
F4<- #insert your value here

P1<- #insert your value here
P2<- #insert your value here
P3<- #insert your value here
P4<- #insert your value here

A<-matrix(c(F1,P1,0,0,F2,0,P2,0,F3,0,0,P3,F4,0,0,P4),
          nrow=4,byrow=FALSE)
A
# The initial population vector

n11<- #insert your value here
n12<- #insert your value here
n13<- #insert your value here
n14<- #insert your value here

n0<-c(n11, n12, n13, n14)
```

What happens if you set `byrow=TRUE` instead?

The next step is to project the population in the future. We have seen in the lecture that

$$\vec{n}(t + 1) = \mathbf{A}\vec{n}(t) \quad (17)$$

To do so, we write the two following functions in R. The sign `%*%` is for matrix multiplication in R.

```
n1<-n0%%A
nt2<-A%%n0
```

Try to run both lines. One is incorrect, which one and why?

Next let us use this matrix to project the population. To do so we use a `for` loop over a specified time span. We first need to build an empty array with the `matrix` function where we store the data that we calculate during the loop. We call this array `n`, it is an array that has the same number of rows that the matrix `A` and has as many column as the time span we set. Thus each column corresponds to one year. `n` will look like this:

$$\begin{pmatrix} N_{age=1,t=1} & N_{age=1,t=2} & N_{age=1,t=3} & \dots & N_{age=1,t=T} \\ N_{age=2,t=1} & N_{age=2,t=2} & N_{age=2,t=3} & \dots & N_{age=2,t=T} \\ N_{age=3,t=1} & N_{age=3,t=2} & N_{age=3,t=3} & \dots & N_{age=3,t=T} \\ N_{age=4,t=1} & N_{age=4,t=2} & N_{age=4,t=3} & \dots & N_{age=4,t=T} \end{pmatrix}$$

By first setting the time span to 4, check if we get the same population as the one described in the text. Then try different values of time span. To access column 5 of this matrix, you can type `n[,5]`. Also in R the first index specifies the row number and the second one the column number. Because we want to see all rows at the same time, we leave out an index for the row. (`n[3,5]` would for example only give the entry for age class 3 at time 5.)

Complete the following code to calculate how the population develops over time

```
tspan <- ... #insert value for time span
rows <- ... # insert number of rows

# Build some matrices for storing eventual output
n <- matrix(0,nrow=rows,ncol=tspan) # empty matrix

n[,1] <- c(...,...,...,...) # insert value for the initial
                             # population in each age class

# Project population forward and store output
for (t in ...){
  n[,t] <- ..... # %*% = matrix multiplication in R
}
```

The sum of each column is the total population size. We can then plot it.

```
Ntot<-colSums(n) # This calculates the sum
                 # of all the age classes per time step
plot(Ntot, type="l", xlab="year", ylab="N(t)")
```

We see that the population is exponentially growing.

### 1.2.2 Asymptotic behaviour

#### *Asymptotic rate of increase*

From the matrix, we can find the asymptotic growth rate of that population. For this we need to look at the dominant eigenvalue of the population matrix. Remember from the lecture that the dominant eigenvalue of a population matrix is the long term asymptotic rate of increase,  $\lambda$ . An eigenvalue of a matrix is a **scalar that when multiplied by a specific vector, gives the same result as when that vector is multiplied by the concerning matrix**. Mathematically this means:

$$A\vec{w} = \lambda\vec{w} \quad (18)$$

the vector  $\vec{w}$  is not any vector but an eigenvector, that is a vector such that equation 18 holds. We will come back to it later. If you do not feel comfortable with the concept of eigenvectors/matrices, you might want to return to the last exercise of the mathematical tools script.

In R we do not have to just see the eigenvectors, instead they can be found using the `eigen()` function.

```
eigens.A<-eigen(A)
eigens.A
```

Let us focus at the moment only on the first values. The first values returned by the function; 2.176, -0.463, -0.103, 0 are the eigenvalues of the matrix **A**. A matrix has as many eigenvalues as it has dimensions. The only one we are interested in is the dominant eigenvalue, i.e. the largest eigenvalue. From looking at the eigenvalues, we can see that it is the first one. However, the dominant eigenvalue can be found by using the function `which.max()` that returns the position of the maximum value of the object. Then we extract the value in that position.

```
position<-which.max(eigens.A$values)
position

lambda2<-eigens.A$values[position]
lambda2
```

Alternatively,  $\lambda$  can be found directly from the `lambda` function of the `popbio` package in R, which will give you exactly the same result.

```
library(popbio)

## Loading required package: quadprog

lambda(A)

## [1] 2.176064
```

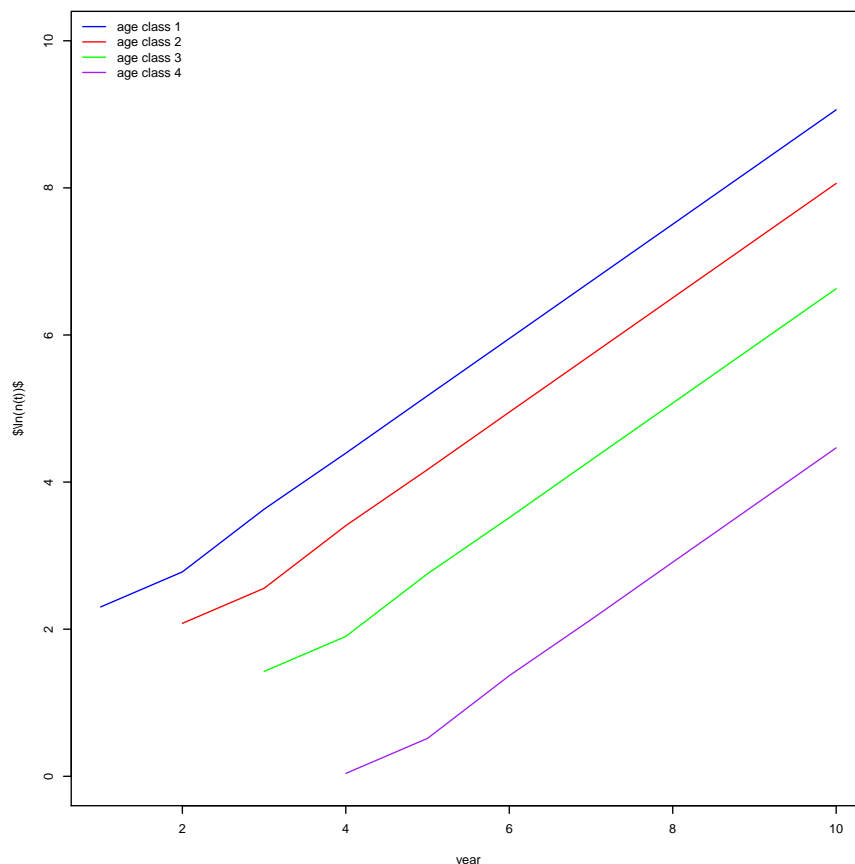
Thus our asymptotic finite rate of increase  $\lambda$  is 2.1760644. So we know what the population growth rate looks like on the long-term.

#### *Stable-age distribution*

In the previous section, we looked at the growth of the whole population. It is also informative to examine what the population structure will look like on the long-term. We already calculated these numbers before, let us now plot them.

```
plot(log(n[1,]), type="l", col="blue", ylim=c(0,5),
      xlab="year", ylab="ln(n(t))")
lines(log(n[2,]), col="red")
lines(log(n[3,]), col="green")
lines(log(n[4,]), col="purple")
legend("topright",
      c("age class 1 ", "age class 2", "age class 3",
        "age class 4"),
      bty = "n",
      lty=c(1,1,1,1),
      col=c("blue", "red", "green", "purple"))
```





When reading the graph, keep in mind that the  $y$ -axis is logarithmic. We see that after five years, the age class trajectories are parallel to each other and growing with a constant rate. If a population is growing with constant birth and death rate, as is the case of our owl population, then the population converges to a stable age distribution. Once a population has reached the stable age distribution, it grows exponentially at the asymptotic finite rate. A population at the stable age distribution always has the same relative number of individuals in each age class every time step. However, the absolute number of individuals will change. The stationary age distribution is a special case of stable age distribution where the absolute number of individuals does neither increase nor decrease. The population is not growing. What is the asymptotic growth rate ( $\lambda$ ) of such a stationary population?

The stable age distribution is given by the right eigenvector corresponding to the dominant eigenvalue. Let us now look at the vector elements of the `eigen` function. The columns of `eigen.A$ vector` are the right eigenvectors of the matrix `A`. When applying the matrix to them, it is the same as multiplying them by the corresponding eigenvalues, which are scalars. This means that the

length but not the direction of this vector is changed by the matrix. Check this for the eigenvector corresponding to the dominant eigenvalue.

There are as many right eigenvectors as there are eigenvalues. To each eigenvalue one eigenvector corresponds. Its position is the same as the one of the dominant eigenvalue so we can extract it with the following lines of code.

```
w<-eigens.A$vector[,position]
w
```

The right eigenvector as such does not give us the proportion found in each age class. This is because the length of the vector is set to 1 instead of the sum of its elements (what is the difference between these two? Which is biologically more relevant? Which is mathematically more relevant?). However, to find the proportion of each age class, we want to rescale it (remember that if you find an eigenvector all vectors that point in the same direction are also eigenvectors. If we multiply an eigenvector by a scalar it thus remains an eigenvector.):

```
ssd<-w/sum(w)
ssd
```

Therefore we know now that a proportion of 0.68 of the population is in age class 1, 0.25 in age class 2, 0.06 in age class 3, and 0.01 in age class 4.

### 1.2.3 Sensitivity and elasticity analysis

*Sensitivity analysis* Sensitivities and elasticities of  $\lambda$  evaluate the relative importance of each matrix element for the asymptotic growth rate. Sensitivity is the effect of additive change in a matrix element on  $\lambda$  and the elasticity is the proportional effect of a proportional change in the parameter to  $\lambda$ .

The sensitivity tells us how an increase in absolute value of one of the parameters affects  $\lambda$ . How much does  $\lambda$  increase if the fertility of the age class 1 increases by a very small amount (for example 0.01)? Let us play around with the projection matrix and change each element of this matrix by 0.01 one by one to see how it affects  $\lambda$ .

```
# Build the Leslie Matrix for the barn owl population

F11<-      + 0.01 #insert your value here
F21<- #insert your value here
F31<- #insert your value here
F41<- #insert your value here

P11<- #insert your value here
P21<- #insert your value here
P31<- #insert your value here
P41<- #insert your value here
```

```
A1<-matrix(c(F11,P11,0,0,F21,0,P21,0,F31,0,0,P31,F41,0,0,P41), nr=4)
A1
lambdaF1<- lambda(A1)
```

The sensitivity of  $\lambda$  to one element of the matrix is given by:

$$s_{ij} = \frac{\partial \lambda}{\partial a_{ij}} \quad (19)$$

So the sensitivity can be approximated by taking the difference between the value of  $\lambda$  of the original matrix and the modified matrix and dividing it by the change we apply to the element of the matrix.

- $\partial \lambda$  can be approximated by the difference between the original  $\lambda$  and the new  $\lambda$ .
- $\partial a_{ij}$  can be approximated by the difference between the original  $a_{ij}$  and the new  $a_{ij}$

If the change to the element of the matrix is very small, the sensitivities calculated so will be close from the actual sensitivities. Below you can find the actual sensitivities and compare them with the ones you can find using this approximation. If you want to calculate all sensitivities using this approximation, it is probably best to use a `for` loop (or actually two, why two?). Do not forget to only change one matrix element at a time. Also investigate how small the change in the matrix element has to be to make sure that the approximation returns reasonable numbers.

$n_1$	$n_2$	$n_3$	$n_4$
0.7873	0.2894	0.0692	0.0079
0.5571	0.2048	0.0489	0.0056
0.0904	0.0333	0.0079	0.0009
0.0000	0.0000	0.0000	0.0000

*Elasticity analysis* Elasticities are proportional sensitivities. Before we asked the question: if a matrix element increases by a small amount, how much does  $\lambda$  increase? Elasticities ask a different question: with what factor does  $\lambda$  change if a matrix element is changed by a certain factor. They thus examine the effect of a proportional change of transition elements on  $\lambda$ . For example, you may be interested in knowing by how much does  $\lambda$  change if the fertility of age class 2 is increased by 25%.

The elasticity of  $\lambda$  to a proportional change of one element of the matrix is given by:

$$e_{ij} = \frac{a_{ij}}{\lambda} \frac{\partial \lambda}{\partial a_{ij}} \quad (20)$$

You can thus calculate them from your approximations of the sensitivities. Compare your values with the values below, which are calculated by the real solution to this problem.

$n_1$	$n_2$	$n_3$	$n_4$
0.5825	0.1968	0.0079	0.0000
0.2048	0.0000	0.0000	0.0000
0.0000	0.0079	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000

#### 1.2.4 Transient dynamics

Most of what we have described above focuses on the asymptotic, (i.e. long-term) behaviour of the system. However, the short-term dynamics can be very different. In nature, a population may never be observed in the asymptotic regime. Transient dynamics focuses on short-term responses.

There are year-to-year variations in growth rate before the asymptote is reached. The code below calculate the annual finite growth rates of the barn owl population.

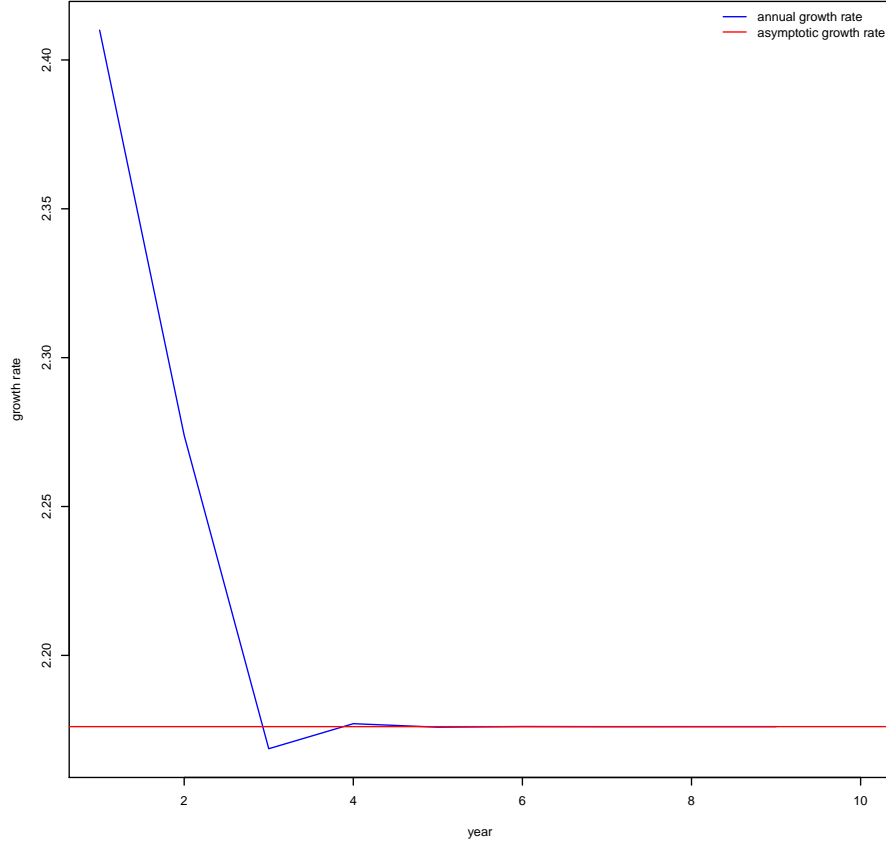
```
Ntot # the total population size vector.
      # Each element is the population size
      # at one time step. As calculated before
Ntott<-Ntot[2:(length(Ntot)+1)]
# the total population size at t+1

R<-Ntott/Ntot
# the annual finite growth rates
#vector
```

We can plot this in a graph to see how much time it takes this population to reach the asymptotic finite growth rate. For this we first create a vector of the same length as the annual growth rate vector in which all elements contain the value of  $\lambda$ .

```
lam<-lambda(A)

plot(R, type="l", col="blue", xlab="year", ylab="growth rate")
abline(h=lam, col="red")
legend("topright",
      c("annual growth rate","asymptotic growth rate"),
      bty = "n",
      lty=c(1,1),
      col=c("blue", "red"))
```



It thus takes around five years before the population reaches its asymptotic growth rate.

## 2 Rotifer data

We will discuss this topic in class.

### Answer of exercise 1

The column  $x$  and  $S(x)$  are directly given in the text.

$$b(x) = \frac{n(x)}{S(x)} \quad (4)$$

where  $n(x)$  is the number of female offspring produced by all the females of age  $x$ .  $n(x) = \sum_{i=1}^k n(i)$ , where  $n(i)$  is the number of female offspring per female of age  $x$  in the cohort and  $k$  the number of female age  $x$  in the cohort.

$$l(x) = \frac{S(x)}{S(0)} \quad (5)$$

$$g(x) = \frac{S(x+1)}{S(x)} \quad (6)$$

$$Rep.rate = l(x)b(x) \quad (7)$$

$$R_0 = \sum_{x=0}^k l(x)b(x) \quad (8)$$

where  $k$  is the maximum age.

$$G = \frac{\sum_{x=0}^k l(x)b(x)x}{\sum_{x=0}^k l(x)b(x)} \quad (9)$$

It may therefore be interesting to add an additional column to the table:

$$l(x)b(x)x = l(x)b(x)x \quad (10)$$

$x$	$S(x)$	$b(x)$	$l(x)$	$g(x)$	Rep. rate	$l(x)b(x)x$
0.00	10.00	0.00	1.00	0.80	0.00	0.00
1.00	8.00	2.00	0.80	0.50	1.60	1.60
2.00	4.00	3.00	0.40	0.25	1.20	2.40
3.00	1.00	1.00	0.10	0.00	0.10	0.30
4.00	0.00	0.00	0.00		0.00	0.00

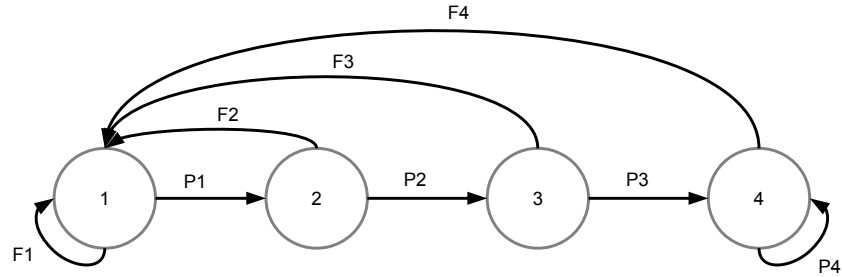
$$R_0 = 2.9$$

$$G = 1.482759$$

$$r = 0.7180607$$

## Answer of exercise 2

1. a)



```

b) #Finding the survival rate of age class 1
age1<-c(10, 16, 38) #individuals in age class 1
                        # in time t
age2<-c(8, 13, 30) #individuals in age class 2
                        #in time t+1 (i.e. the surviving
                        #individuals from age class 1 in t)

p1<-age2/age1 #yearly survival rate

P1<-mean(p1) # mean of survival rate
                #of age class 1, P1

#Finding the survival rate of age class 2
age2<-c(8, 13) #individuals in age class 2
                        #in time t
age3<-c(4, 7) #individuals in age class 3
                        # in time t+1

p2<-age3/age2 #yearly fertility rate

P2<-mean(p2) #yearly survival rate

#Finding the survival rate of age class 3
P3 <- 1/4      #individuals in age class 4
                        # in time t+1 over the number
                        #of individuals in age class
                        #3 in time t

#Finding the survival rate of age class 4
P4 <- 0

```

```

c) #Finding the contribution of age class 1
   #to the next generation

age1<-c(10, 16, 38) #individuals in age class 1
                        # in time t
offa1<-c(16, 26, 61) #offspring in time t+1
                        #produced by age class 1

f1<-offa1/age1 #yearly fertility rate

F1<-mean(f1) #mean of fertility, F1

#Finding the contribution of age class 2 to
#the next generation
age2<-c(8, 13) #individuals in age class 2
                # in time t
offa2<-c(12, 19) #offspring in time t+1
                #produced by age class 2

f2<-offa2/age2 #yearly fertility rate

F2<-mean(f2) #mean of fertility, F2

#Finding the contribution of age class 3 to
#the next generation
F3 <- 1/4 #offspring in t+1 produced by individuals
          #in age class 3 at t

#F4 is zero
F4<-0

```

d)

$$n_1(t+1) = F_1 n_1(t) + F_2 n_2(t) + F_3 n_3(t) + F_4 n_4(t) \quad (11)$$

$$n_2(t+1) = P_1 n_1(t) \quad (12)$$

$$n_3(t+1) = P_2 n_2(t) \quad (13)$$

$$n_4(t+1) = P_3 n_3(t) \quad (14)$$

e)

$$\begin{pmatrix} F1 & F2 & F3 & F4 \\ P1 & 0 & 0 & 0 \\ 0 & P2 & 0 & 0 \\ 0 & 0 & P3 & P4 \end{pmatrix} \quad (15)$$



f)

$$\begin{pmatrix} 1.61 & 1.48 & 0.25 & 0 \\ 0.8 & 0 & 0 & 0 \\ 0 & 0.52 & 0 & 0 \\ 0 & 0 & 0.25 & 0 \end{pmatrix} \quad (16)$$