Solution of assignment 9, ST2304

Problem 1 The Ricker model

1. We see that when $N_t = K$, the expression $e^{r(1-N_t/K)}$ becomes 1 and N_{t+1} equals N_t. This indicates that the population size does not change from time t to $t+1$.

   For N_t much smaller than K, the population size N_t changes with factor e^r each timestep ($(1 - N_t/K)$ will approach 1). This will give an exponential growth of the populations size (no restriction of carrying capacity, K).

2. It follows that the change in population size

$$\Delta N_t = N_{t-1} - N_t = N_t(e^{r(1-\frac{N_t}{K})} - 1) \tag{1}$$

   If we choose $r = 0.1$ and $K = 100$, a graph of ΔN_t can be made using the curve function. Notice that ΔN_t has its largest value when $N_t = K/2$ and equals 0 when $N_t = K$.

```
r <- .1
K <- 100
curve(x*(exp(r*(1-x/K))-1),from=0,to=120,
   xlab=expression(N[t]),ylab=expression(N[t+1]-N[t]))
```

3. Writing a function which computes the population size from time $t = 2$ to $tmax$, given the start population size N_1, the intrinsic growth rate r and the carrying capacity K.

```
Nfunc <- function(r,N1,K,tmax) {
   N=rep(NA,tmax)
   N[1]=N1
   for(i in 2:tmax) {
      N[i]= N[i-1]*exp(r*(1-(N[i-1]/K)))
   }
   plot(1:tmax,N, ylab="Population size", xlab="Time")
   return(N)
}
Nt <- Nfunc(r=0.5,N1=100,K=150,tmax=10)
```

Problem 2 The solution to the Euler Lotka equation is the root of the function

$$f(\lambda) = \sum \lambda^{-i} l_i m_i - 1$$

To use Newton's method we need derivate of $f(\lambda)$

$$f'(\lambda) = \sum -i\lambda^{(-i-1)} l_i m_i.$$

The iteration equation then becomes

$$\begin{aligned}\lambda_{t+1} &= \lambda_t - f(\lambda_t)/f'(\lambda_t) \\ &= \lambda_t - (\sum \lambda_t^{-i} l_i m_i - 1)/(\sum -i\lambda_t^{(-i-1)} l_i m_i)\end{aligned}$$

which can be solved as follows in R.

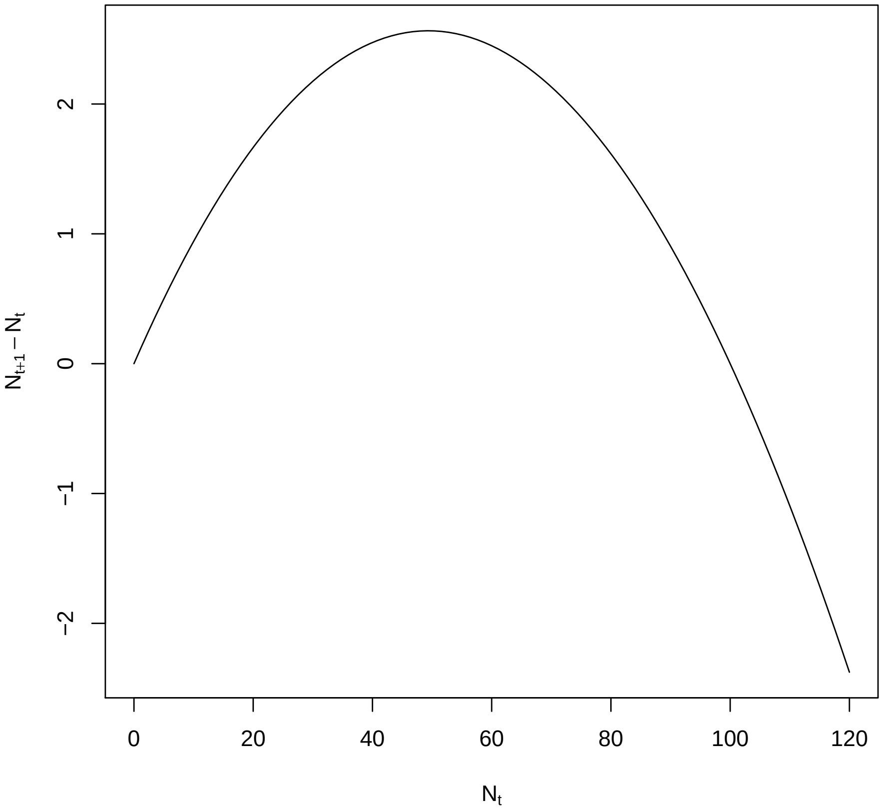

Figure 1: The chaqnge in population size as a function of last years population size, with parameter values $r = 0.1$ and $K = 100$

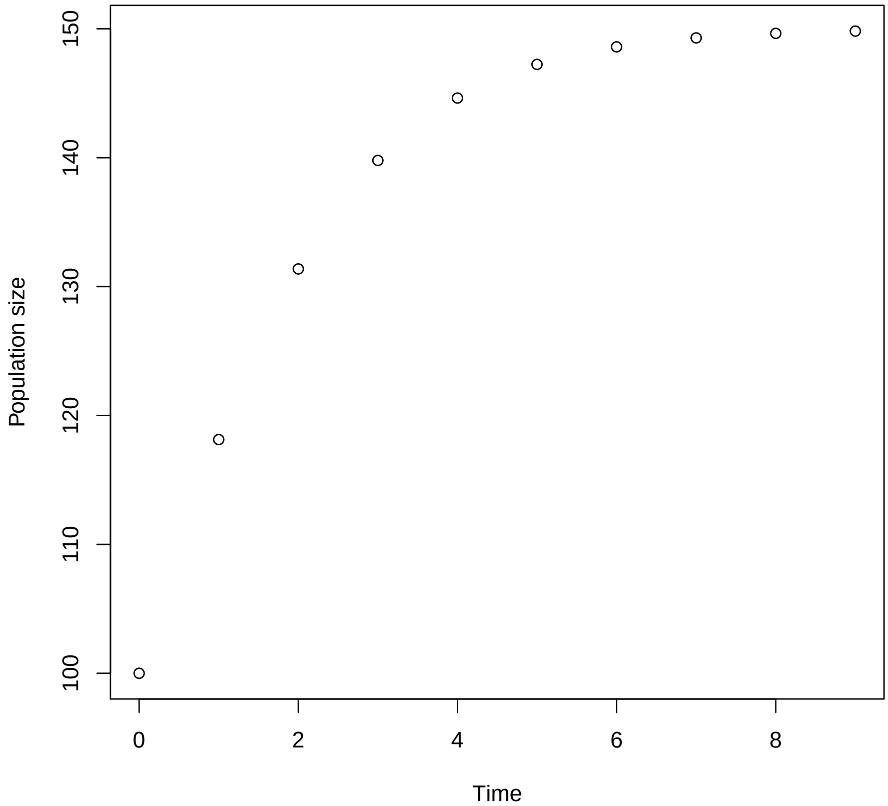

Figure 2: Population size against time, for parameter values r=0.5,K=150,N_1=100, $tmax$=10

```
eulerlotka  <-  function(m,l) {
  n  <-  length(m)
  i  <-  1:n
  lambda  <-  1
  while (abs(sum(lambda^(-i)*l*m)-1)>1e-8) {
    lambda  <-  lambda-(sum(lambda^(-i)*l*m)-1)/sum(-i*lambda^(-i-1)*l*m)
  }
  lambda
}
eulerlotka(c(.9,.8,.25),c(0,0,32))
```

Note how the iterations are repeated as long as $f(\lambda)$ in absolute value is larger than the desired accuracy of the solution.