Fiche TD avec le logiciel R : TP1-newEnglish

INSA 4-BIM
TP1 : Recursion equation in $\mathbb{R}$

Christelle LOPES
christelle.lopes@univ-lyon1.fr

Table des matières

1 Introduction

The aim is to become familiar with R software to analyse discrete-time models. This first session concerns the study of the most famous population models characterised by a recursion equation.

2 Definition of the models

Discrete time population dynamic models are defined by a recursion equation :

$$N_{t+1} = f(N_t)$$

We will study here the most famous population models :

— The exponential model
$$N_{t+1} = rN_t$$

— The logistic model
$$N_{t+1} = rN_t\left(1 - \frac{N_t}{K}\right)$$

— The Gompertz model
$$N_{t+1} = -rN_t \ln\left(\frac{N_t}{K}\right)$$

where r and K are positive parameters.

Let N_0 be the initial condition.

3 Qualitative analysis of discret-time models

3.1 Fixed points

1. How do you obtain the fixed points of a discrete-time population dynamic model ?

2. Give the fixed points of the exponential model.

3. Give the fixed points of the logistic model.

4. And the ones of the Gompertz model.

3.2 Stability of fixed points

5. How do you determine the stability of the fixed points of a discrete-time model ?

6. Determine the stability of the fixed points of the exponential model.

7. Do the same for the logistic model.

8. And the Gompertz model.

4 Numerical simulations

4.1 Model implementation

We want to create generic functions (depending on the parameters and the initial condition) that calculate population densities according to the three population models for t from 0 to t_{max}. Each function, when applied, should return a table with time in the first column and population densities in the second column.

To avoid the use of loops *for()*, we can use recursive functions. A recursive function is a function that refers to itself. It can thus be used to calculate the n^{th} term of the equation $u_n = f(u_{n-1})$. In R, a recursive function can be defined as follows :

```
> #n is time, u0 the initial condition, func the recursive function
>
> frecurs <- function(n=0, u0, func)
+   {
+ if (n==0) return(u0)
+ else
+       {
+       return(frecurs(n=n-1, u0=func(u0), func))
+       }
+ }
```

1. Create a recursive function common to the three models *frecurs()* above. Then, create a specific function for each model (*i.e. func()* above) to calculate the corresponding densities, and apply the recursive function to each model. Make sure that all values of N_t are returned for $t = 0$ to $t = t_{max}$, not only at t_{max}.
   Verification for the exponential model with $r = 2$, $N_0 = 2$ and $t_{max} = 50$:

```
  t  N
1 0  2
2 1  4
3 2  8
4 3 16
5 4 32
6 5 64
```

   Verification for the logistic model with $r = 2$, $K = 10$, $N_0 = 2$ and $t_{max} = 50$:

```
  t        N
1 0 2.000000
2 1 3.200000
3 2 4.352000
4 3 4.916019
5 4 4.998589
6 5 5.000000
```

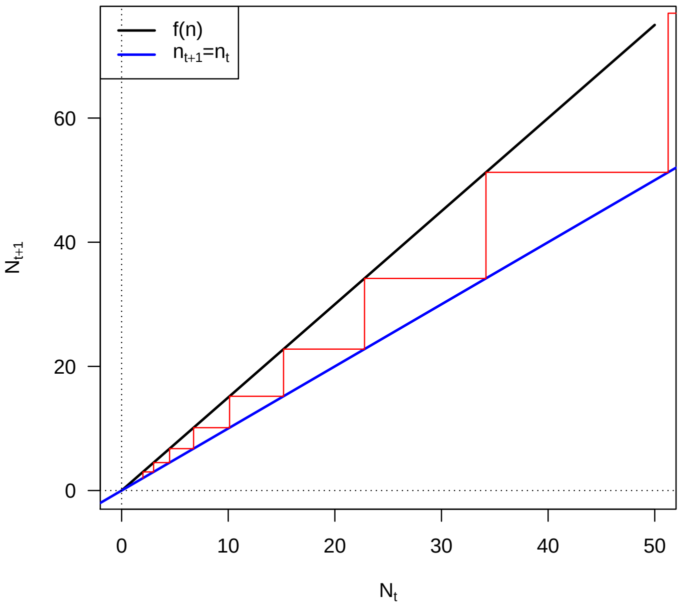
Verification for the Gompertz model with $r = 2$, $K = 10$, $N_0 = 2$ and $t_{max} = 50$:

```
  t        N
1 0 2.000000
2 1 6.437752
3 2 5.670446
4 3 6.433885
5 4 5.674771
6 5 6.430139
```

4.2 Cobweb (stair step) diagram

The cobweb representation (or stair step diagram) consists in representing the recursion equation $N_{t+1} = f(N_t)$ using the equation curves $N_{t+1} = f(N_t)$ and $N_{t+1} = N_t$. An example is given below for the exponential model :

```
> r0 <- 1.5
> tmax <- 50
> K0 <- 0
> Ni <- 2
> N=frecurs(t=tmax,r=r0,K=K0,N0=Ni,func=Mexpo)
> #frecurs is my recursive function,
> #and Mexpo the reccurence function of the exponential model
>
> curve(r0*x, from=0, to=50, n=length(N), las=1,
+       xlab=expression(N[t]), ylab=expression(N[t+1]),lwd=2)
> abline(a=0,b=1, col="blue",lwd=2)
> abline(h=0,lty=3)
> abline(v=0,lty=3)
> points(N, N, type="S", col="red")
> legend("topleft",c("f(n)",expression (paste(paste(n[t+1],"="),n[t]))),lty=1,col=c("black","blue"),lwd=2)
```



2. Do the same for the logistic and the Gompertz models with $N_0 = 2$, $r = 1.5$, $K = 10$ and $t_{max} = 50$. Comment the results.

4.3 Dynamics of the models

We can also represent model predictions by plotting N_t over time. For each of the three models :

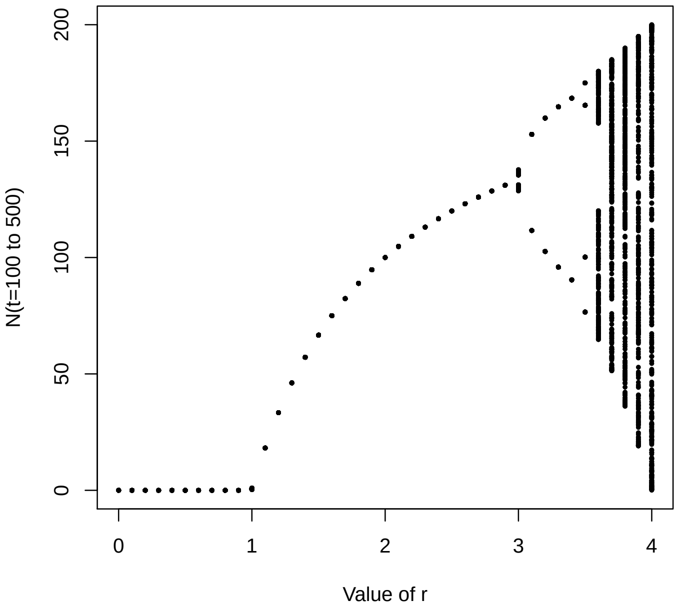
3. Plot (in 4 different graphics), the time-course of N_t, for t from 0 to 50, $N_0 = 10$, $K = 1000$ and different r values : $r = 1.5$, $r = 2.8$, $r = 3.1$ and $r = 3.7$. Be careful to represent the point estimates of your densities by points, we do not simulate a continuous-time model...

4. Test different initial conditions. Comment.

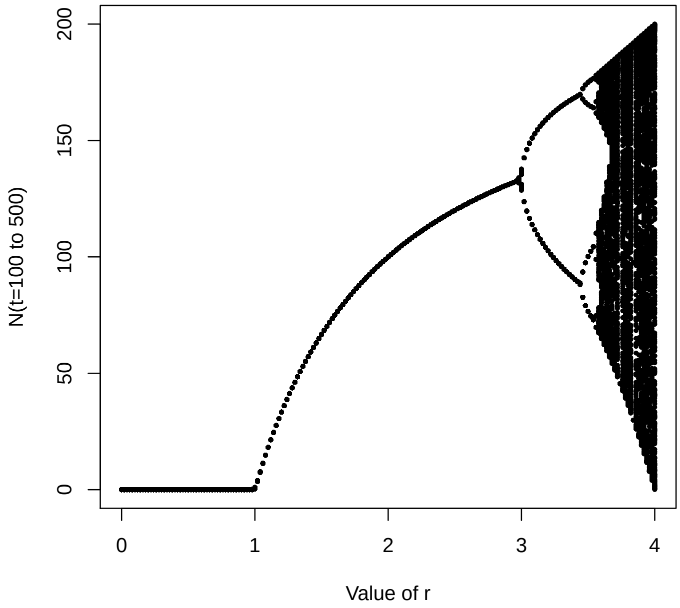
4.4 Bifurcation diagram

We can also represent the asymptotic behavior of a model according to the values of one of its parameter, and then plot a bifurcation diagram. Be careful, you have to simulate the model long enough to be sure you have reached the asymptotic regime.

We showed above that the predictions of the logistic model depend on the value of the population growth rate (r parameter). The bifurcation diagram of the logistic model is plotted below for $K = 200$ and r varying from 0 to 4 with a step of 0.1 :



5. Reproduce this diagram with ../config/figs/Rlogo.pdf
6. Which parameter should you change to obtain the following diagram ?



7. From this diagram, find the different types of dynamics and associate each type of dynamics with the value of r in the interval [0,4].

8. Zoom into the area where $3 < r < 4$ and give the values of r leading to a periodic dynamic. Give the period.

9. Represent the bifurcation diagram for $3.4 < r < 4.0$ and $3.847 < r < 3.857$. What do you notice ?

5 Towards a stable structure of chaos ?

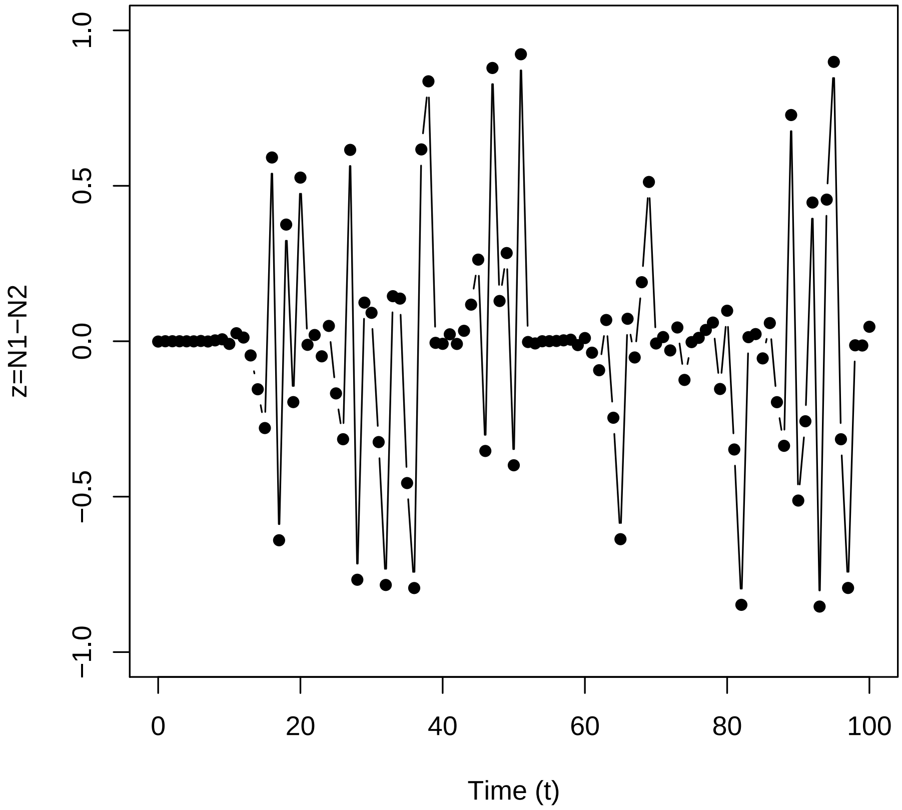
The chaotic dynamics predicted by models such as the discrete-time logistic model are part of what is known as deterministic chaos, which can be seen as a paradigm of what is behind by the word hazard. But then, how to use such models in prediction ? Would it be not possible to give mean predictions ? This is what we are going to look at in this part. For that purpose, we will study with more details the logistic model with $K = 1$ and $r = 3.98$.

The first characteristic of such chaotic dynamics is the dependence on initial conditions, that drastically limits a precise prediction.

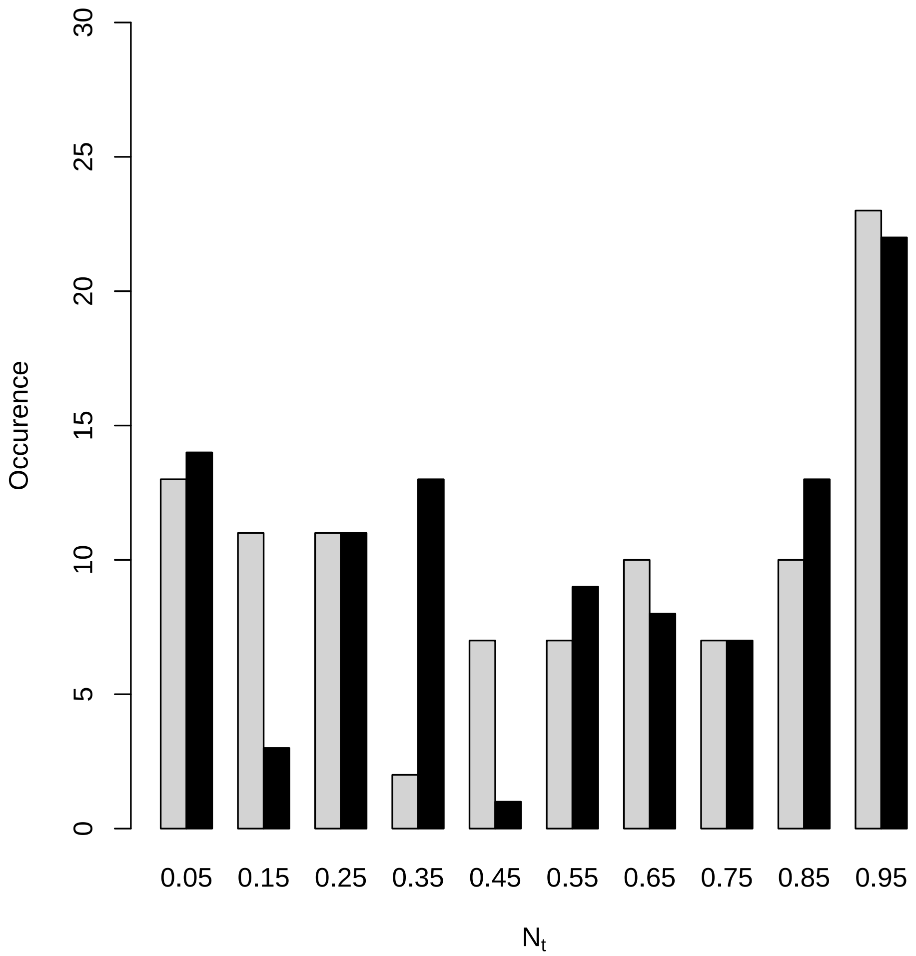
5.1 Sensitivity to initial conditions

1. Simulate the previous discrete-time logistic model over 100 time steps for two very close initial conditions : $N1_0 = 0.500$ and $N2_0 = 0.501$. Represent the dynamics in the same figure.

2. In order to more easily quantify the difference between the two initial conditions, represent $z_t = N1_t - N2_t$:



3. In the same histogram, represent the distribution of N_t values for the two initial conditions previously simulated over 100 time steps. This can be done using the *multhist()* function of the *plotrix* package.

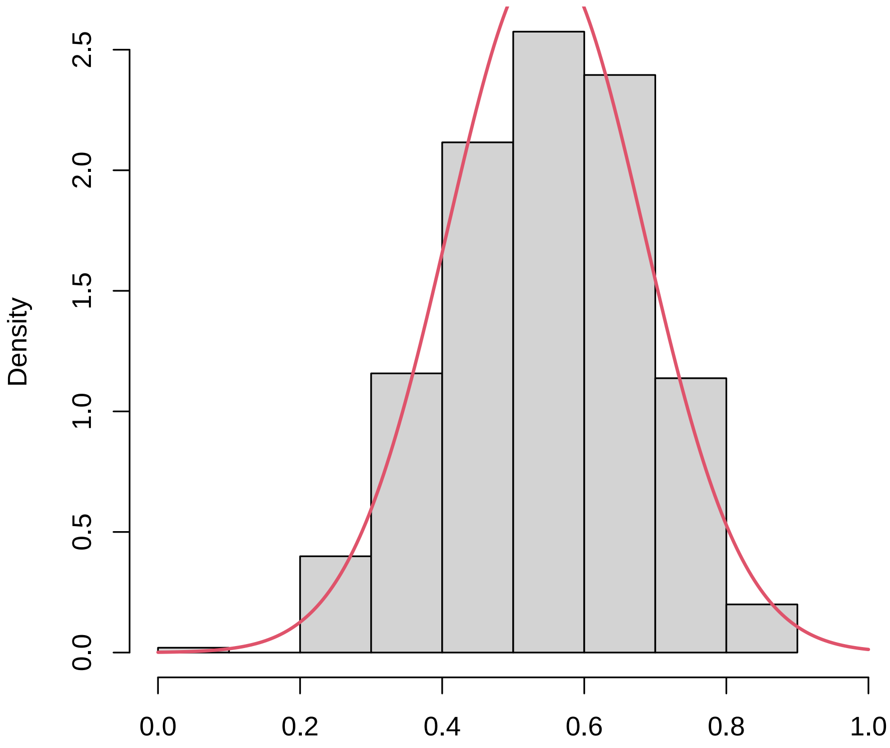


4. Statistically compare the two distributions. What do you notice ?

5. Simulate the model with very different initial conditions (from 0.1 to 0.9) on 1000 time steps and represent the distribution of the values in the same histogram. Compare the obtained distributions.

5.2 Towards a mean prediction

In fact, we are able to identify an asymptotic statistical distribution of our model predictions, that is independent on the initial conditions. It then becomes possible to make a "mean" prediction. Indeed, each tested initial condition represents a simulation of a chaotic variable. By an "original" application of the central limit theorem, Pavé (2006)[1] has thus shown that the marginal distribution of the sums of such chaotic variables, which are themselves chaotic, tends towards a Gaussian law :



6. Reproduce this figure through the following steps :
   (a) From the previous simulations, calculate the marginal sum of N_t for each t corresponding to the different initial conditions ;

1. Pavé, 2006. Modeling living systems, their diversity and their complexity : some methodological and theoretical problems. C. R. Biologies 329, p.3-12.

(b) Represent the distribution of these sums ;

(c) Calculate the mean, μ, and the standard deviation, σ, of these marginal sums ;

(d) In the same graphic, represent the curve representing the probability distribution of the Gaussian law of mean μ and standard deviation σ.

5.3 Towards the "control" of chaotic trajectories

As shown before, chaotic regimes are also sensitive to model parameter values.

7. Simulate the model for $N_0 = 0.5000$, $r_1 = 3.9800$ and $r_2 = 3.9801$.

You can see very different dynamics for very close values of r. This sensitivity to the parameter values can however be used to "drive" periodic trajectories to a chosen state (Shinbrot *et al.*, 1993) [2]. The idea is to use sensitivity to small disturbances to drive the system to a state chosen in advance. The following illustration is taken from Pavé (2012) [3] where these aspects are discussed and concrete examples presented.

We are still in the case of a density population N_t whose dynamic is described by the discrete logistic model (with $K = 1$). It is assumed that r has a nominal value of $r_0 = 3.6$ and that the current state is $N_t = 0.4$. We want to reach the neighborhood of $N_t = 0.8$ ($0.78 < N_t < 0.82$), and for this, we try to "play" with the r value to achieve this goal as soon as possible. We can thus build an adaptive control procedure, by writing the following model :
$N_{t+1} = (r_0 + \varepsilon)N_t(1 - \frac{N_t}{K}) = (r_0 + \varepsilon)N_t(1 - N_t)$ for $K = 1$.

8. For these conditions, determine the value of ε that allows to reach the goal as soon as possible and specify the number of time steps.

Thus, having a chaotic system allows to quickly reach a given goal, starting far from this goal and slightly modifying a parameter (or some parameters) of the model. Such "soft piloting" procedures can therefore be very interesting in different applied contexts (management of natural populations, renewable resources, etc.) but also in more fundamental approaches, related to evolution for example (origin of speciation, species extinction ...).

2. Shinbrot T., Grebogi C., Yorke JA, Ott E. (1993) Using Small Disturbances to Control Chaos, Nature 363, pp. 411-417.
3. Pavé, 2012. Modeling living systems, from the cell to the ecosystem. Ed. Lavoisier, pp. 633.