PREDATOR-PREY MODELLING

Report submission as a requirement for the module
“Problem Solving and Modelling”

Francesco Gruosso

UG Degree in Computer Science
at the School of Computing
Robert Gordon University
Aberdeen, Scotland
November 2020

An overview of the model

This model has been created to predict how the population of prey and predators change over time, according to different conditions. The existing equations for this model are:

Equation 1: $x_{n+1} = (1 + A)x_n - Bx_n y_n$

Equation 2: $y_{n+1} = (1 - C)y_n + Dx_n y_n$

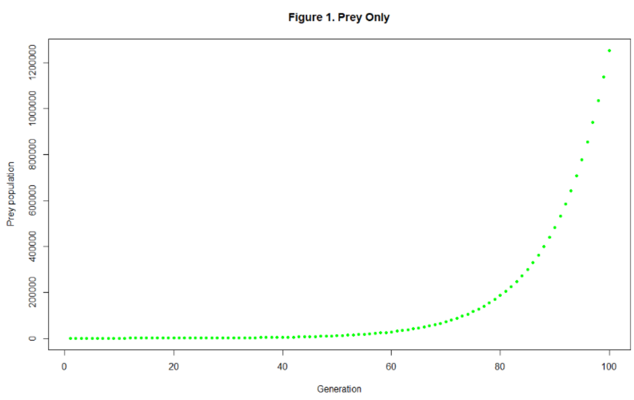
With X indicating the number of prey with a subscript n indicating its nth timestep. Y indicates the number of predators.

Prey only

In this version of the model, predators have been excluded from the environment. The initial population is $x_1 = 100$. As $Bx_n y_n$ describes the prey mortality rate in respects to the number of predators, it can be removed from the equation. The remaining values $x_{n+1} = (1 + A)x_n$ can be identified as a geometric sequence, with $(1 + A)$ being the common ratio for prey growth, and x_1 the first value of the sequence.

$$x_{n+1} = (1 + A)x_n$$

Consequently any step x_n is obtainable by applying r^$(n - 1)\ x_1$

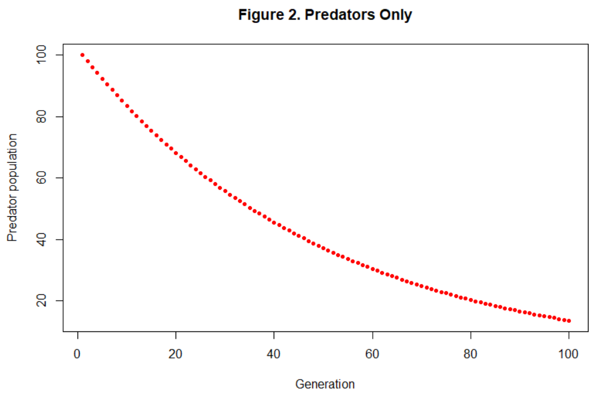



```r
# Given the values
A = 0.1
x1 = 100

# With a function that takes an argument n, the values can be easily changed
preyGeneration = function(n) (1 + A)^(n - 1) * x1

plot(preyGeneration(1:100), main="Figure 1. Prey Only", pch=20, col = "green", xlab = "Generation", ylab = "Prey population")
```

```r
# 100th generation (x100)
preyGeneration(100)

## [1] 1252783
```

The above model predicts that **in the absence of predators, the prey population would grow exponentially over time**, at a rate of 1.1 per generation (**Figure 1.**). This results into approximately 1252783 thousands of animals by the 100th generation. However, this model transitions into big numbers very quickly, as *it does not take into account other factors* such as various problems that come with animal overpopulation; Namely lack of space, food, and the damage caused to the ecosystem. These make this model's validity questionable.

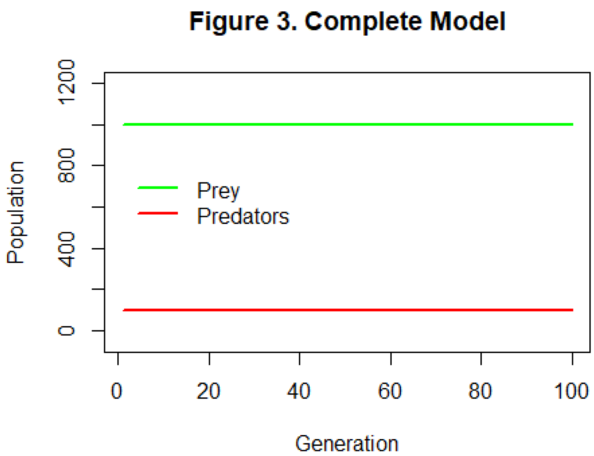
Predators only



Similarly to the previous model, this second model (**Figure 2**.) considers an environment with predator animals only. The initial population is $y_1 = 100$. In this case, $+Dx_n y_n$ represents the predators' growth rate and is therefore, excluded from the equation. $(1 - C)y_n$ represents the predators' mortality rate. The same logics and function can be applied:

```r
# Given the values
C = 0.02
y1 = 100

predatorGeneration = function(n) (1 - C)^(n - 1) * y1

plot(predatorGeneration(1:100), col = "red", main="Figure 2.
Predators Only", pch = 20, xlab = "Generation", ylab = "Predator
population")

# 100th generation (y100)
predatorGeneration(100)

## [1] 13.53261
```

In the absence of a prey population, a rapid and exponential decline of predators is predicted and expected, as there is an absence of prey and therefore food resources. By generation 100, the number of predators drops to approximately 13.53, and as the timeline progresses, the number of predators will stabilise near 0.



Complete model

With the full model, **considering both prey and predators in the same environment**, different outcomes of variation in population can be observed over time, this depends on the number of initial population of each species.

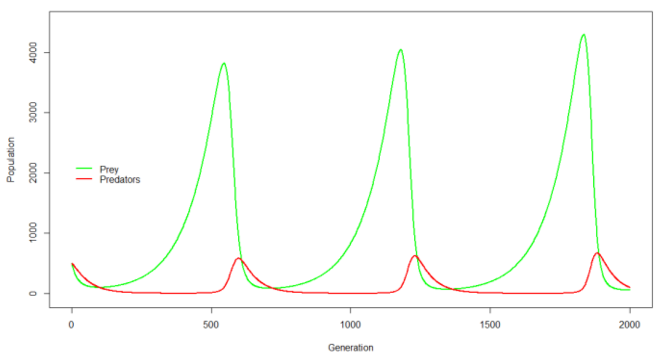
Various tests show that with ***initial prey population of x1 = 1000, and predator y = 100***, the number of animals remain approximately the same over time. This can be seen by running the model over 100 generations, confirming that **those initial values constitute a fixed point in the system**.

```r
# Given the values
A = 0.01
B = 0.0001
C = 0.02
D = 0.00002

N = 100 # Number of generations N
x = c(1:N) # Prey population
y = c(1:N) # Predator population

# Assign initial population to the first element in each vector
x[1] = C / D; # 1000
y[1] = A / B; # 100
```

```
# Loop through the range of wanted generations with a for loop,
  # and assign values to the appropriate vector
for(n in 1:(N - 1)) {
  x[n + 1] = (1 + A) * x[n] - B * x[n] * y[n]
  y[n + 1] = (1 - C) * y[n] + D * x[n] * y[n]
}
```

As opposed to this, considering a model with the same initial population x_n and y_n (for instance x_n = 500, $y_n = 500$), and a longer time frame, the data shows that the prey population prevents exponential growth of the prey, creating a **cycle** where predators eat most of the prey animals, also resulting in a decline in their own species. As a result of this (less predators in the environment), the prey population will see an increase, restarting the cycle (see **Figure 4.**).

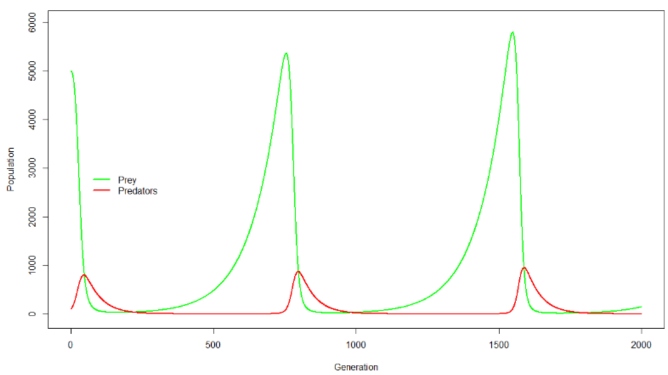
Figure 4. Complete Model - Matching initial population




```
# Number of generations N
N = 2000
# Initial population
x[1] = 500
y[1] = 500
```

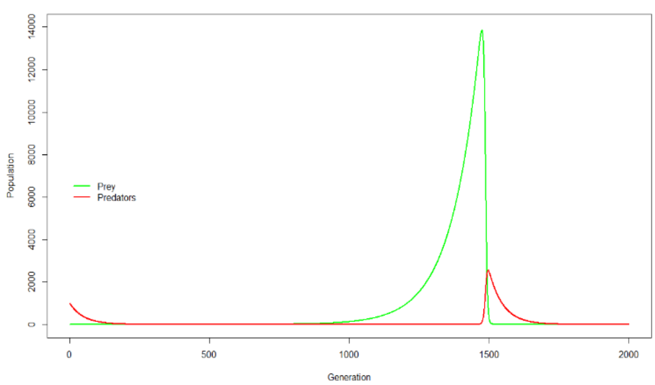
A similar cycle is obtained in most instances of the model, provided that enough time is given to the prey population to repopulate. A few relevant examples can be seen in **Figure 5.** and **Figure 6.**, where there is an abundance of prey and predators, respectively.

Figure 5. Complete Model - Abundance of prey




```
# Initial population
x[1] = 5000;
y[1] = 100;
```

Figure 6. Complete Model - Abundance of predators




```
# Initial population
x[1] = 1;
y[1] = 1000;
```

Due to the last examples, it can be argued that although this specific model could be used to predict the general direction of these species' population for the first few hundred generations, it should not be closely interpreted when applied to bigger time frames, as it only shows a generalised outline of what could happen in such environments.