¿Como se regulan las poblaciones animales?

I. Bartomeus, V. Dominguez

2023

¿Como funcionan las comunidades ecológicas?

Esta es una idea para ver que conceptos se podrían trabajar, no como se podrían trabajar. El como (e.g. como presentarlo, como programar, …) es otra guerra. El objetivo es aprender matemáticas, programación y ecología.

> De momento voy añadiendo ideas generales, sin documentar la guía a fondo. Trabajo de forma iterativa y esto puede ser considerada v0.1.

Preambulo

Yo empezaría por aprender algunos básicos para programar.

Aquí uso R y control de versiones. Se puede aprender los básicos de R (https://rforcats.net/) y control de versiones (www.happygitwithr.com) fácilmente, pero quizás es más sencillo usar Scratch (https://scratch.mit.edu/). Scratch permite compartir proyectos que ejecutan código de programación en un lenguaje muy sencillo que se escribe a base de conectar piezas de puzzle. Aquí puedes encontrar el código ejecutable de este proyecto en scratch: https://scratch.mit.edu/projects/920589853/ (en Scratch solo me he asegurado de que podía escribir las funciones, no de ponerlo bonito). Otra opción es directamente usar papel y boli para dibujar las gráficas resolviendolas para t, t+5, t+10, t+15 y t+20.

Nuestra pregunta: ¿Como podemos modelar los cambios naturales en el número de individuos de una especie a lo largo del tiempo?

- "los cambios en el número de individuos" = dinámica poblacional
- Para simplicar usaremos una espcie que vive un solo año, y se reproduce una sola vez al año.

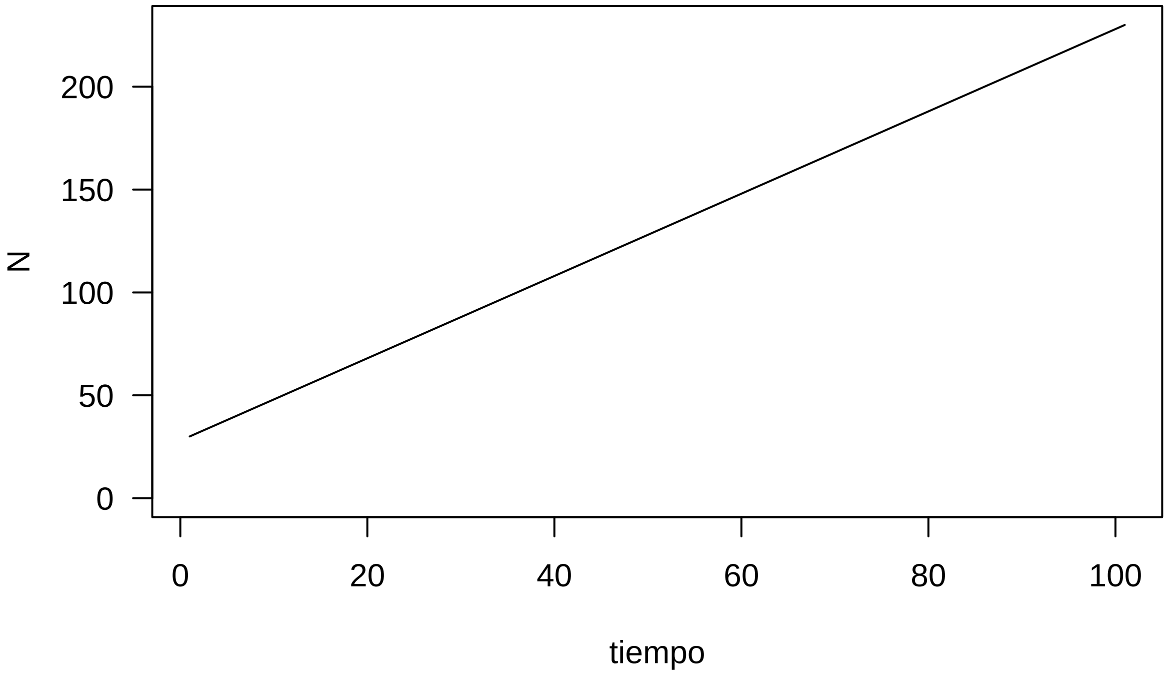
Vamos a empezar simulando las dinamicas de una espcie animal.

Conceptos ecológicos: Los animales viven en poblaciones. Estas no son estáticas, sino que pueden crecer o disminuir con el tiempo. ¿Pueden crecer linearmenete?
Conceptos matematicos: Una linea es `y = a + bx` donde `b` es la pendiente de esta recta. `y` es el número de animales que hay (el tamaño de la población) y `x` es el tiempo transcurrido. (`a` es el tamaño de la población inicial).
Conceptos de programación: Crear funciones y dibujarlas.
Reflexión: Tiene sentido que las poblaciones crezcan indefinidamente de forma lineal? (No)

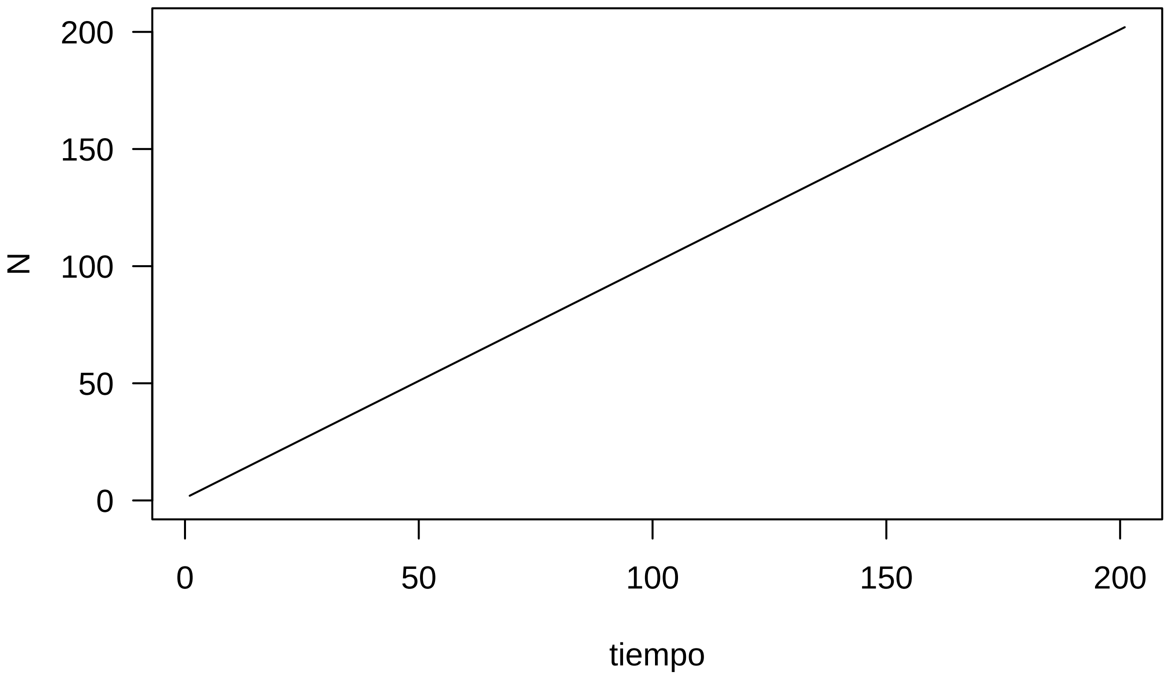
Código:

```
#creamos una función con los parametros que queremos basados en y = a + bx
# y = N (tamaño de la población en cada tiempo); lo que queremos claclular.
# a = N0 (tamaño de la población a tiempo 0)
```

```r
# b = Beta (pendiente de la recta)
# x = tiempo (cuantos años queremos calcular)
crecimiento <- function(N0 = 30, Beta = 2, tiempo = 100){
  x <- seq(0,tiempo) #creamos un vector que vaya de 0 al tiempo especificado
  N <- N0 + Beta*x #calculamos la N en cada tiempo usando la equación
  plot(N, t = "l", las = 1, xlab = "tiempo", ylim = c(0, max(N))) #graficamos
}
#nota: Aquí resolvemos la equación de forma discreta para cada unidad de tiempo, pero normalmente se tr

crecimiento()
```




```r
crecimiento(N0 = 2, Beta = 1, tiempo = 200) #podemos jugar con los parametros.
```

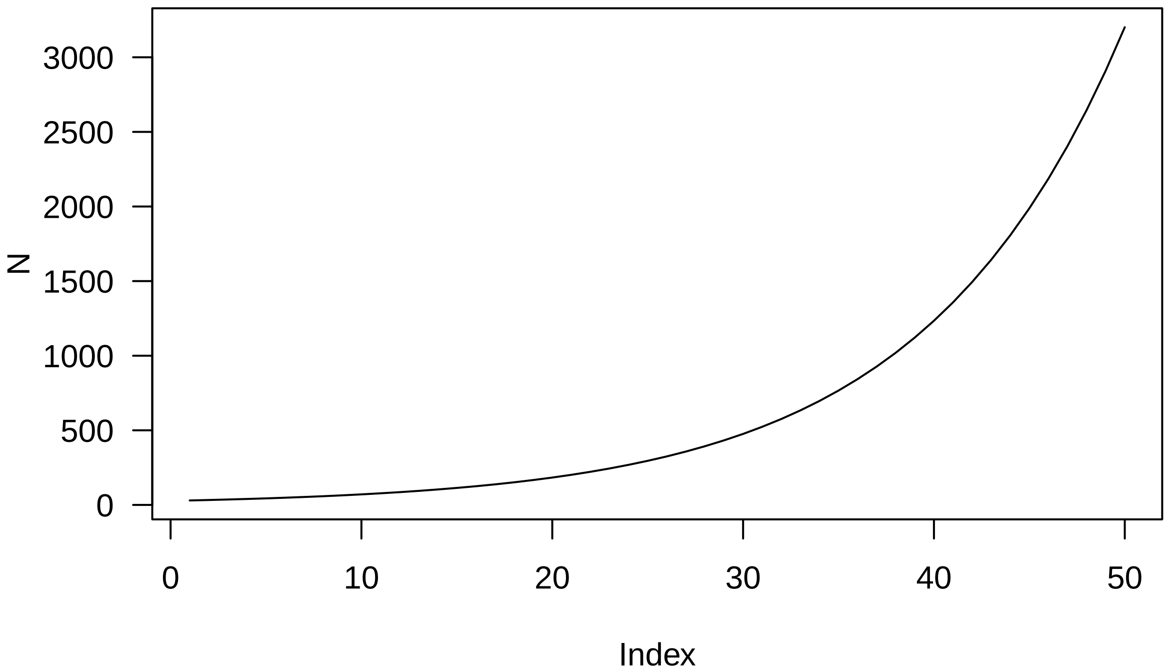


Vamos a simular las dinamicas de una espcie más realisticamente

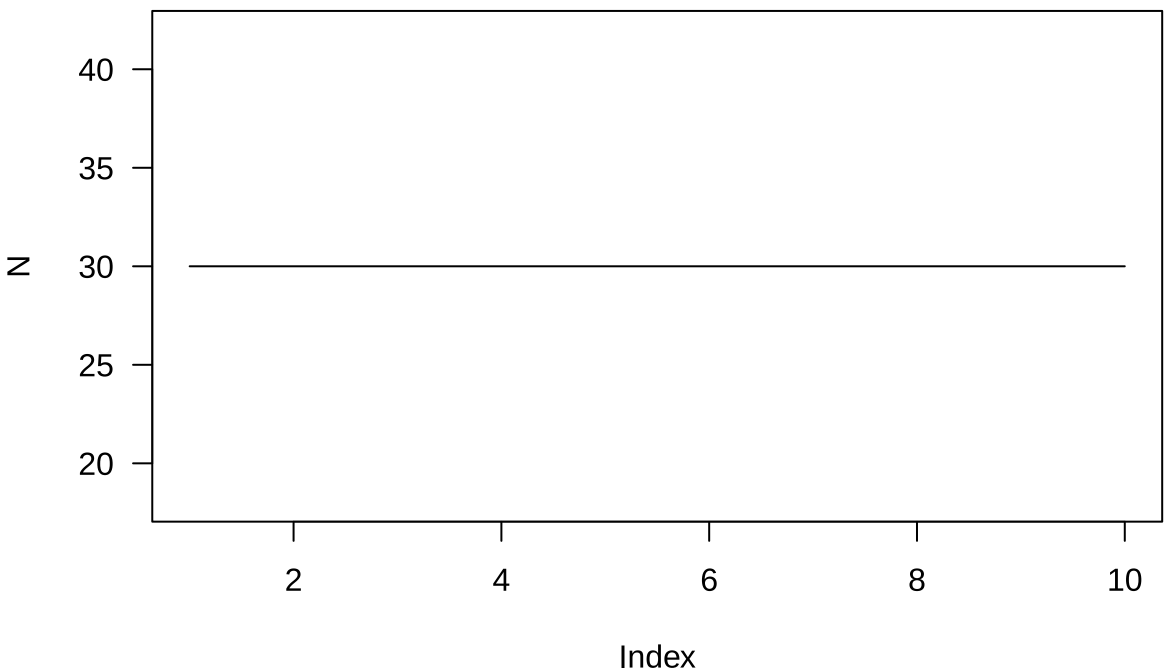
Conceptos ecológicos: El crecimiento no es constante, sino que depende de la población que hay. Es decir, si cada pareja de animales tiene 4 hijos, la población se multiplicara por dos en cada generación. **Conceptos matematicos:** Crecimiento exponencial. N = N0 + r*Nt-1. Aquí introducimos una tasa de crecimiento `r` que multiplica a la población anterior (Nt-1). El tiempo por tanto ya no està explicito en la equación, pero lo modelamos al estar afectando a N. Notad que r puede ser concebido como el % de cambio, por tanto, $r > 0$ implica que hay más nacimientos que muertes, y $r < 0$ que hay más muertes que nacimientos. **Conceptos de programación:** "for" loops (bucles) **Reflexión:** Tiene sentido que las poblaciones crezcan indefinidamente? (No, algo las ha de regular)

Código:

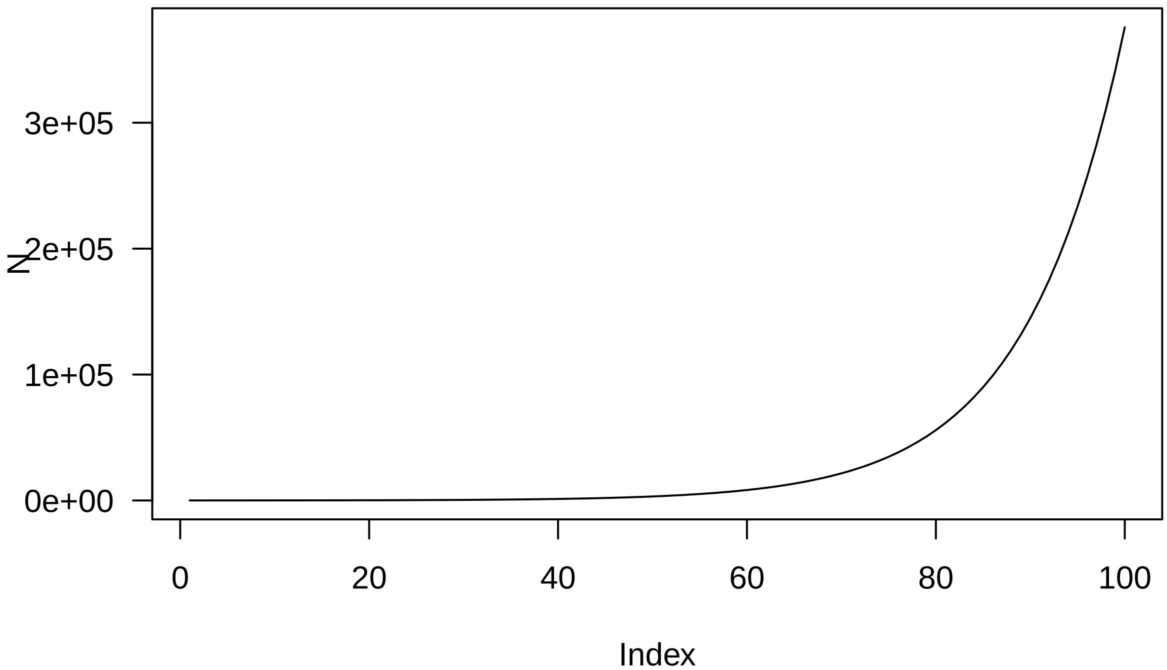
```
#creamos nuestra función:
crecimiento <- function(N0 = 30, r = 0.1, tiempo = 50){
  #la función tiene tres parametros, N0, r y el tiempo a simular.
  N <- rep(NA,tiempo) #creamos un vector vacio (NA = Not Available) para guardar el valor de N para cad
  N[1] <- N0 # asignamos N0 al primer valor (el uno que esta dentro del corchete lo indica)
  for(i in 2:tiempo){ #para asignar el resto de valores, usaremos un bucle
    N[i] <- N[i-1] + r*N[i-1]  #resolvemos la equación para cada tiempo.
  }
  plot(N, t = "l", las = 1) #graficamos.
}
#nota: normalmente se resuelve la derivada de N sobre t (en continuo), pero es mucho más claro de forma

crecimiento() #a partir del año 40 tenemos >1000 individuos. Recuerda que hemos fijado r a 0.1, es deci
```



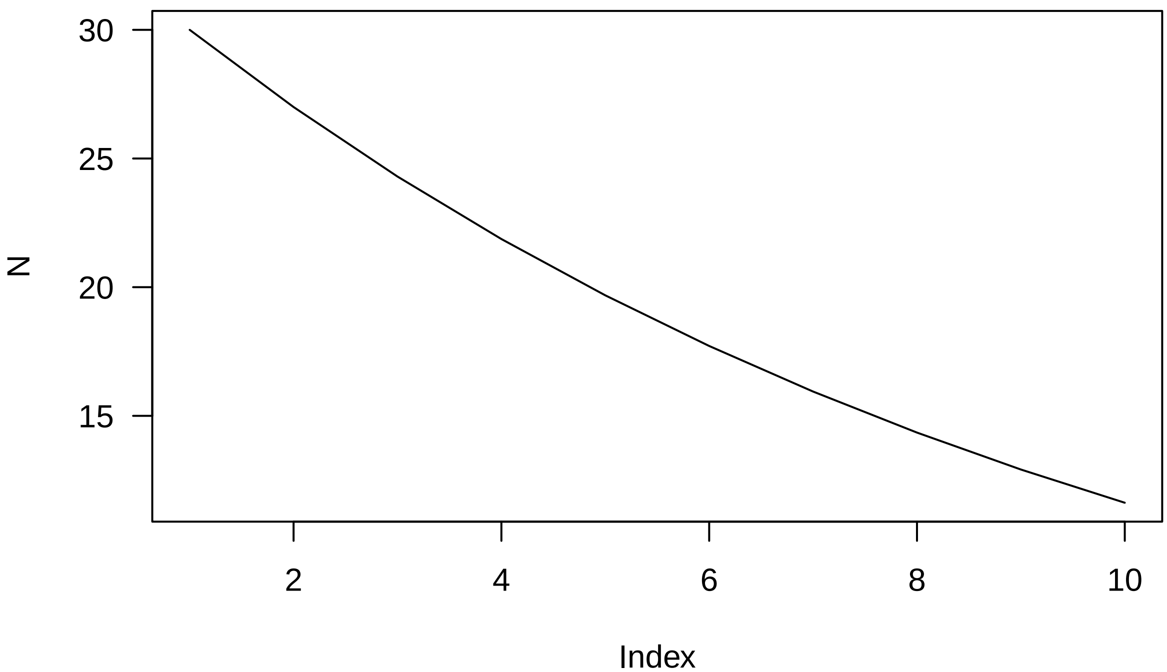

```
crecimiento(N0 = 30, r = 0, tiempo = 10) #población estable
```




```
crecimiento(N0 = 30, r = 0.1, tiempo = 100) #En 100 años pasamos de 30 a > 300000 individuos.
```




```
crecimiento(N0 = 30, r = -0.1, tiempo = 10) #En 10 años pasamos de 30 a < 10 individuos.
```

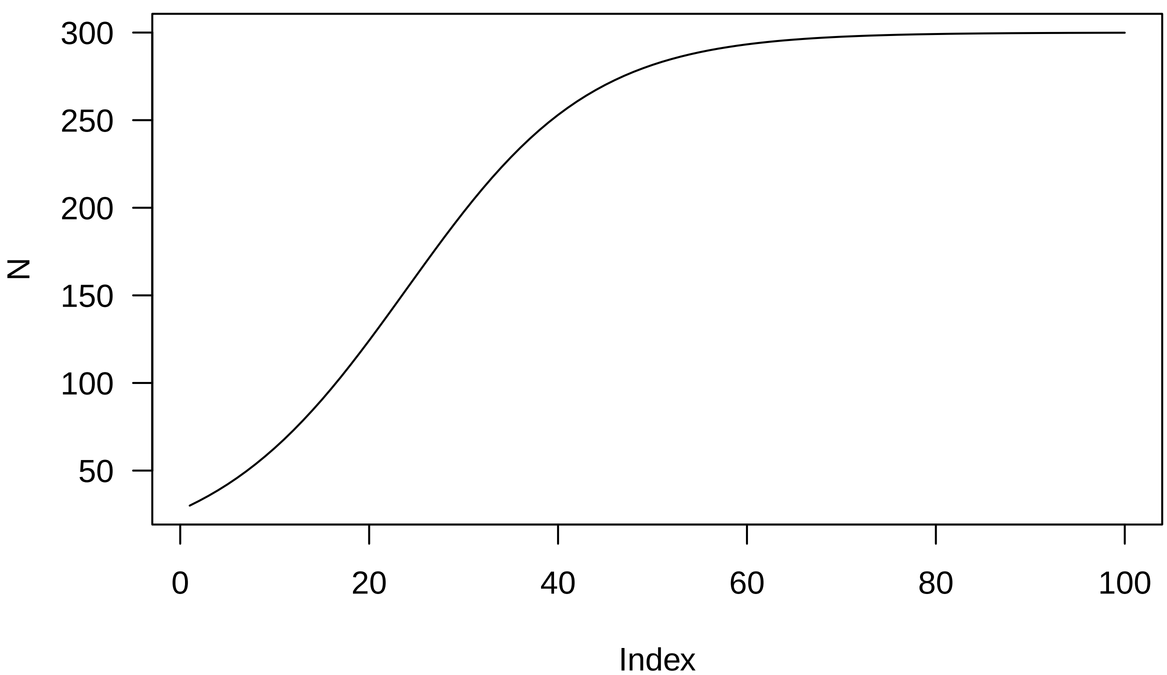


Vamos a simular las dinamicas de una espcie aun más realisticamente

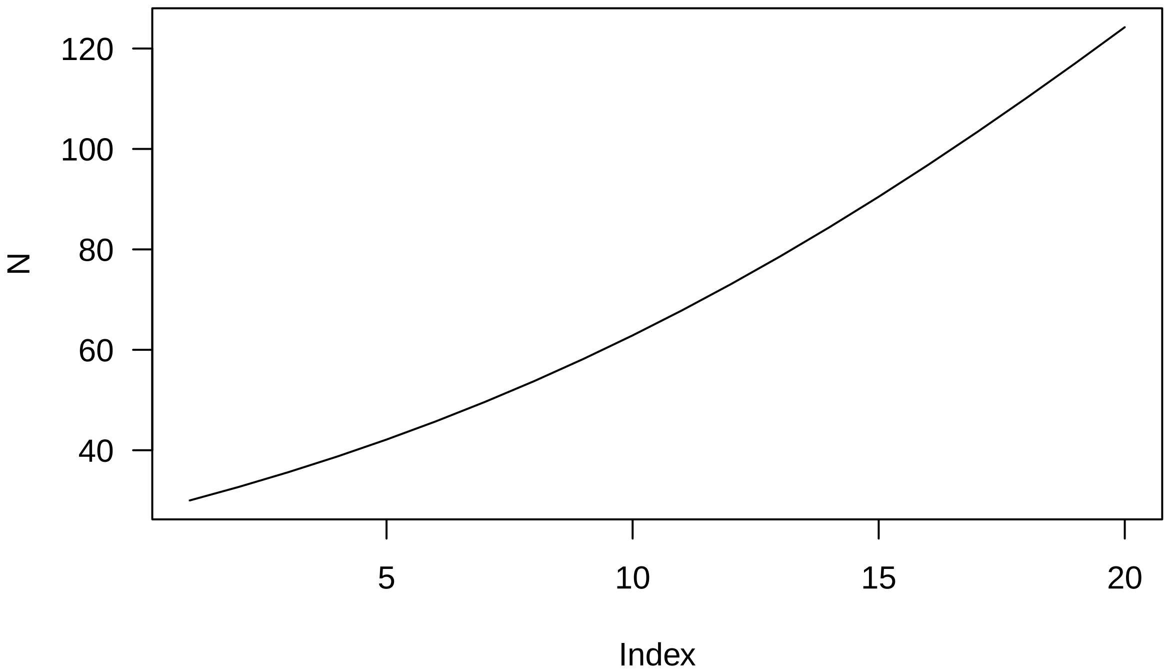
Conceptos ecológicos: Algo ha de regular las poblaciones. Hay una capacidad de carga máxima (un número de individuos máximo que puede albergar ese habitat), y esta refleja la competencia intraspecifica. Es decir, que cuando hay muchos individuos, compiten entre ellos, auto-limitandose. **Conceptos matemáticos:** Añadimos una penalización, lo que da curvas más complejas y realistas. Necesitamos penalizar el crecimiento exponencial cuando la población es muy alta. Para eso podemos multiplicar por números cercanos a 1 cuando la poblacion es baja (crecera exponencialmente) y por numeros cercanos a 0 cuando es muy alta (dejara de crecer). Eso lo conseguimos con el termino `(1 - N[i-1]/k)` Donde `k` es la capacidad de carga, o el numero máximo de individuos que pueden coexistir en esa habitat. **Conceptos de programación:** Repasar funciones y bucles. **Reflexión:**¿las poblaciones crecen hasta alcanzar un número máximo y después se estabilizan? (A veces, pero es raro que una población sea estable, y no solemos ver equilibrios estaticos en ecología).

Código:

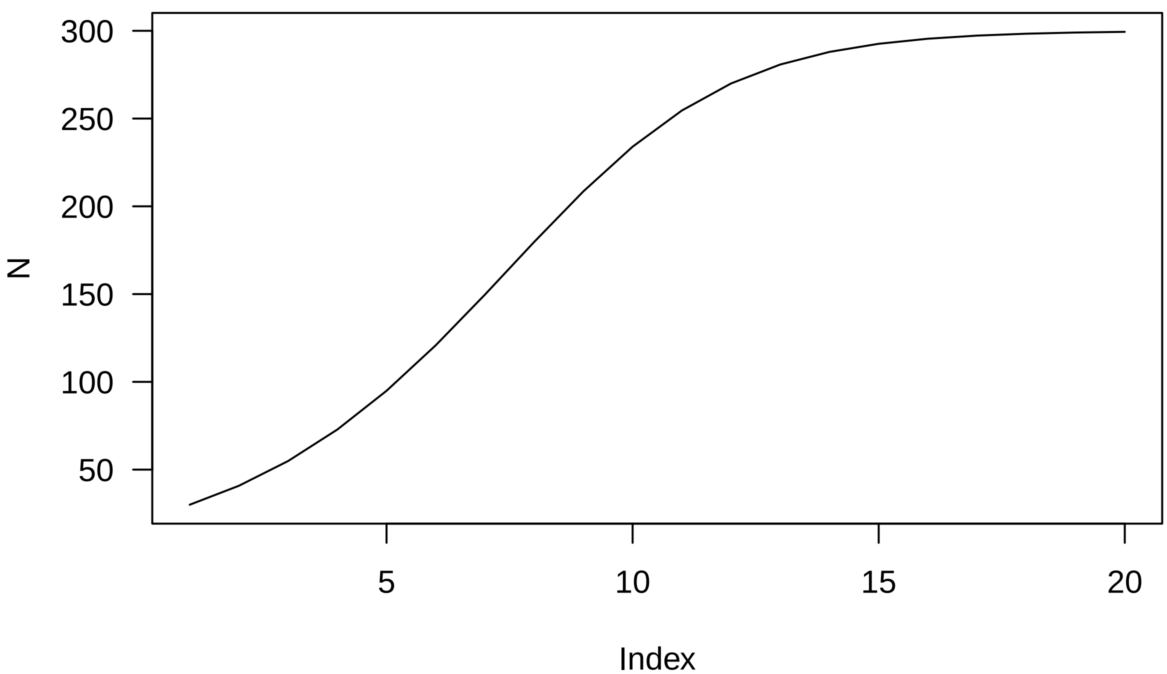
```
crecimiento <- function(N1 = 30, r = 0.1, k = 300, tiempo = 100){
  #añadimos el parametro k (individuos máximos)
  N <- rep(NA,tiempo) #creamos vector vacio
  N[1] <- N1 #le damos la población inicial
  for(i in 2:tiempo){ #bucle para calcular N en cada año i
    N[i] <- N[i-1] + r*N[i-1] * (1 - N[i-1]/k) #nuestra equación
  }
  plot(N, t = "l", las = 1) #graficar
}

crecimiento() #vemos que en unos 80 años, se estabiliza la población en 300 individuos
```



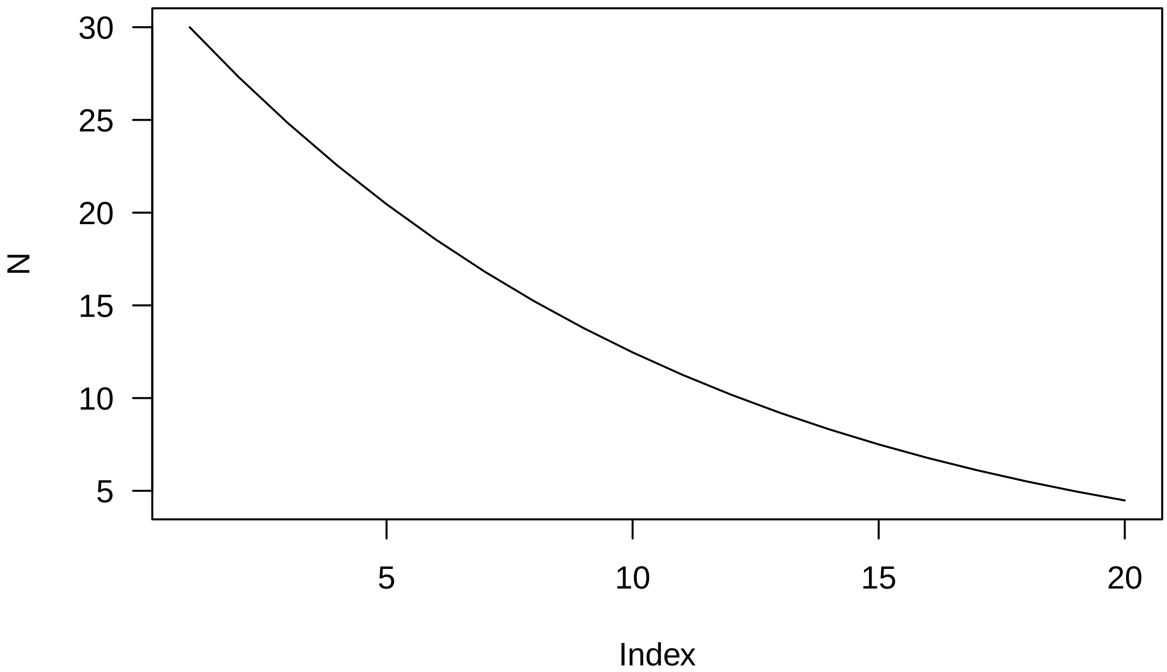

```
crecimiento(N1 = 30, r = 0.1, k = 300, tiempo = 20) #En los primeros años, el crecimiento es casí expon
```



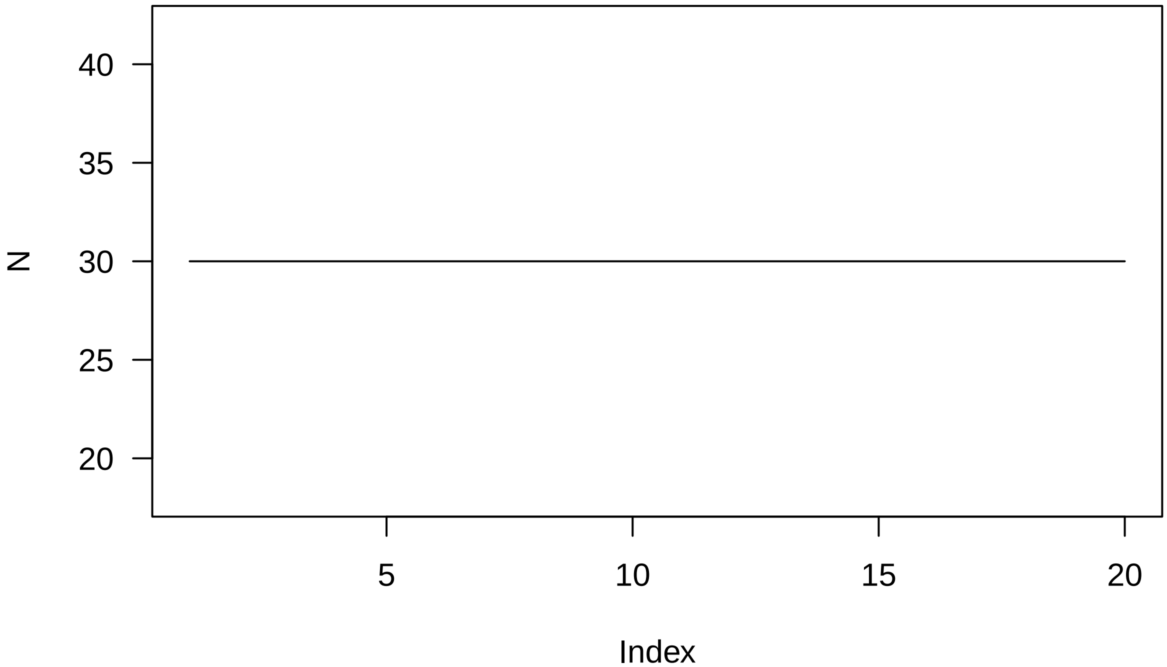

```
crecimiento(N1 = 30, r = 0.4, k = 300, tiempo = 20) #Si la población crece muy rápido, se satura antes
```




```
crecimiento(N1 = 30, r = -0.1, k = 300, tiempo = 20) #Resultado esperado
```



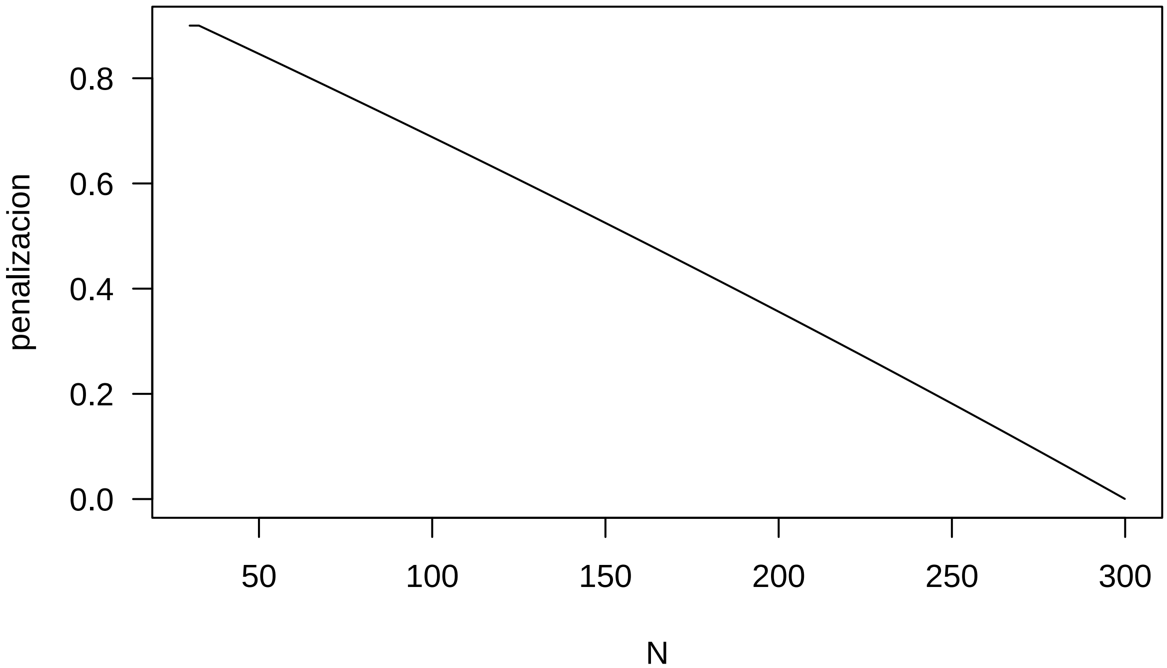

```
crecimiento(N1 = 30, r = 0, k = 300, tiempo = 20) #Resultado esperado
```



¿que hemos hecho con el segundo termino?

```
#vamos a guardar algunos objetos de la función anterior para explorarlos después
N <- rep(NA,100) #creamos vector vacio para guardar Ns en 20 años
N[1] <- 30 #N inicial
r = 0.1
k = 300 #capacidad de carga
for(i in 2:100){ #bucle para calcular N en cada año i (de 2 a 20)
    N[i] <- N[i-1] + r*N[i-1] * (1 - N[i-1]/k) #nuestra equación
  }
```

```r
penalizacion <- rep(NA) #vector vacio para guardar las penalizaciones
penalizacion[1] <- (1 - N[1]/k) #penalización para N1
for(i in 2:100){ #bucle para N2 a 20
  penalizacion[i] <- (1 - N[i-1]/k)
}
plot(penalizacion ~ N, t = "l", las = 1)
```



Este termino es la competencia intraspecifica! Cuanto mayor es la población, menor es el valor hasta alcanzar 0 y decrece de forma casi lineal.

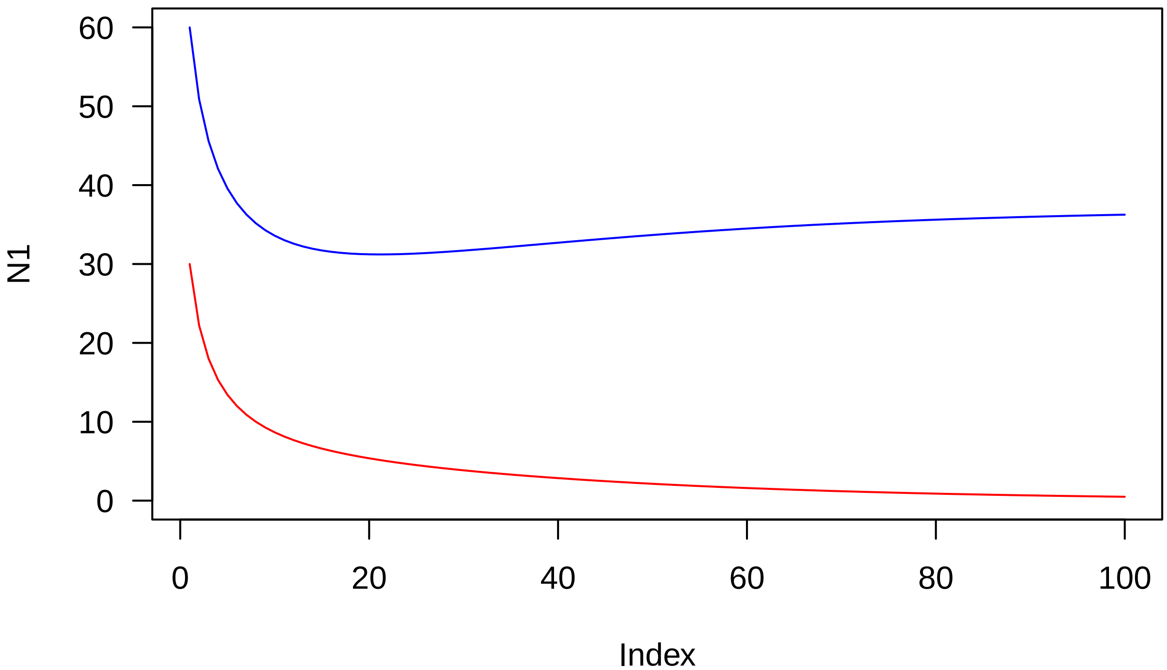
Vamos a simular las dinamicas de dos espcies

Conceptos ecológicos: Las especies no viven solas. También compiten con otras especies. La competencia también limita el crecimiento, regulando las poblaciones. **Conceptos matematicos:** Podemos usar un sistema de equaciones que dependen una de otra. A los coeficientes de competencia les llamamos alpha (`a`) y pueden ser intraspecificos (`a11` y `a22`; que equivalen a la autolimitación que hemos visto antes) o intraspecificos (`a12` y `a21`; que equivalen a la competencia entre especies). Estos coeficientes van multiplicados por los tamaños poblacionales, por tanto, cuanto más grande la población, más fuerte la competencia. Aquí, los parametros iniciales son muy imprtantes para que las poblaciones sean viables. **Conceptos de programación:** Repasar funciones, bucles y graficarlas. **Reflexión:** A partir de aquí se puede seguir complicando los modelos, por ejemplo añadiendo más especies, diferentes tipos de interacciones (mutualistas, antagonistas), variables abioticas (e.g. lluvia) que afectan la r segun el año, estocasticidad, etc…

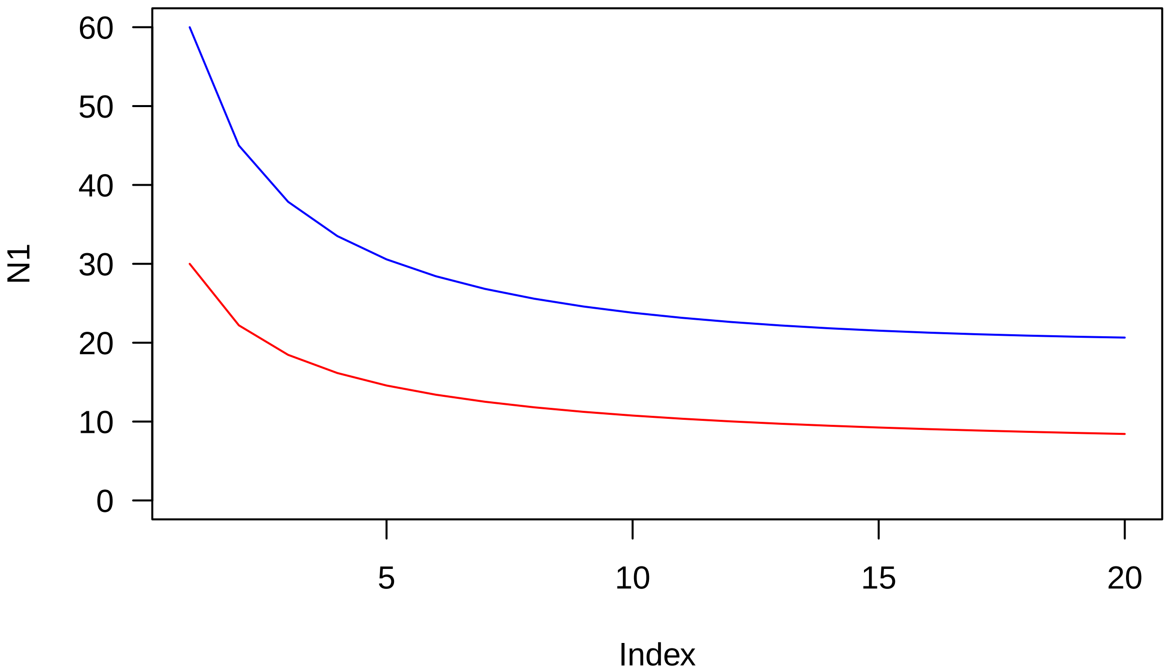
Código:

```r
crecimiento <- function(N01 = 30, r1 = 0.1, a11 = -0.005, a12 = -0.0035,
                        N02 = 60, r2 = 0.1, a21 = -0.0027, a22 = -0.003,
                        tiempo = 100){
  N1 <- rep(NA,tiempo)
  N1[1] <- N01
  N2 <- rep(NA,tiempo)
  N2[1] <- N02
  for(i in 2:tiempo){
```

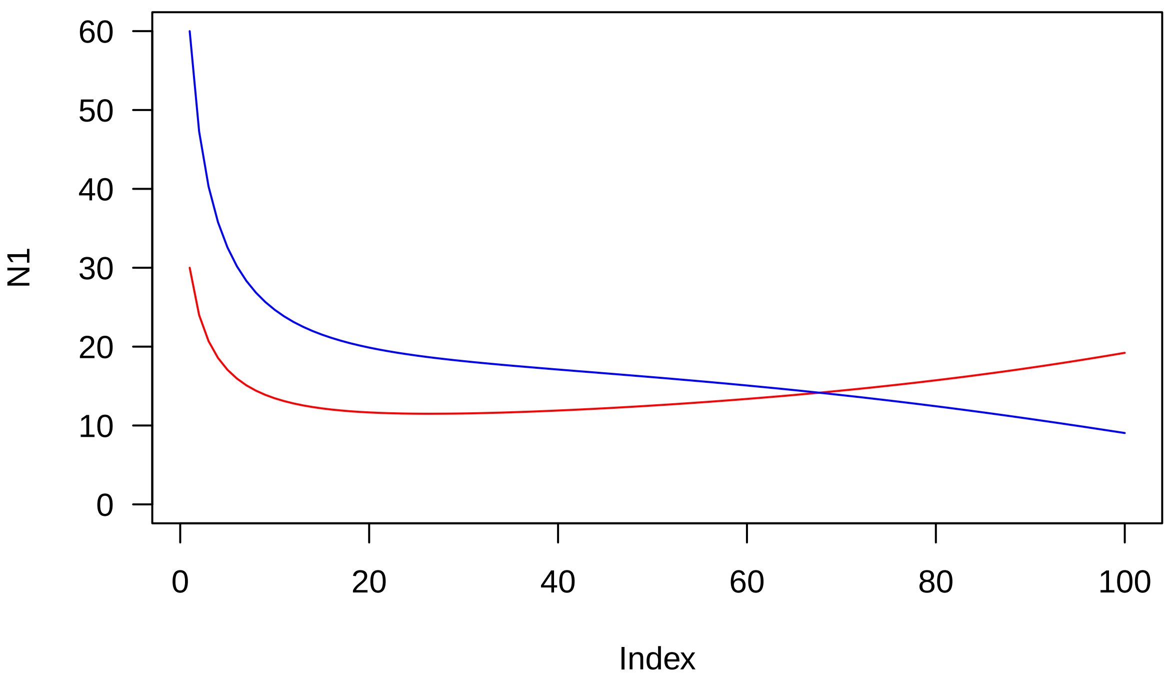
```
    N1[i] <- N1[i-1] + N1[i-1] * (r1 + a11*N1[i-1] + a12*N2[i-1])
    N2[i] <- N2[i-1] + N2[i-1] * (r2 + a21*N2[i-1] + a22*N1[i-1])
  }
  plot(N1, t = "l", col = "red", las = 1, ylim = c(0, max(c(max(N2), max(N1)))))
  points(N2, t = "l", col = "blue")
}

crecimiento() #gana azul y rojo se acaba extinguiendo
```



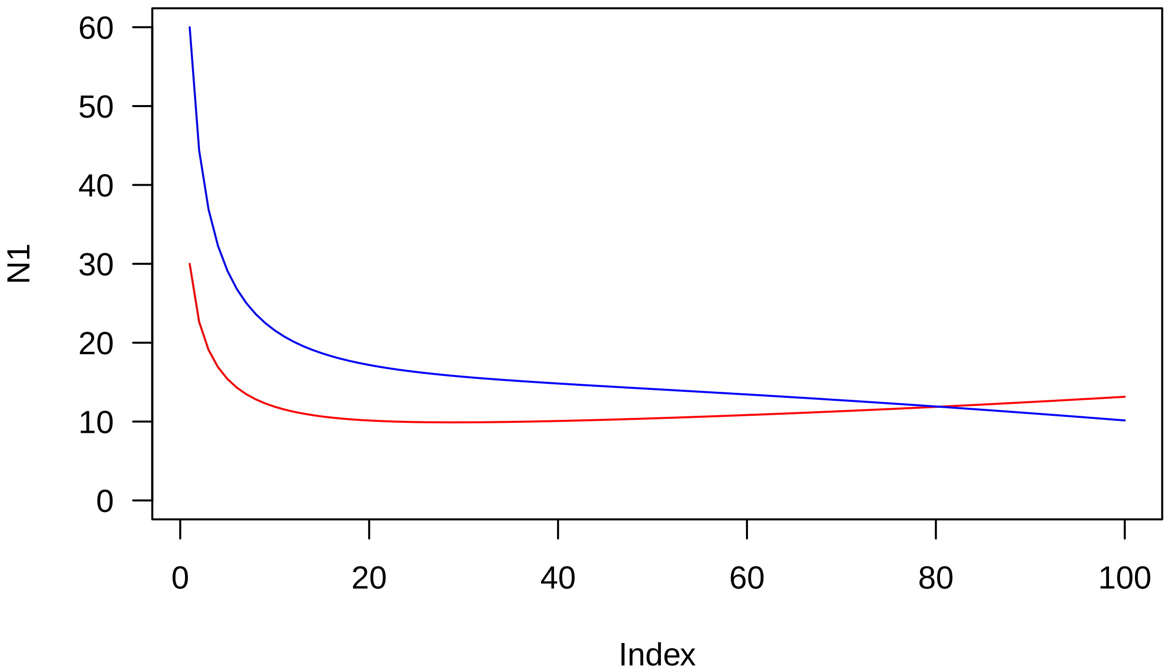

```
crecimiento(N01 = 30, r1 = 0.1, a11 = -0.005, a12 = -0.0035,
            N02 = 60, r2 = 0.11, a21 = -0.0035, a22 = -0.005,
            tiempo = 20) #ambas coexisten, pero a poblaciones más bajas, al menos a corto plazo
```




```
crecimiento(N01 = 30, r1 = 0.1, a11 = -0.003, a12 = -0.0035,
            N02 = 60, r2 = 0.1, a21 = -0.0027, a22 = -0.005,
            tiempo = 100) #roja acaba ganando, aunque empieze en desventaja
```




```
crecimiento(N01 = 30, r1 = 0.1, a11 = -0.0045, a12 = -0.0035,
            N02 = 60, r2 = 0.1, a21 = -0.0030, a22 = -0.006,
            tiempo = 100)
```



Notad que para que no se extingan, la competencia dentro de especies (auto-limitación) ha de ser mayor que la competencia entre especies. Esto es una regla general en Ecología.

En la naturaleza hay muchos tipos de interacciones (competencia, depredación, mutualismos) y muchas espcies interaccionando a la vez. Estos modelos pueden llegar a tener mucha complejidad, y nos ayudan a saber las reglas que permiten a las especies coexistir. Los ecolólogos usamos estas herramientas para entender como se regulan las poblaciones y comparamos modelos con observaciones reales.

Esta familia de modelos fueron propuestos por primera vez por Vito Voletrra y Alfred Lotka en los años 1920, y se han ido refinando a lo largo del tiempo, pero se siguen conociendo como equaciones Lotka-Volterra.