Crecimiento Logístico de una población

Ramiro Bustamante - Jorge Cortés

08-04-2022

Crecimiento Denso-dependiente

Cuando los recursos son limitantes, el crecimiento exponencial no puede sostenerse indefinidamente debido a que al aumentar el tamaño de la población sus tasa de crecimiento poblacional tienden a disminuir. El crecimiento poblacional resultante será un crecimiento sostenido a bajas densidades poblacionales y una reducción del crecimiento a densidades poblacionales mayores. Este es el llamado crecimiento logístico que se puede representar por un modelo matemático adecuado.

Ya sabemos que el modelo de crecimiento exponencial es como el que sigue:

$$\mathrm{N}_t = \mathrm{N}_0 * \lambda$$

Podemos descomponerlo como:

$$\mathrm{N}_{t+1} = \mathrm{N}_t * (1 + r) = \mathrm{N}_t + r * \mathrm{N}_t$$

Ahora incorporamos un factor que indique el efecto negativo de la densidad poblacional sobre el crecimiento poblacional, podemos escribir la siguiente ecuación:

$$\mathrm{N}_{t+1} = \mathrm{N}_t + r(1 - a\mathrm{N}_t)$$

Buscamos a que tamaño de población el crecimiento será cero al resolver:

$$0 = r(1 - a\mathrm{N}_t)$$

Obteniendo que $\mathrm{N}_t = \frac{1}{a}$, lo podemos renombrar como un valor K en el equilibrio llamado capacidad de carga. Reescribiendo la ecuación tenemos:

$$\mathrm{N}_{t+1} = \mathrm{N}_t + r(1 - \frac{\mathrm{N}_t}{K}) * \mathrm{N}_t$$

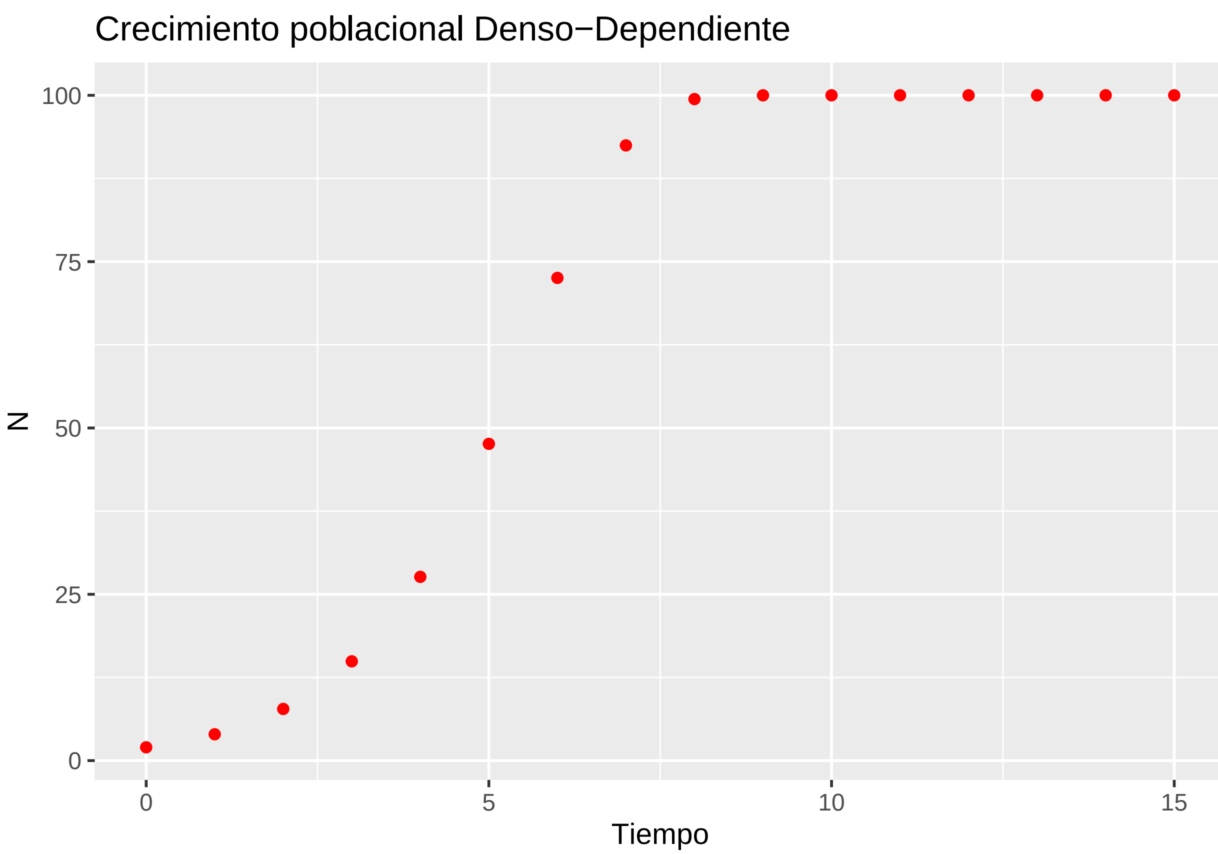
Así, podemos generar una función simple (en tiempo discreto) para proyectar el tamaño poblacional futuro incorporando los parámetros del modelo logístico.

```r
##install.packages("ggplot2")
##install.packages("ggpubr")
##install.packages("zoo")
##instal.packages("lattice")
library(ggplot2)
library(ggpubr)
library(zoo)
library(lattice)

dlogistic <- function(K = 100, r = 1, N0 = 2, t = 15) {
    N <- c(N0, numeric(t))
```

```
    for (i in 1:t) N[i + 1] <- {
        N[i] + r * N[i] * (1 -  N[i]/K)
    }
        return(N)
}
```

Podemos reemplazar los parámetros con valores predeterminados para generar una proyección en el tiempo del tamaño poblacional y graficarlos:

```
Nts <- dlogistic()

t <- 15; K <- 100
crec_logistic <- qplot(0:t, Nts,
                       xlab = "Tiempo",
                       ylab = "N",
                       main = "Crecimiento poblacional Denso-Dependiente",
                       colour=I("Red"))
crec_logistic
```



Incremento poblacional percápita vs Tamaño poblacional

Exploremos como cambia el incremento total de la población en cada tiempo y el incremento percápita.

El incremento de la población entre dos tiempos, se puede representar por la siguiente ecuación:

$$\frac{\Delta N}{\Delta t} = \frac{\mathrm{N}_{t+1} - \mathrm{N}_t}{\Delta t}$$

Dado que $\Delta t = 1$, el incremento se puede simplificar como:

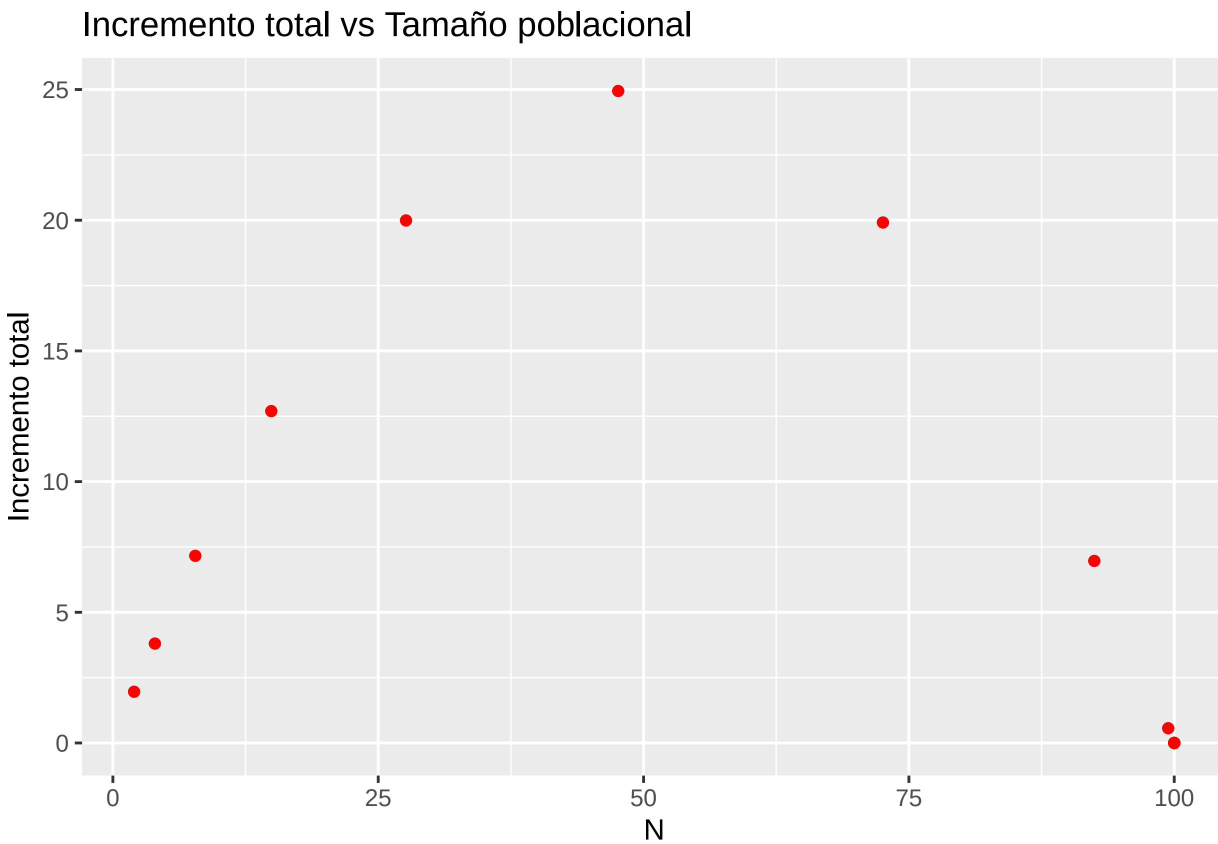
$$\mathrm{N}_{t+1} - \mathrm{N}_t$$

El incremento per capita es:

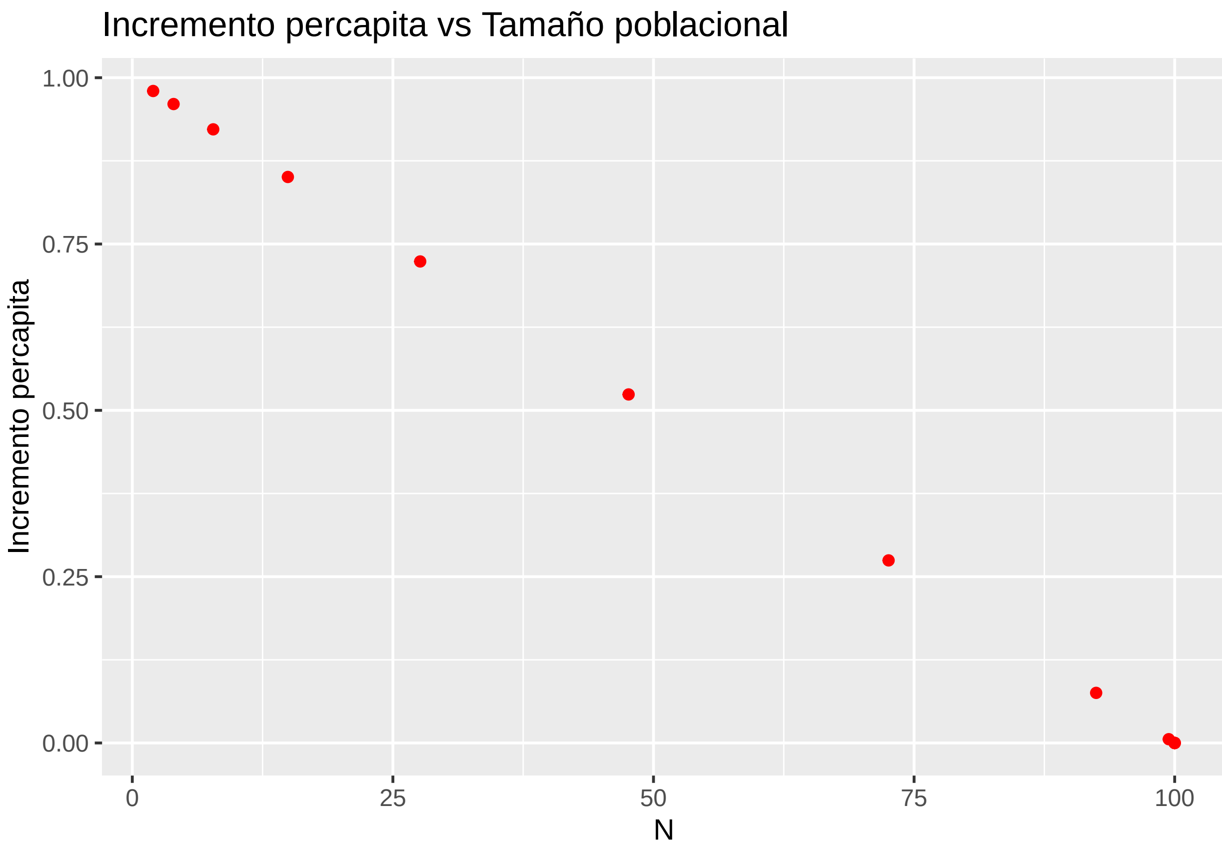
$$\frac{\mathrm{N}_{t+1} - \mathrm{N}_t}{\mathrm{N}_t}$$

Estos nuevos parámetros se representan en el siguiente Script:

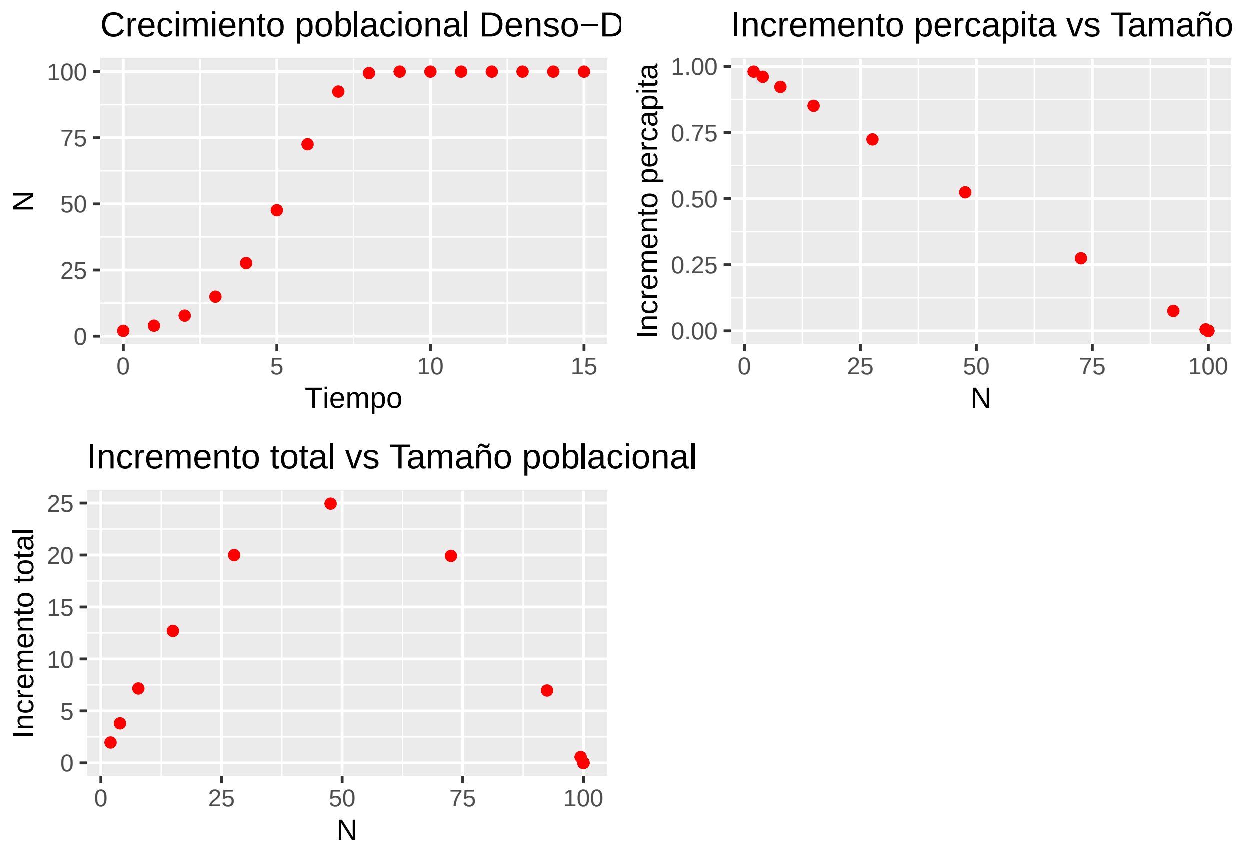
```
total.incr <- Nts[1:t + 1] - Nts[1:t]
per.capita.incr <- total.incr/Nts[1:t]


inc_total <- qplot(Nts[1:t], total.incr,
      xlab = "N",
      ylab = "Incremento total",
      main = "Incremento total vs Tamaño poblacional",
      colour=I("Red"))
inc_percapita <- qplot(Nts[1:t], per.capita.incr,
      xlab = "N",
      ylab = "Incremento percapita",
      main = "Incremento percapita vs Tamaño poblacional",
      colour=I("Red"))

inc_total
```




```
inc_percapita
```



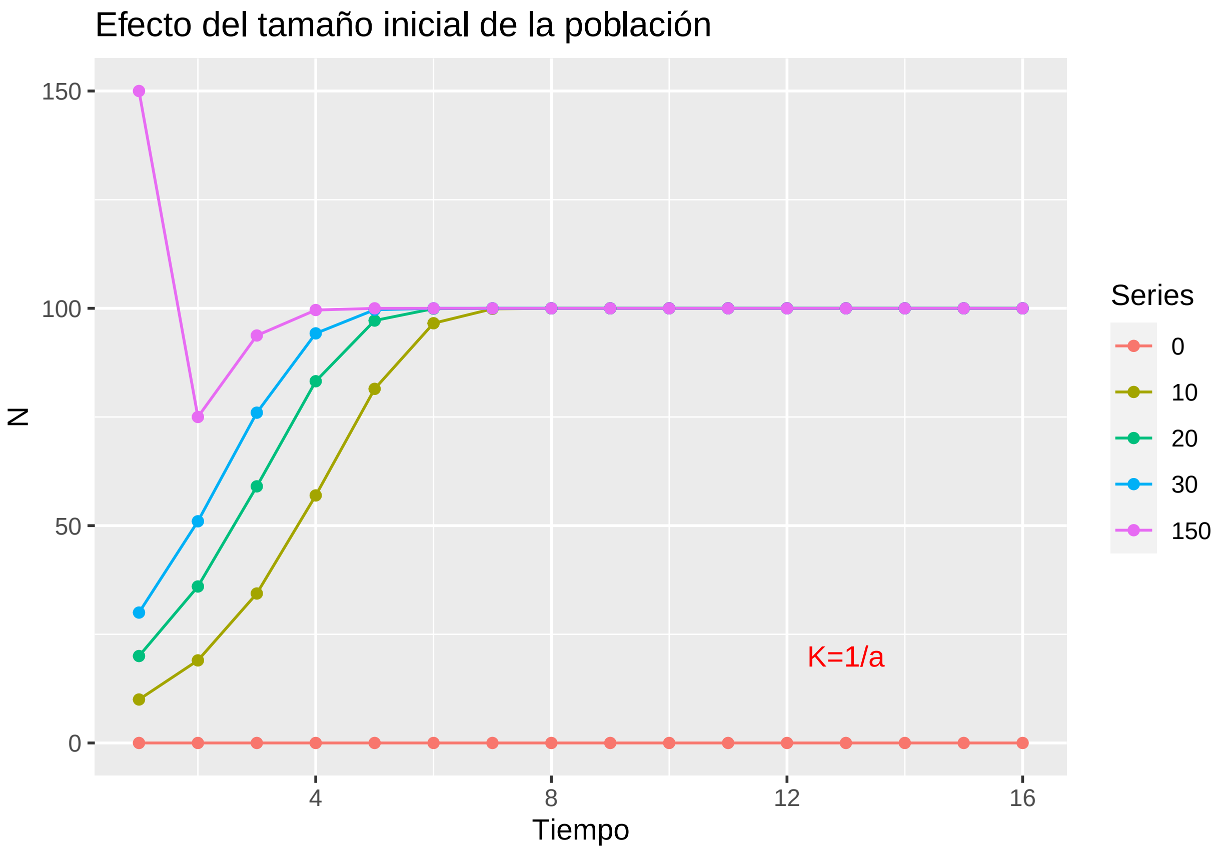

```
ggarrange(crec_logistic, inc_percapita, inc_total)
```



Efecto del tamaño inicial de la población

Tomaremos 5 valores de población inicial y aplicaremos nuestro modelo logístico a todos los valores para N_0.

```
N0s <- c(0, 10, 20, 30, 150)
N <- data.frame(sapply(N0s, function(n) dlogistic(N0 = n)))

names(N)[names(N) == "X1"] <- "0"
names(N)[names(N) == "X2"] <- "10"
names(N)[names(N) == "X3"] <- "20"
names(N)[names(N) == "X4"] <- "30"
names(N)[names(N) == "X5"] <- "150"

autoplot(zoo(N), facet = NULL) +
  geom_point() +
  annotate(geom="text", x=13, y=20, label= "K=1/a", color = "red")+
  ggtitle(label = "Efecto del tamaño inicial de la población")+
  xlab(label = "Tiempo")+
  ylab(label = "N")
```

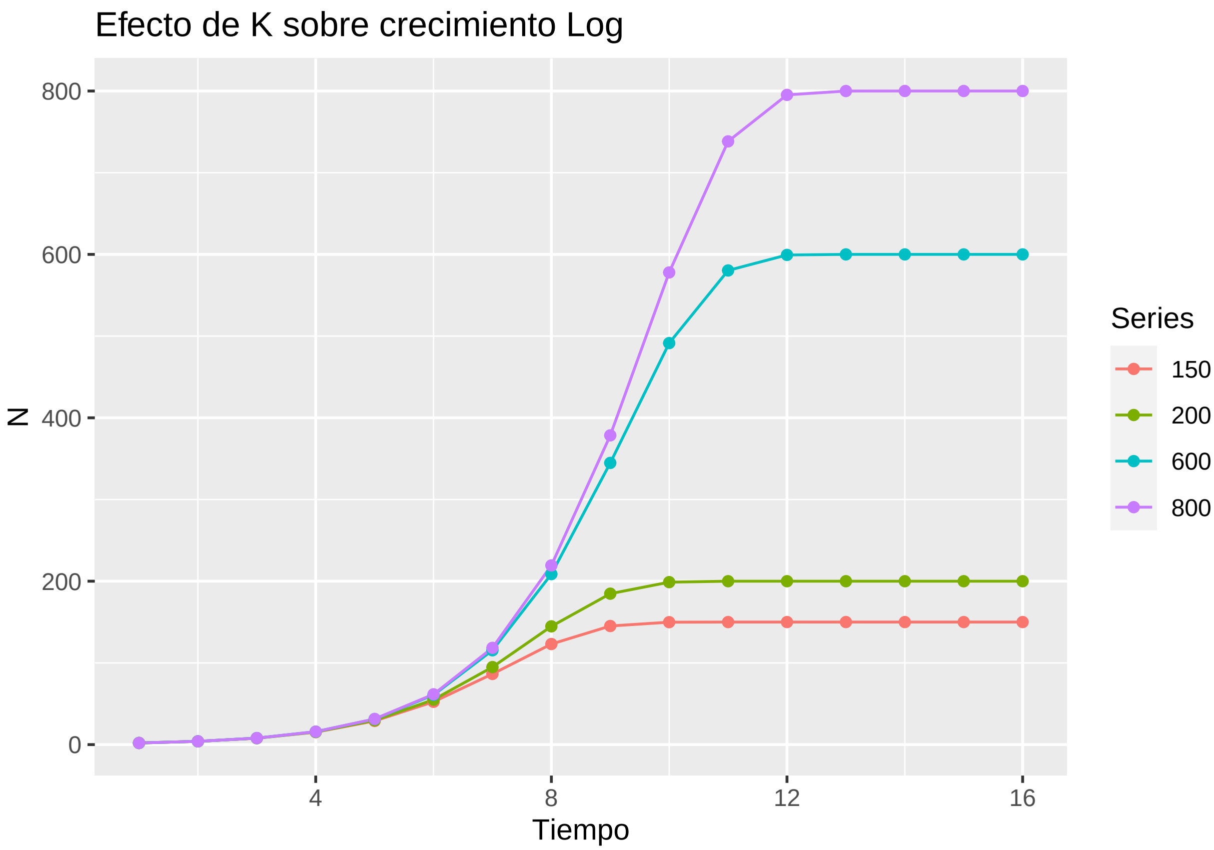


¿Qué efecto puedes notar al cambiar el tamaño inicial de la población?

Efecto del valor de K

Tomaremos 4 valores de K y le aplicaremos nuestro modelo logístico.

```
K.s <- c(150, 200, 600, 800)
N <- data.frame(sapply(K.s, function(a) dlogistic(K = a, t = 15)))

names(N)[names(N) == "X1"] <- "150"
names(N)[names(N) == "X2"] <- "200"
names(N)[names(N) == "X3"] <- "600"
names(N)[names(N) == "X4"] <- "800"

autoplot(zoo(N), facet = NULL) + geom_point() +
  ggtitle(label = "Efecto de K sobre crecimiento Log")+
  xlab(label = "Tiempo")+
  ylab(label = "N")
```



¿Qué efecto puedes notar al cambiar los valores de K?

Efecto de lambda sobre el crecimiento poblacional

En esta oportunidad vamos a reemplazar λ por r, dado que:

$$\lambda = e^r$$

y que:

$$r = ln(\lambda)$$

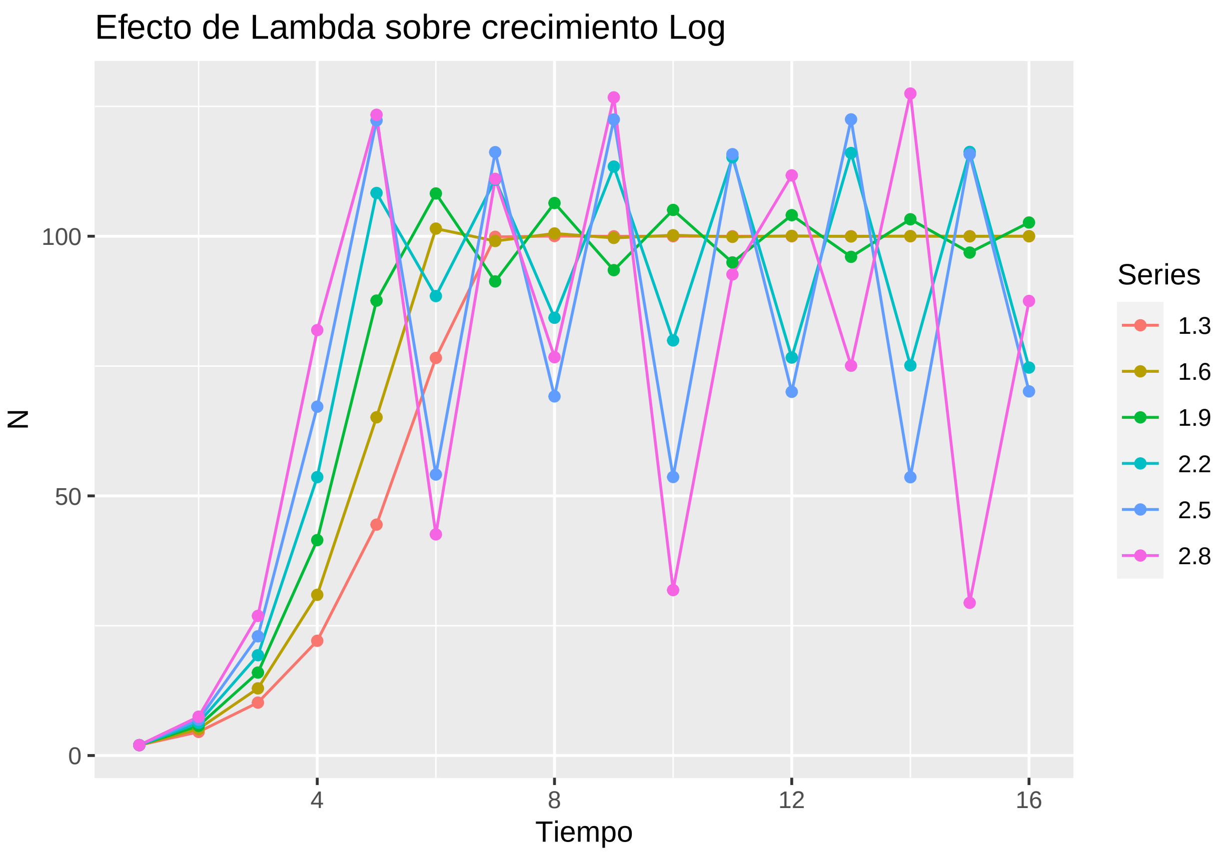
Tomaremos valores entre 1.3 y 2.8 para λ de manera secuencial:

```
lambda <- seq(1.3, 2.8, by = 0.3)
t <- 15
Ns <- data.frame(sapply(lambda, function(r) dlogistic(r = r,t = t)))

Ns
```

```
##           X1        X2        X3        X4        X5        X6
## 1    2.00000   2.00000   2.00000   2.00000   2.00000   2.00000
## 2    4.54800   5.13600   5.72400   6.31200   6.90000   7.48800
## 3   10.19150  12.93154  15.97708  19.32189  22.95975  26.88444
## 4   22.09019  30.94642  41.48346  53.61667  67.18037  81.92322
## 5   44.46375  65.13779  87.60536 108.32890 122.30124 123.38865
## 6   76.56529 101.47135 108.23626  88.47916  54.11450  42.58363
```

```
## 7   99.89100  99.08255  91.29848 110.90495 116.19127 111.04355
## 8  100.03255 100.53700 106.39275  84.29786  69.15916  76.70673
## 9   99.99022  99.67319  93.47004 113.41831 122.48232 126.73575
## 10 100.00293 100.19438 105.06680  79.93691  53.64014  31.86125
## 11  99.99912  99.88277  94.95211 115.22010 115.80887  92.64885
## 12 100.00026 100.07012 104.05896  76.63955  70.03868 111.71897
## 13  99.99992  99.95785  96.03391 116.02691 122.49996  75.06049
## 14 100.00002 100.02526 103.27061  75.11675  53.59385 127.47570
## 15  99.99999  99.98483  96.85321 116.23802 115.77095  29.40614
## 16 100.00000 100.00910 102.64397  74.71355  70.12549  87.53113
```

```r
names(Ns)[names(Ns) == "X1"] <- "1.3"
names(Ns)[names(Ns) == "X2"] <- "1.6"
names(Ns)[names(Ns) == "X3"] <- "1.9"
names(Ns)[names(Ns) == "X4"] <- "2.2"
names(Ns)[names(Ns) == "X5"] <- "2.5"
names(Ns)[names(Ns) == "X6"] <- "2.8"

autoplot(zoo(Ns), facet = NULL) + geom_point() +
  ggtitle(label = "Efecto de Lambda sobre crecimiento Log")+
  xlab(label = "Tiempo")+
  ylab(label = "N")
```



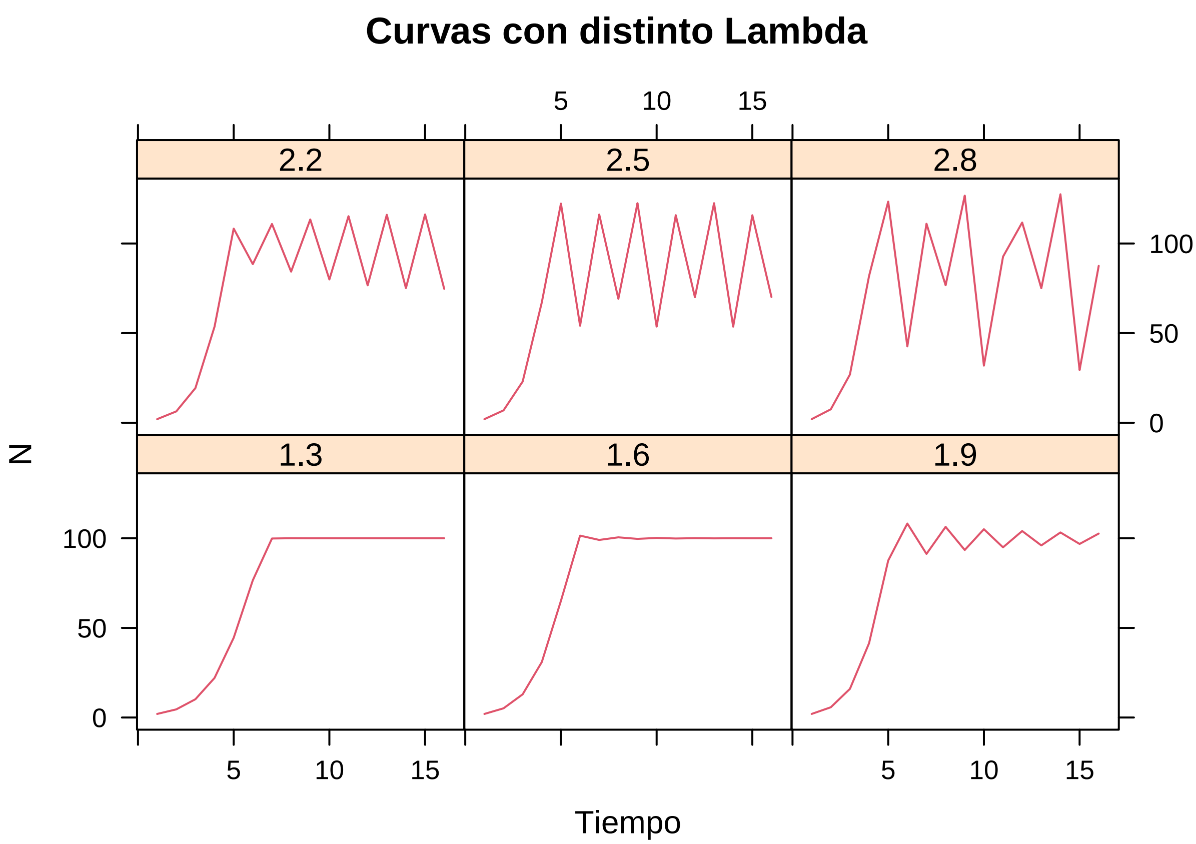
¿Qué efecto puedes notar al cambiar el valor de λ? ¿Era algo esperado?

Grafiquemos cada curva por separado para notar mejor el efecto de cambiar λ.

```
tmp <- data.frame(lambda = as.factor(lambda), t(Ns))

Ns2 <- reshape(tmp, varying = list(2:ncol(tmp)), idvar = "lambda", v.names = "N", direction = "long")
str(Ns2)
```

```
## 'data.frame':    96 obs. of  3 variables:
##  $ lambda: Factor w/ 6 levels "1.3","1.6","1.9",..: 1 2 3 4 5 6 1 2 3 4 ...
##  $ time  : int  1 1 1 1 1 1 2 2 2 2 ...
##  $ N     : num  2 2 2 2 2 ...
##  - attr(*, "reshapeLong")=List of 4
##   ..$ varying:List of 1
##   .. ..$ : chr [1:16] "X1" "X2" "X3" "X4" ...
##   ..$ v.names: chr "N"
##   ..$ idvar  : chr "lambda"
##   ..$ timevar: chr "time"
```

```
print(xyplot(N ~ time | lambda,
             data = Ns2,
             type = "l",
             layout = c(3, 2, 1),
             col = 2,
             main="Curvas con distinto Lambda",
             xlab = "Tiempo"))
```



¿Qué puedes observar? ¿Existe algún patron?

Anexo

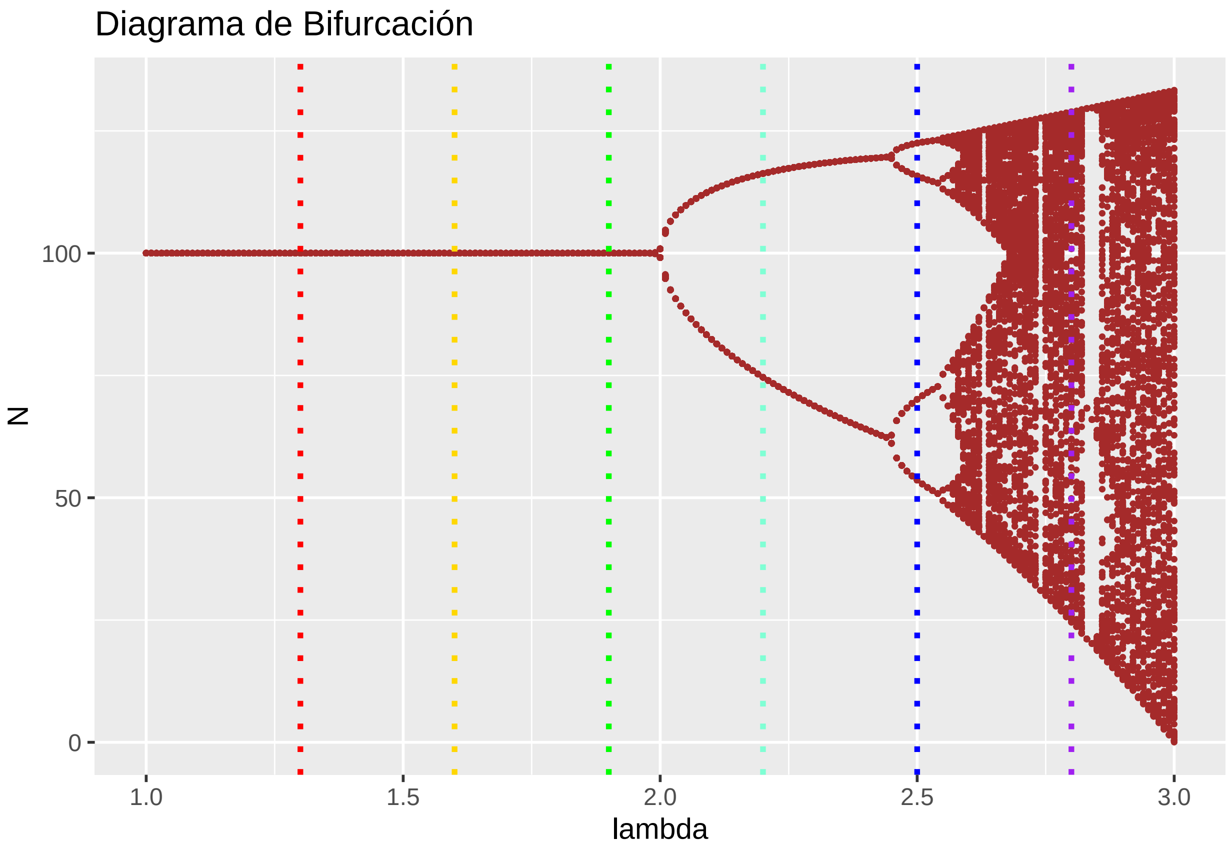
Diagrama de Bifurcación

Veremos otra forma de representar las dinámicas poblacionales al incorporar diferentes valores de λ: estos son los diagramas de bifurcación los cuales resumen la conducta demográfica de las poblaciones al modificar los parámetros demográficos, en este caso λ.

```
num.lambda <- 201; t <- 400
lambda.s <- seq(1, 3, length = num.lambda)

tmp <- sapply(lambda.s, function(r) dlogistic(r = r, N0 = 99, t = t))

tmp.s <- stack(as.data.frame(tmp))
names(tmp.s) <- c("N", "Old.Column.ID")
tmp.s$lambda <- rep(lambda.s, each = t + 1)
tmp.s$time <- rep(0:t, num.lambda)
```

Tomamos sólo los valores finales que alcanza N para cada simulación.

```
N.bif <- subset(tmp.s, time > 0.5 * t)

d_bif <- qplot(lambda, N, data=N.bif,
      main = "Diagrama de Bifurcación",
      geom="point",
      size=I(0.5),
      colour=I("brown"))
d_bif+
  geom_vline(xintercept = 1.3,
             linetype="dotted",
            color = "red",
            size=1.0) +
  geom_vline(xintercept = 1.6,
             linetype="dotted",
            color = "gold",
            size=1.0) +
  geom_vline(xintercept = 1.9,
             linetype="dotted",
             color = "green",
             size=1.0)+
  geom_vline(xintercept = 2.2,
             linetype="dotted",
             color = "aquamarine",
             size=1.0)+
  geom_vline(xintercept = 2.5,
             linetype="dotted",
             color = "blue",
             size=1.0) +
  geom_vline(xintercept = 2.8,
             linetype="dotted",
             color = "purple",
             size=1.0)
```



¿Qué puedes observar? ¿Qué significan las bifurcaciones que se observan? ¿Cómo se relaciona este Diagrama de Bifurcación con lo observado en los gráficos anteriores?