

# Extensión del modelo Mankiw-Romer-Weil

Avellanal, Bordagorri & Parrilla

## Contents

<b>Parámetros de eficiencia y ahorro</b>	<b>2</b>
<b>Simulación del modelo</b>	<b>2</b>
Verhulst (1838) . . . . .	2
Ricker (1954) . . . . .	13
<b>Especificaciones de leyes generales de población</b>	<b>23</b>
Ley de población de Verhulst (1838) . . . . .	23
Ley de población de Beverton (1957) . . . . .	23
Ley de población de Ricker (1954) . . . . .	23
Ley de población de Hassell (1975) . . . . .	23

## Parámetros de eficiencia y ahorro

Se analizará la sensibilidad del modelo a los parámetros que determinan la dinámica poblacional.

```
# Parametros de ahorro
s_k = 0.25 # tasa de ahorro del capital fisico
s_h = 0.3 # tasa de ahorro del capital humano
# Parametros de eficiencia
delta = 0.05 # tasa de depreciacion
alpha = 0.40 # parametro de eficiencia de escala del capital fisico
beta = 0.4 # parametro de eficiencia de escala del capital humano
```

## Simulacion del modelo

A continuación presentamos los resultados de las simulaciones para dos formas funcionales distintas de la dinámica poblacional. Estas ecuaciones son fenomenológicas en la medida que describen la evolución de una población de organismos y no se basan en los microfundamentos que hacen a la dinámica observada.

### Verhulst (1838)

#### Parámetros modelo poblacional

```
M = 500
```

El tamaño de la población es meramente ilustrativo. Algunas especificaciones de la logística tienen el problema de que para  $L[t]$  muy grandes,  $L[t+1]$  puede tomar valores negativos.

```
R <- seq(0.01,3.5,length=50)
```

La longitud del vector de valores puede ser arbitrariamente más grande para ver más en detalle los puntos críticos donde el comportamiento del modelo cambia cualitativamente.

```
times = 1000
```

La cantidad de iteraciones también puede ajustarse si es que el modelo demora en converger, aunque en el presente trabajo no sea el caso.

Para obtener los datos del modelo, se crea una matriz para cada variable donde cada fila corresponde con un valor diferente del parámetro  $r$ . Cada columna es una iteración para cada uno de los valores del parámetro.

```
{
sensi = length(R)

data.L <- matrix(0, nrow = sensi, ncol = times)
data.L[,1] = rdunif(1,a = 1, b = 150) #tamaño inicial de la poblacion
```

```

data.n <- matrix(0, nrow = sensi, ncol = times)
data.n[,1] = 0.02 # crecimiento de la poblacion inicial (se podría poner
                  # una tasa de crecimiento aleatoria: runif(1,0.01,0.04))

data.k <- matrix(0, nrow = sensi, ncol = times)
data.k[,1] = 1

data.h <- matrix(0, nrow = sensi, ncol = times)
data.h[,1] = 1

data.y <- matrix(0, nrow = sensi, ncol = times)
data.y[,1] = 1
}

```

Esta estructura de iteracion se puede ampliar para probar configuraciones de mas de un parametro. En este caso nos restringimos a estudiar los cambios que se producen en el modelo al variar uno de los parámetros.

```

for (i in 1:sensi) {
  for (t in 2:times) {

    r= R[i] # en cada loop r toma un valor diferente del vector R
    data.L[i,t] <- data.L[i,t-1]*exp(r*(1-(data.L[i,t-1]/M))) #Verhulst (1838)

    data.n[i, t] <- (data.L[i, t] / data.L[i, t - 1]) - 1
    data.k[i, t] <- (s_k * data.y[i, t - 1] + (1 - delta) * data.k[i, t - 1]) /
      (data.L[i, t] / data.L[i, t - 1]) #evolucion de k
    data.h[i, t] <- (s_h * data.y[i, t - 1] + (1 - delta) * data.h[i, t - 1]) /
      (data.L[i, t] / data.L[i, t - 1]) #evolucion de h
    data.y[i, t] <- data.k[i, t] ^ alpha * data.h[i, t] ^ beta
  }
}

```

## Estado estacionario

### Capital Fisico

$$k^* = \left[ \frac{s_k^{1-\beta} s_h^\beta}{\delta} \right]^{\frac{1}{1-\alpha-\beta}}$$

```
(k.star = ((s_k^(1-beta)*s_h^(beta))/(delta))^(1/(1-alpha-beta)))
```

```
## [1] 4500
```

### Capital Humano

$$h^* = \left[ \frac{s_h^{1-\alpha} s_k^\alpha}{\delta} \right]^{\frac{1}{1-\alpha-\beta}}$$

```
(h.star = ((s_h^(1-alpha)*s_k^(alpha))/(delta))^(1/(1-alpha-beta)))
```

```
## [1] 5400
```

### Capacidad de carga

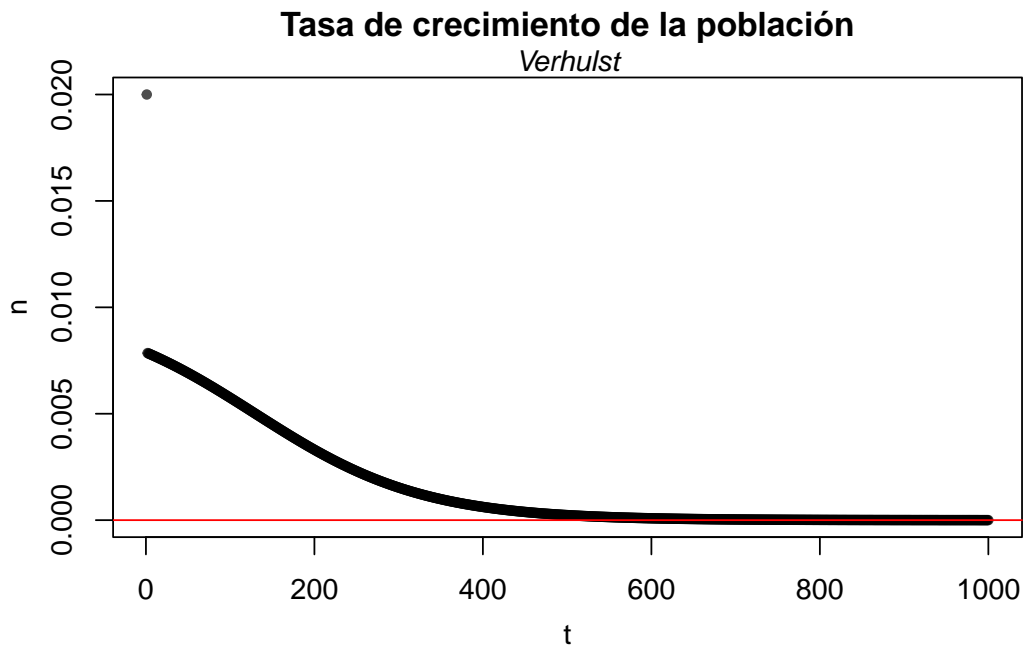
$$L^* = M \text{ (capacidad de carga)}$$

M = 500

## Graficos

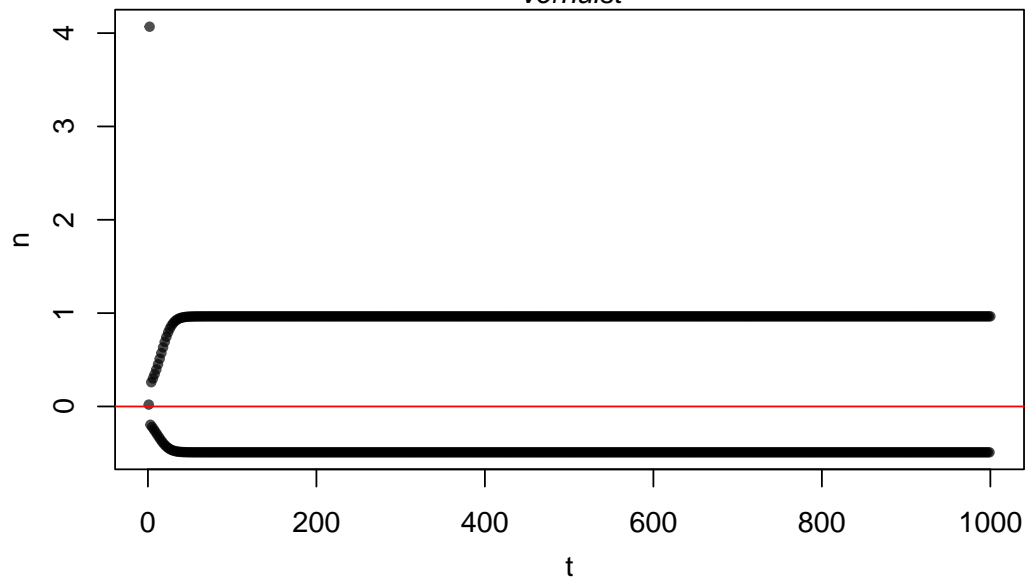
### Dinámica poblacional

```
for (i in c(1,30,50)) {  
  plot(data.n[i,],type = "p", pch = 20, xlab = "", ylab = "",  
        col=alpha("black", 0.7))  
  title("Tasa de crecimiento de la población", line = 1.2, cex.main = 1.2)  
  title(xlab="t", line=2.3, cex.lab=1) # xlab  
  title(ylab="n", line=2.3, cex.lab=1) # ylab  
  mtext("Verhulst", side = 3, font = 3) # sub  
  abline(h=0, lwd= 1, col= "red", lty = 1) # linea roja  
}
```



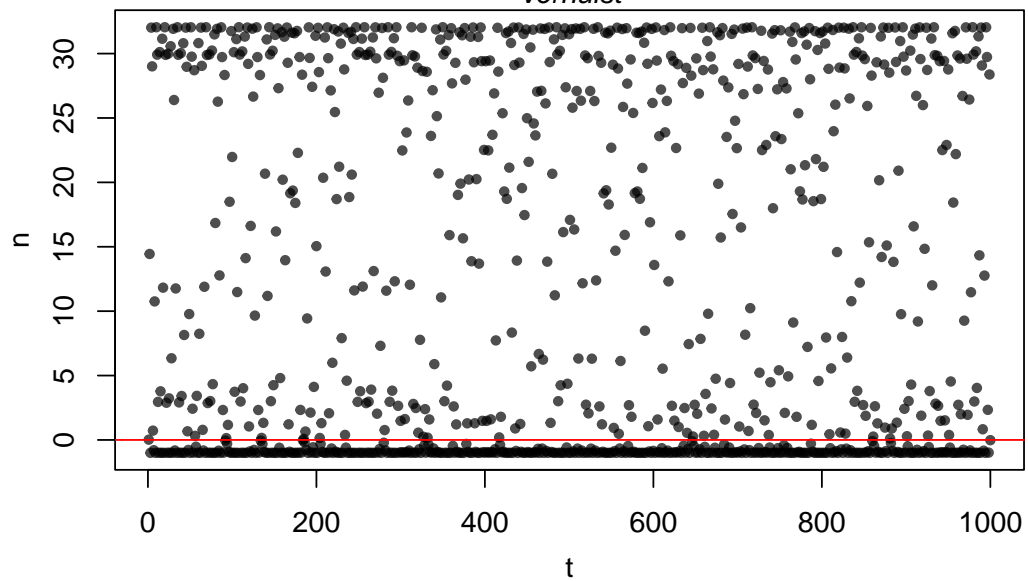
## Tasa de crecimiento de la población

*Verhulst*



## Tasa de crecimiento de la población

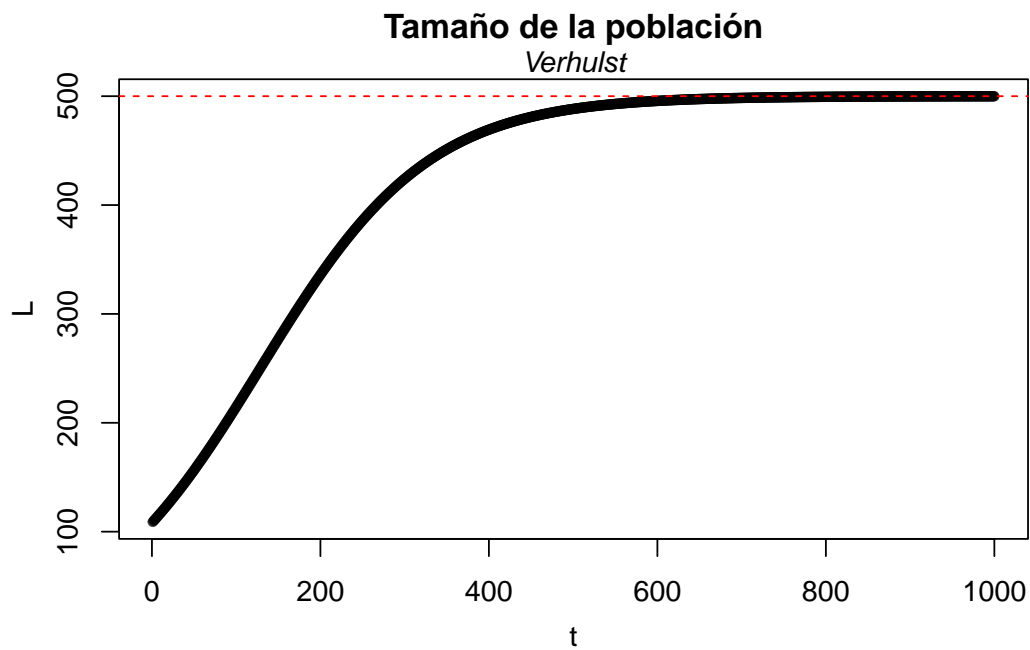
*Verhulst*

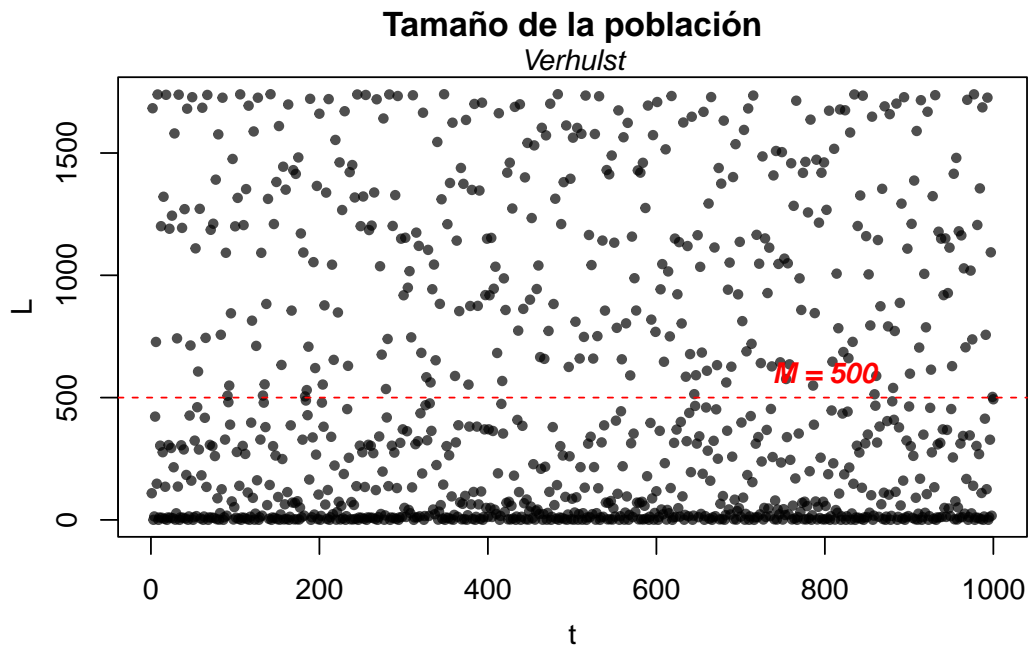
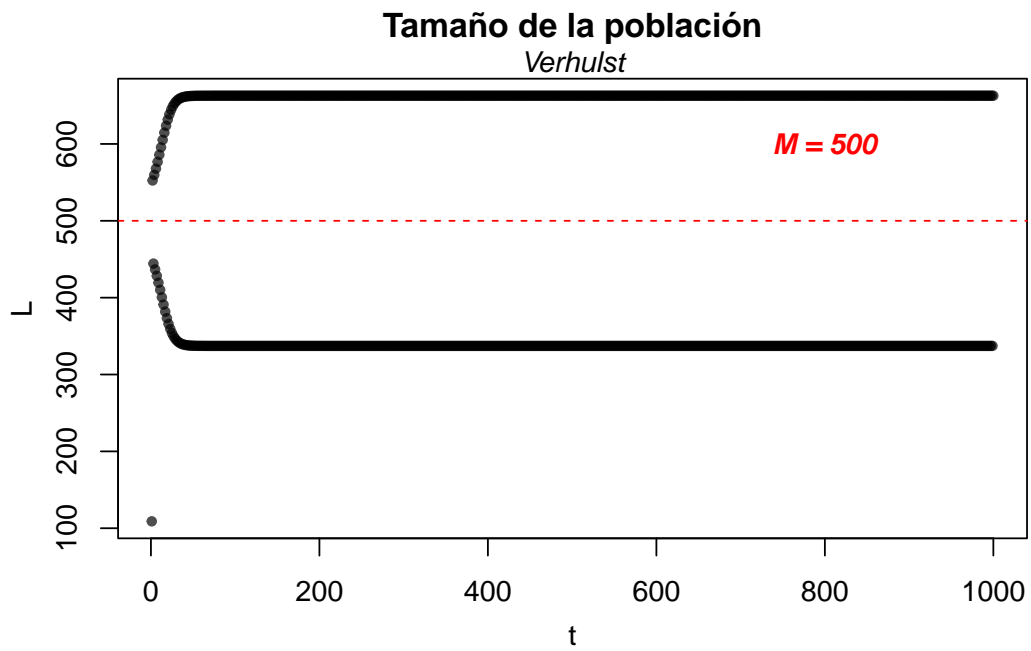


```

for (i in c(1,30,50)) {
plot(data.L[i,], type = "p", pch = 20, xlab = "", ylab = "", col=alpha("black", 0.7))
  title("Tamaño de la población", line = 1.2, cex.main = 1.2) #titulo
  title(xlab="t", line=2.3, cex.lab=1) # xlab
  title(ylab="L", line=2.3, cex.lab=1) # ylab
  mtext("Verhulst", side = 3, font = 3) # sub
  abline(h = M, col="red",lwd=1, lty = 2) # linea roja
  text(800, 600, "M = 500", font =4, cex.main =2, col= "red")
}

```

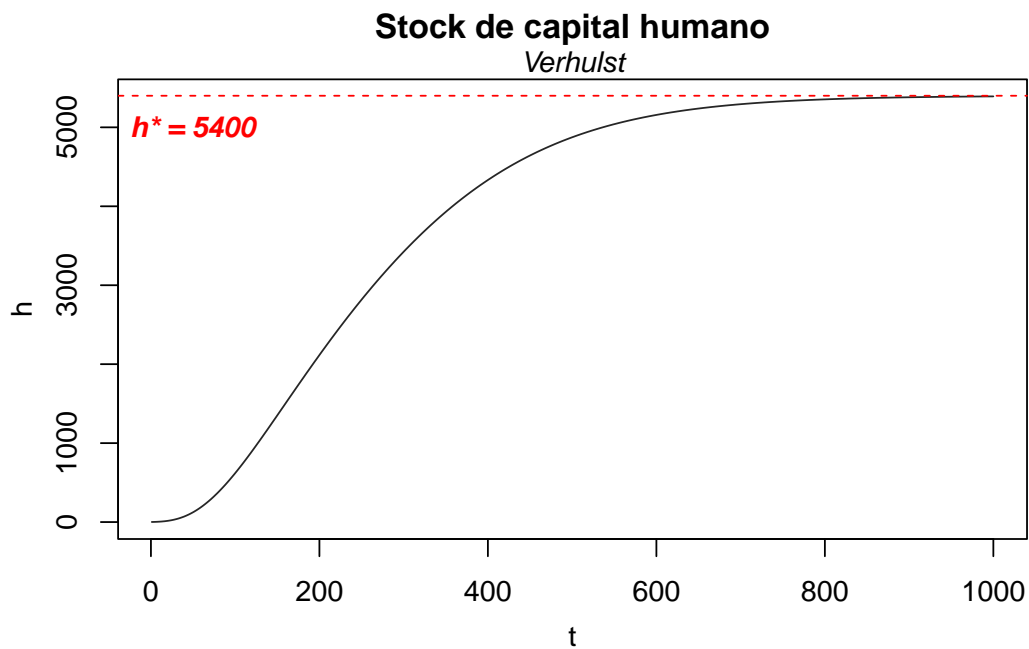


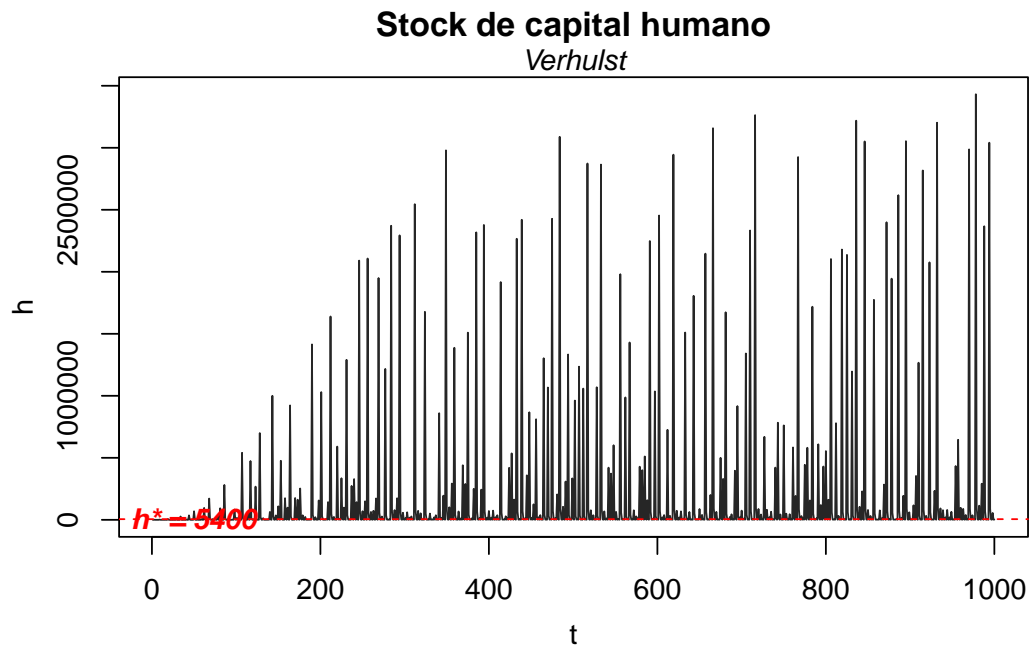
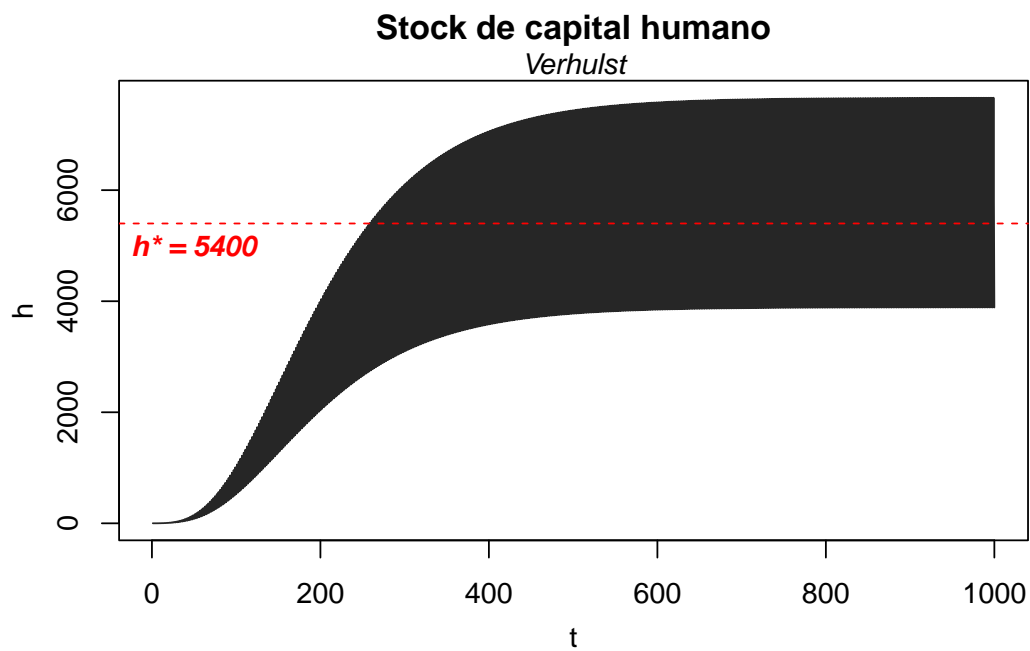




## Dinamica del Capital

```
for (i in c(1,30,50)) {  
  plot(data.h[i,], type = "l", pch = 20,xlab = "", ylab = "", col=alpha("black", 0.85))  
  title("Stock de capital humano", line = 1.2) #title  
  title(xlab="t", line=2.3, cex.lab=1) # xlab  
  title(ylab="h", line=2.3, cex.lab=1) # ylab  
  mtext("Verhulst", side = 3, font = 3) #sub  
  abline(h = h.star, col="red",lwd=1, lty = 2) # linea roja  
  text(50, 5000, "h* = 5400", font =4, cex.main =2, col= "red")  
}
```

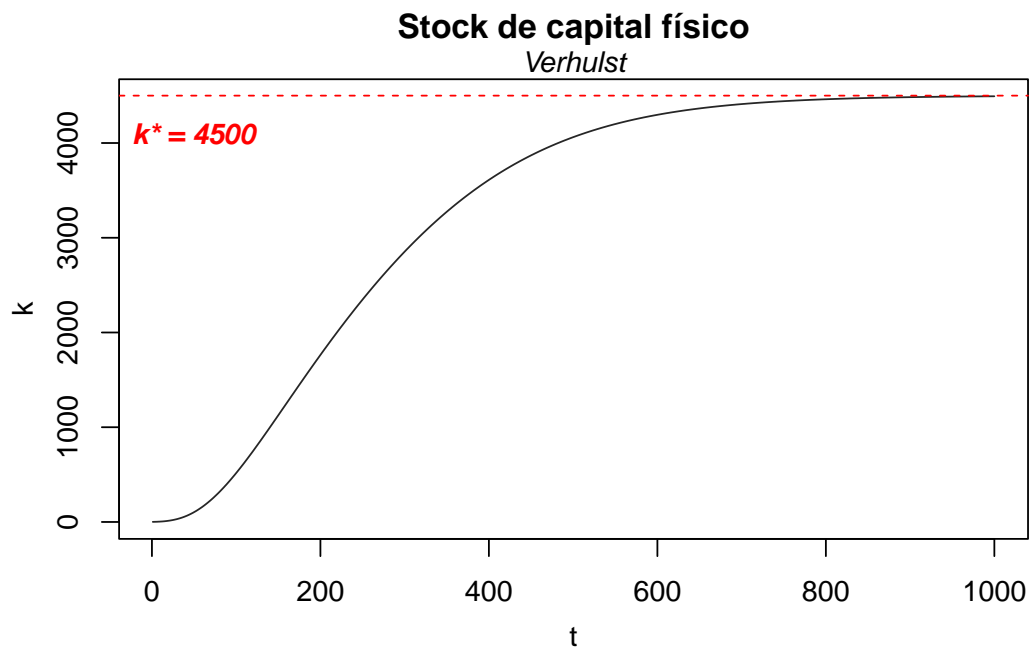


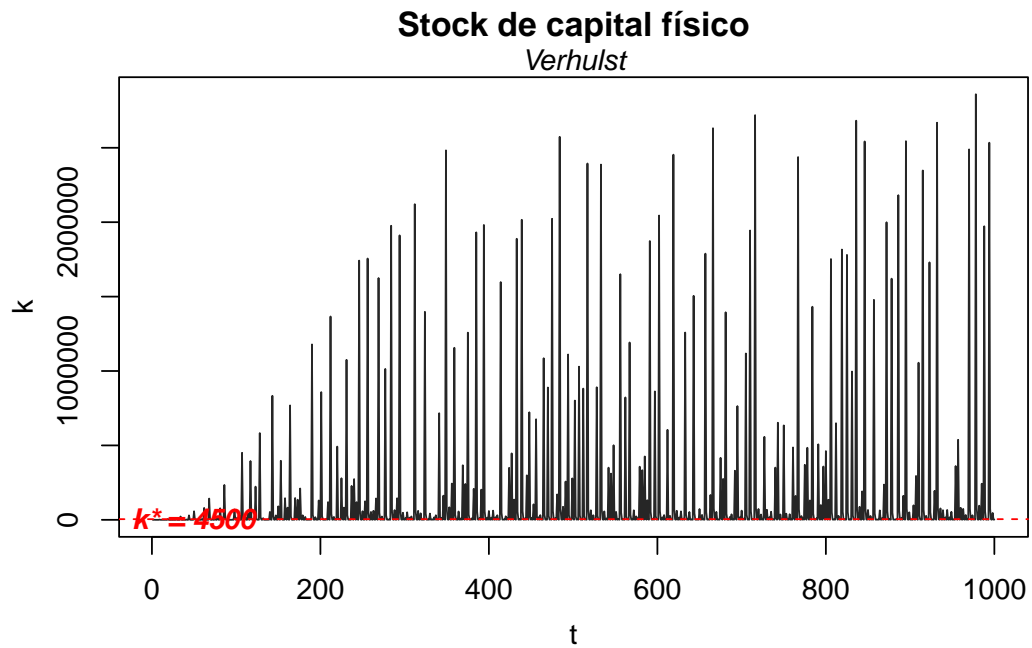
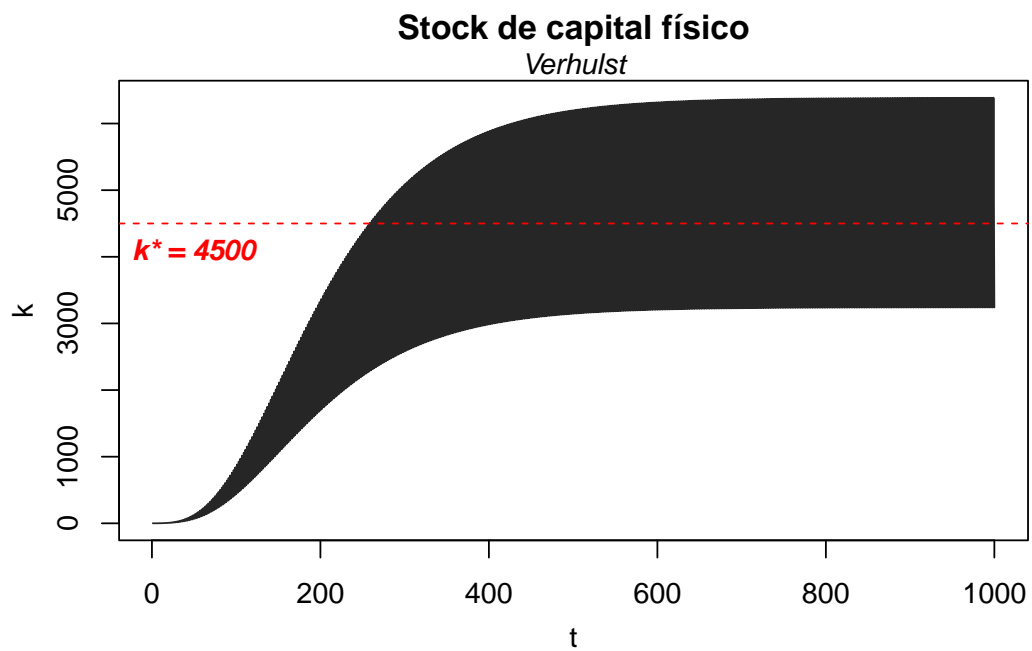


```

for (i in c(1,30,50)) {
plot(data.k[i,], type = "l", xlab = "", ylab = "", col=alpha("black", 0.85))
  title("Stock de capital físico", line = 1.2)
  title(xlab="t", line=2.3, cex.lab=1) # xlab
  title(ylab="k", line=2.3, cex.lab=1) # ylab
  mtext("Verhulst", side = 3, font = 3)
  abline(h = k.star, col="red",lwd=1, lty =2)
  text(50, 4100, "k* = 4500", font =4, cex.main =2, col= "red")
}

```





## Ricker (1954)

Para observar el comportamiento del modelo de Ricker, se repitió el mismo procedimiento que para la especificación de Verhulst. Se usaron los mismos parametros de eficiencia y ahorro, y a pesar de que en Ricker hayan dos parámetros se verá la evolución de las variables para distintos valores de  $a$ . No obstante, el parámetro  $b$  depende de la capacidad de carga y  $a$  por lo que también variará en cada loop pero la variación exógena es de  $a$ .

### Parámetros modelo poblacional

```
M <- 500
A <- seq(0.01,20, length= 60)
```

Como en el modelo anterior, la capacidad de carga (M) puede cambiar y la longitud del vector A también. En este caso el intervalo de valores es más grande porque donde el equilibrio se vuelve inestable es en  $e^2$ .

```
{
times = 1000
sensi = length(A)

data.L <- matrix(0, nrow = sensi, ncol = times)
data.L[,1] = rdunif(1,a = 1, b = 150)

data.n <- matrix(0, nrow = sensi, ncol = times)
data.n[,1] = 0.02

data.k <- matrix(0, nrow = sensi, ncol = times)
data.k[,1] = 1

data.h <- matrix(0, nrow = sensi, ncol = times)
data.h[,1] = 1

data.y <- matrix(0, nrow = sensi, ncol = times)
data.y[,1] = 1
}
for (i in 1:sensi) {
  for (t in 2:times) {

    a = A[i]
    b = log(a) / M # en cada loop cambia tanto el valor de a como de b
    data.L[i, t] <- a * data.L[i, t - 1] * exp(-b * data.L[i, t - 1])

    data.n[i, t] <- (data.L[i, t] / data.L[i, t - 1]) - 1
    data.k[i, t] <- (s_k * data.y[i, t - 1] + (1 - delta) * data.k[i, t - 1]) /
      (data.L[i, t] / data.L[i, t - 1]) #evolucion de k
    data.h[i, t] <- (s_h * data.y[i, t - 1] + (1 - delta) * data.h[i, t - 1]) /
```

```

      (data.L[i, t] / data.L[i, t - 1]) #evolucion de h
    data.y[i, t] <- data.k[i, t] ^ alpha * data.h[i, t] ^ beta
  }}

```

## Estado estacionario

### Capital Fisico

$$k^* = \left[ \frac{s_k^{1-\beta} s_h^\beta}{\delta} \right]^{\frac{1}{1-\alpha-\beta}}$$

```

(k.star = ((s_k^(1-beta)*s_h^(beta))/(delta))^(1/(1-alpha-beta)))

```

```
## [1] 4500
```

### Capital Humano

$$h^* = \left[ \frac{s_h^{1-\alpha} s_k^\alpha}{\delta} \right]^{\frac{1}{1-\alpha-\beta}}$$

```

(h.star = ((s_h^(1-alpha)*s_k^(alpha))/(delta))^(1/(1-alpha-beta)))

```

```
## [1] 5400
```

### Capacidad de carga

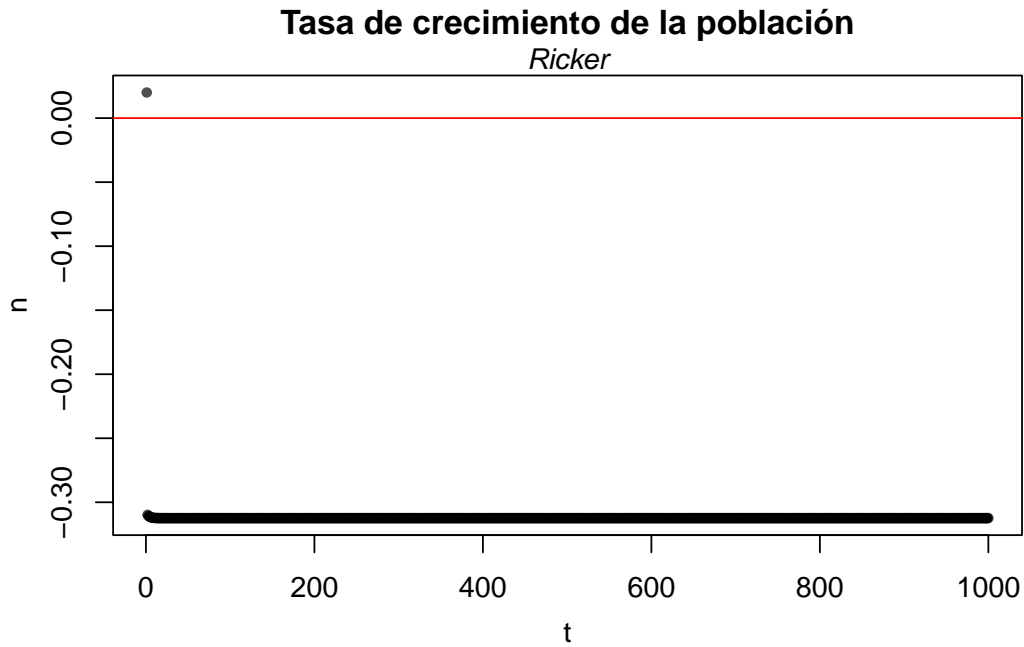
$$L^* = M \text{ (capacidad de carga)}$$

M = 500

## Graficos

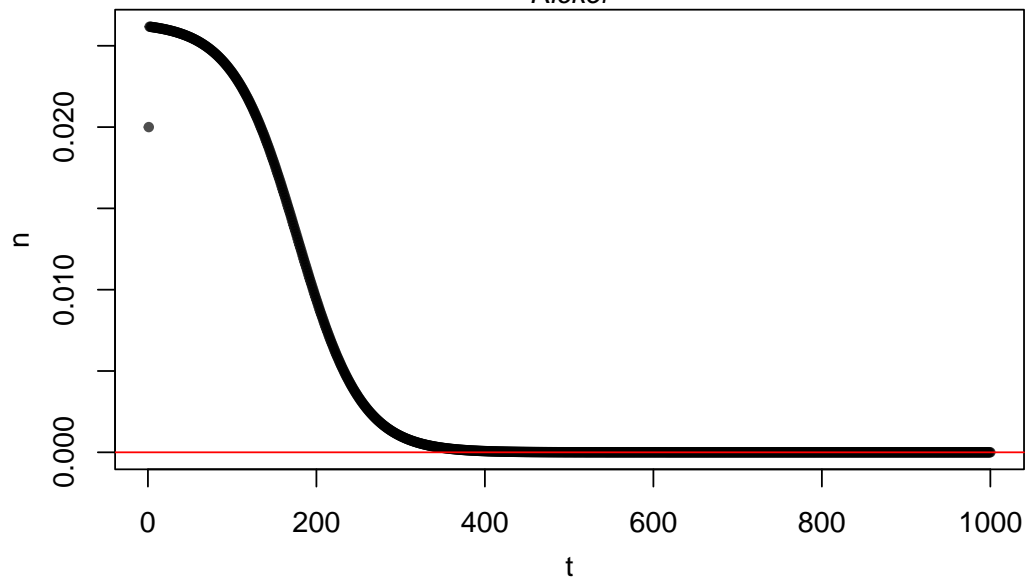
### Dinámica poblacional

```
for (i in c(3,4,60)) {  
  plot(data.n[i,], type = "p", pch = 20, xlab = "", ylab = "",  
        col=alpha("black", 0.7))  
  title("Tasa de crecimiento de la población", line = 1.2, cex.main = 1.2)  
  title(xlab="t", line=2.3, cex.lab=1) # xlab  
  title(ylab="n", line=2.3, cex.lab=1) # ylab  
  mtext("Ricker", side = 3, font = 3) # sub  
  abline(h=0, lwd= 1, col= "red", lty = 1) # linea roja  
}
```



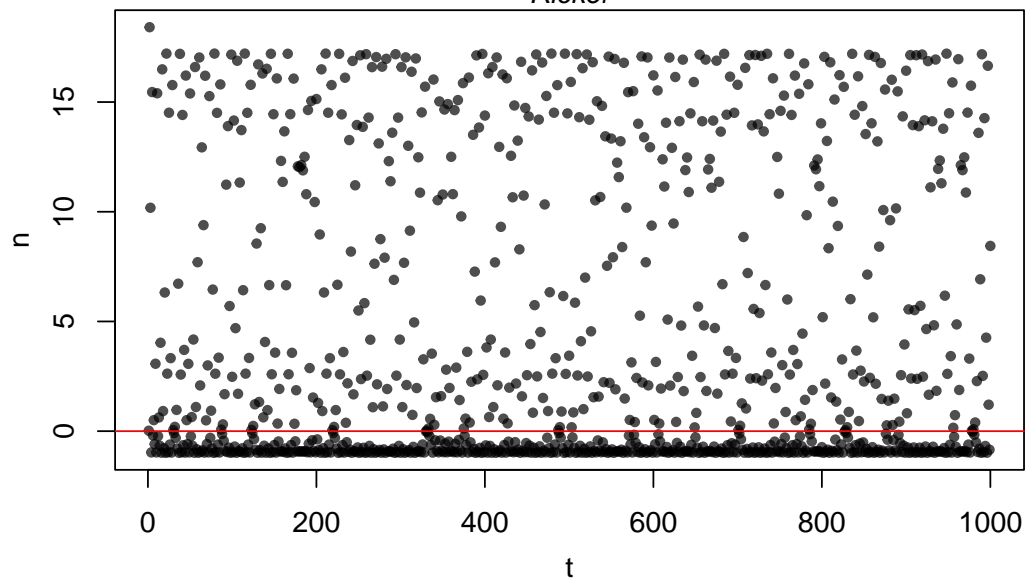
## Tasa de crecimiento de la población

*Ricker*



## Tasa de crecimiento de la población

*Ricker*

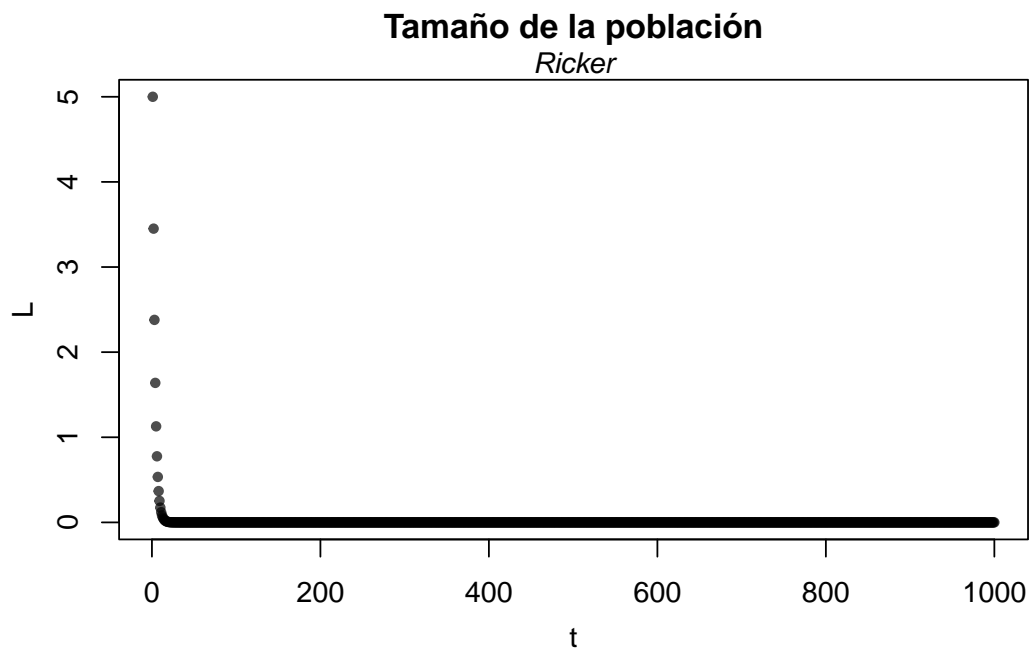


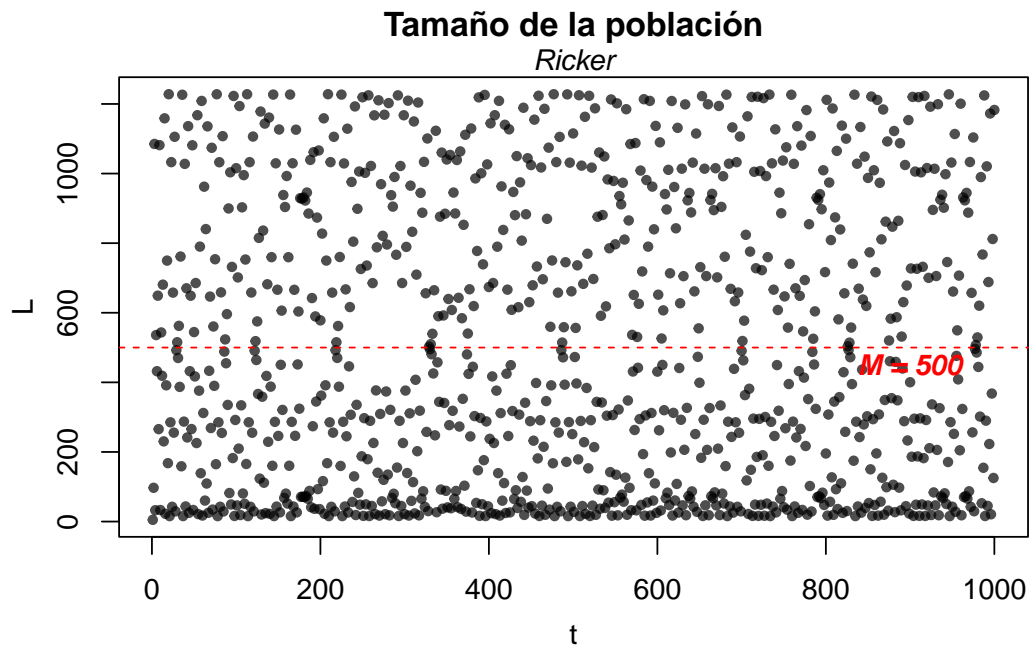
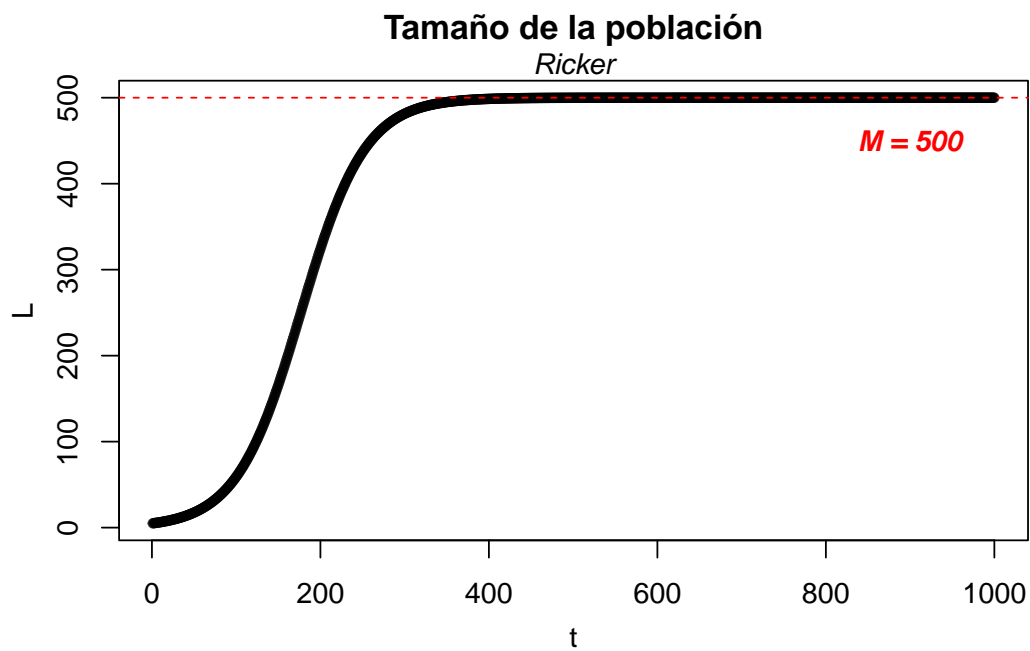


```

for (i in c(3,4,60)) {
  plot(data.L[i,],type = "p", pch = 20, xlab = "", ylab = "",
       col=alpha("black", 0.7))
  title("Tamaño de la población", line = 1.2, cex.main = 1.2) #titulo
  title(xlab="t", line=2.3, cex.lab=1) # xlab
  title(ylab="L", line=2.3, cex.lab=1) # ylab
  mtext("Ricker", side = 3, font = 3) # sub
  abline(h = M, col="red",lwd=1, lty = 2) # línea roja
  text(900, 450, "M = 500", font =4, cex.main =2, col= "red")
}

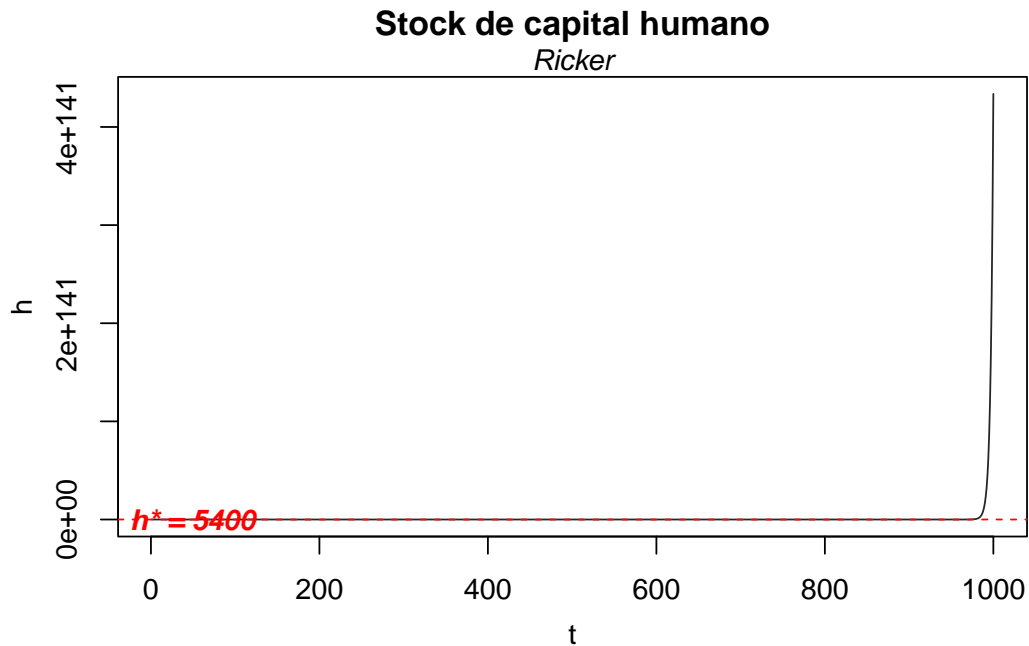
```

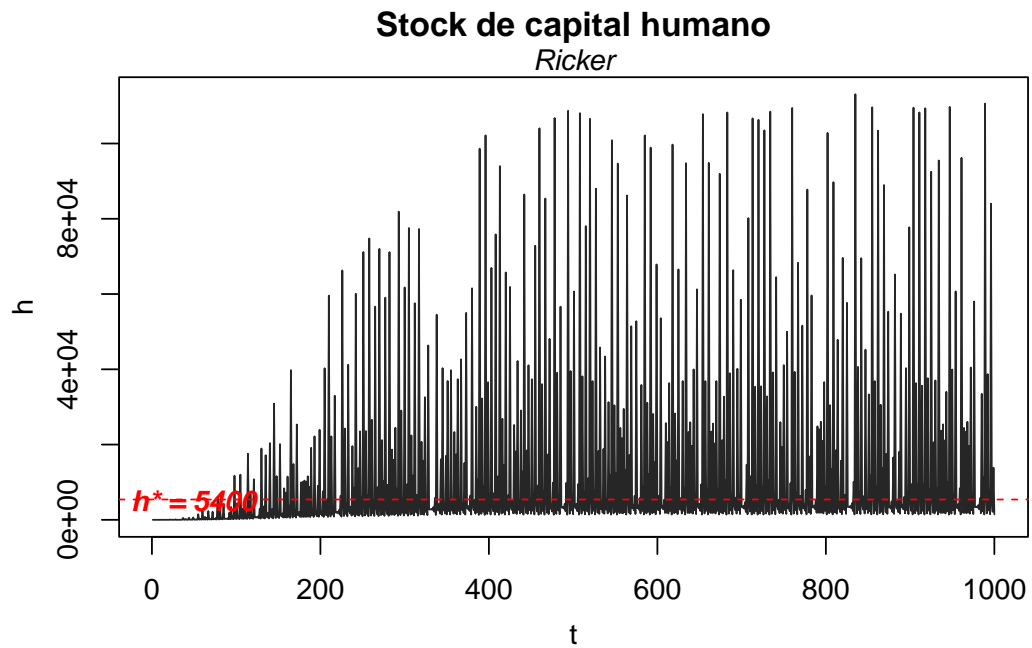
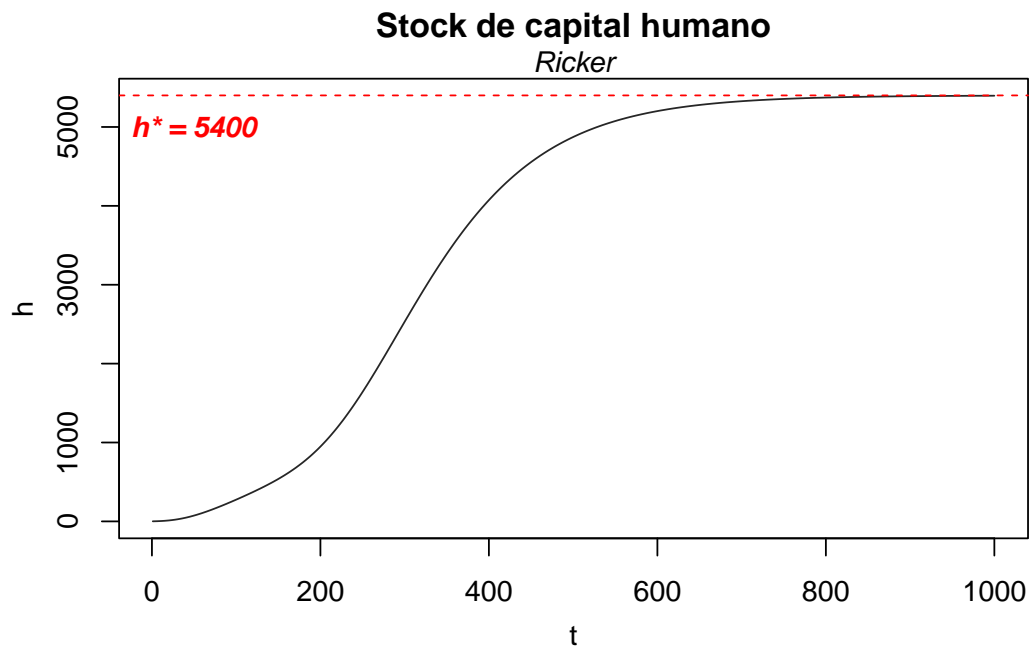




## Dinamica del Capital

```
for (i in c(3,4,60)) {  
  plot(data.h[i,], type = "l", pch = 20, xlab = "", ylab = "",  
        col=alpha("black", 0.85))  
  title("Stock de capital humano", line = 1.2) # title  
  title(xlab="t", line=2.3, cex.lab=1) # xlab  
  title(ylab="h", line=2.3, cex.lab=1) # ylab  
  mtext("Ricker", side = 3, font = 3) # sub  
  abline(h = h.star, col="red",lwd=1, lty = 2) # linea roja  
  text(50, 5000, "h* = 5400", font =4, cex.main =2, col= "red")  
}
```

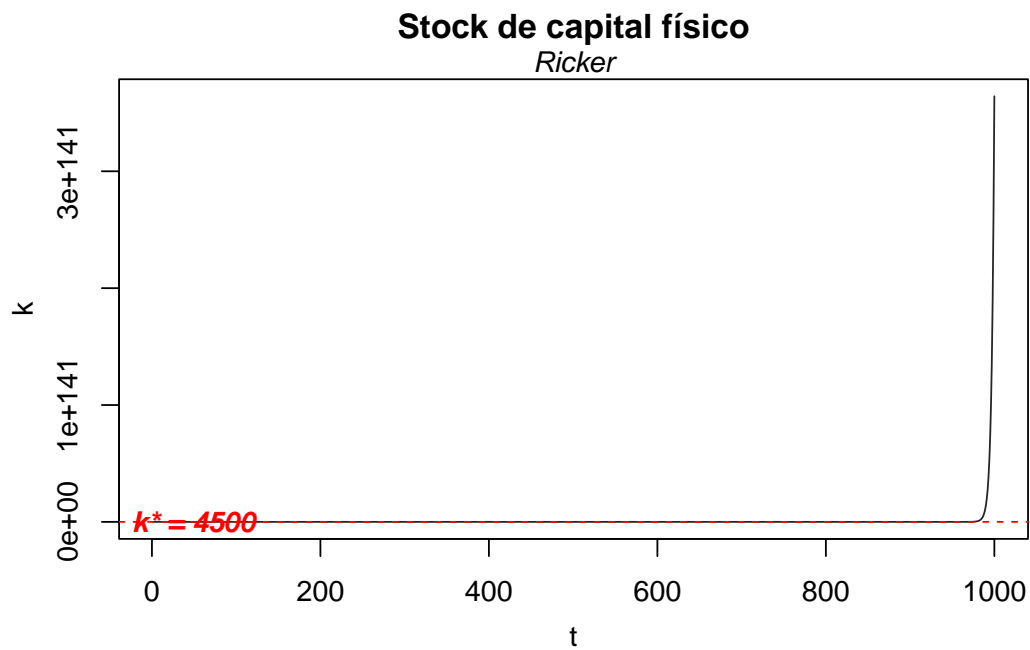


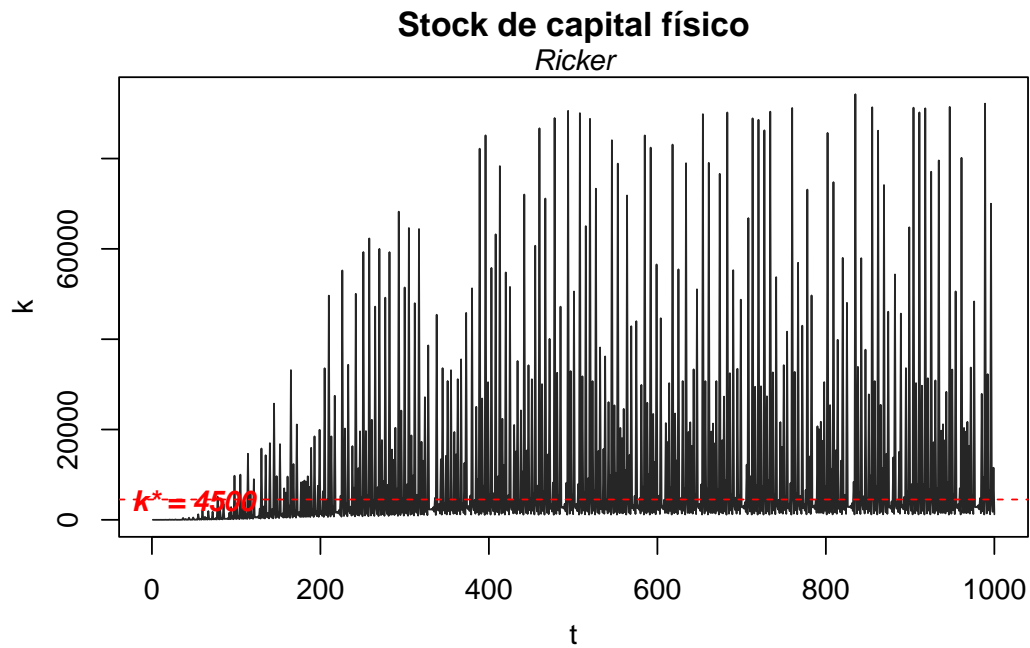
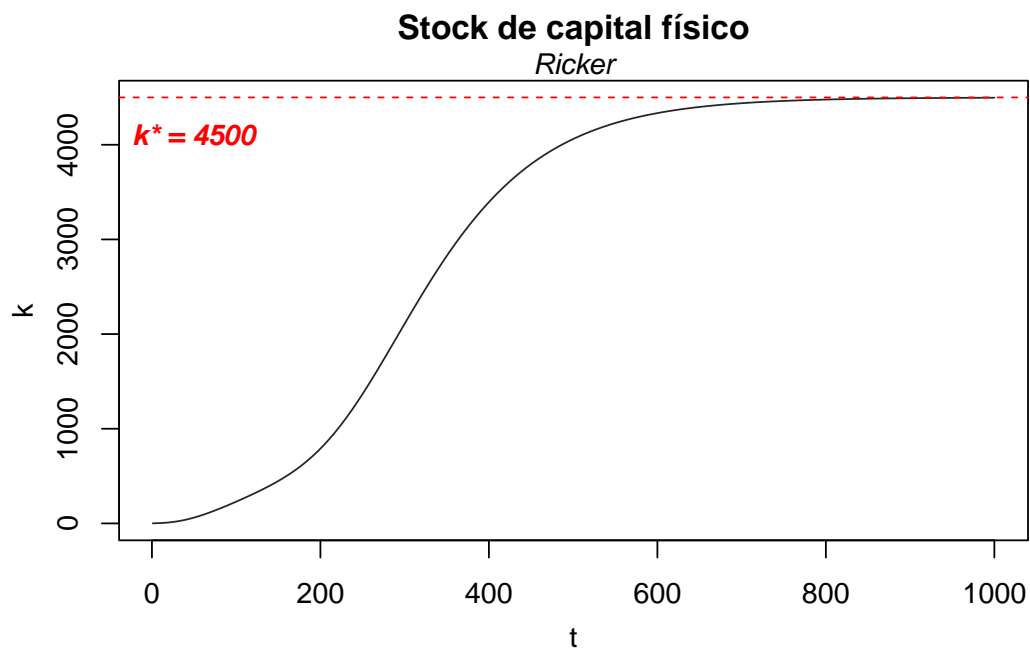


```

for (i in c(3,4,60)) {
plot(data.k[i,],type = "l", xlab = "", ylab = "", col=alpha("black", 0.85))
  title("Stock de capital físico", line = 1.2)
  title(xlab="t", line=2.3, cex.lab=1) # xlab
  title(ylab="k", line=2.3, cex.lab=1) # ylab
  mtext("Ricker", side = 3, font = 3)
  abline(h = k.star, col="red",lwd=1, lty =2)
  text(50, 4100, "k* = 4500", font =4, cex.main =2, col= "red")
}

```





## Especificaciones de leyes generales de poblacion

Por ultimo, dejamos planteadas otras posibles formas funcionales que cumplen con las condiciones señaladas por Cayssials (2018), a saber:

1. cuando el tamaño de la población es pequeña, crece a tasa constante  $n_0$  (crecimiento exponencial)
2. cuando la población se acerca a la capacidad de carga, los recursos escasean y disminuye la tasa de crecimiento  $n_0$
3. la tasa de crecimiento  $n_0$  tiende a cero conforme pasa el tiempo.

### Ley de poblacion de Verhulst (1838)

Parametros:  $\{r, M\}$

$$L_{t+1} = L_t \cdot e^{r(1-\frac{L_t}{M})}$$

```
L[t] <- L[t-1]*exp(r*(1-(L[t-1])/M))
```

### Ley de poblacion de Beverton (1957)

Parametros:  $\{a, b\}$

$$L_{t+1} = \frac{aL_t}{1 + bL_t}$$

```
L[t] <- (a*L[t-1])/(1+b*L[t-1])
```

### Ley de poblacion de Ricker (1954)

Parametros:  $\{a, b\}$

$$L_{t+1} = aL_t \cdot e^{-bL_t}$$

```
L[t] <- a*L[t-1]*exp(-b*L[t-1])
```

### Ley de poblacion de Hassell (1975)

Parametros:  $\{a, b, c\}$

$$L_{t+1} = \frac{aL_t}{(1 + bL_t)^c}$$

```
L[t] <- (a*L[t-1]) / (1+b*L[t-1])^c
```