El modelo de crecimiento de Von Bertalanffy.

8 de diciembre de 2011

Deducción del modelo de crecimiento de Von Bertalanffy.

El *modelo de crecimiento de Von Bertalanffy* se utiliza ampliamente en biología marina para modelar la relación edad-talla de los peces, así como de algunas otras especies. En esencia, este modelo parte de las siguientes observaciones:

1. El crecimiento de los peces no se produce a la misma velocidad a lo largo del tiempo.

2. La velocidad de crecimiento inicial es muy rápida; después, a medida que el pez aumenta de tamaño y madura sexualmente, la tasa de crecimiento va decreciendo poco a poco hasta ser prácticamente nula (el pez deja de crecer).

La gráfica siguiente muestra el comportamiento típico de la curva de crecimiento de los peces:

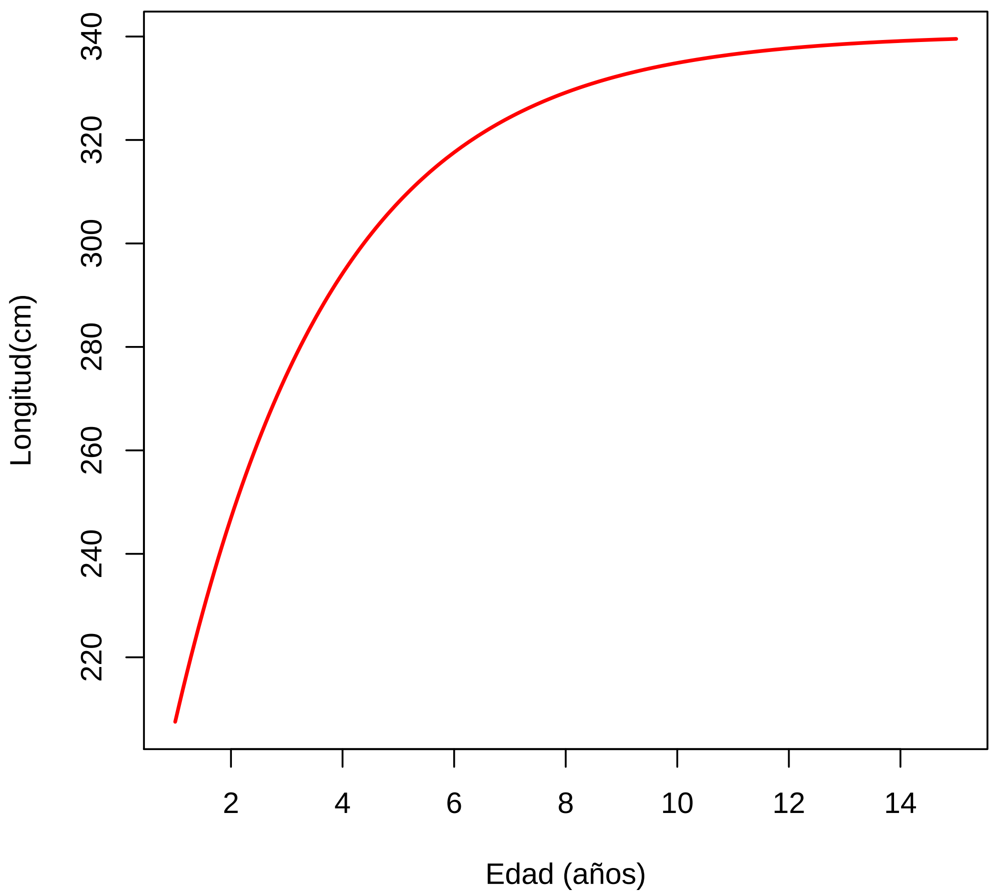

Figura 1: Curva de crecimiento típica de los peces

Para modelar este comportamiento, el modelo de Von Bertalanffy define $L(t)$ como la longitud de un pez cuando tiene la edad t, y L_∞ como la longitud máxima que llega a alcanzar ese pez. Además considera que, en cualquier instante t, la tasa de crecimiento del pez es proporcional a la longitud que aún le falta para alcanzar la talla máxima:

$$L'(t) = r\left(L_\infty - L(t)\right) \tag{1}$$

De esta forma, al comienzo de la vida del pez la diferencia $L_\infty - L(t)$ es grande y por tanto su velocidad de crecimiento $L'(t)$ es alta; asimismo, cuando el pez alcanza una talla próxima a L_∞ la diferencia $L_\infty - L(t)$ es pequeña y consecuentemente también lo es la tasa de crecimiento $L'(t)$.

Tarea 1: resolver la ecuación diferencial anterior, utilizando como condición inicial que $L(t_0) = 0$. (*Nota:* este valor de t_0 habitualmente será negativo; se asume que $t = 0$ es el momento de la eclosión o nacimiento del pez, ya en ese instante su longitud $L(0)$ es mayor que cero, por lo que la longitud 0 se produce en un t_0 anterior a 0)

Sugerencia: Para resolver esta ecuación diferencial y hallar explícitamente la función $L(t)$, la manera más sencilla es pasar en primer lugar los términos en $L(t)$ al primer miembro, esto es:

$$\frac{L'(t)}{L_\infty - L(t)} = r$$

y a continuación integrar en ambos términos de la igualdad:

$$\int_{t_0}^{t} \frac{L'(s)}{L_\infty - L(s)}\, ds = \int_{t_0}^{t} r \cdot ds \tag{2}$$

Discretización de la ecuación diferencial inicial $L'(t) = r\left(L_\infty - L(t)\right)$

La ecuación (1) que da origen al modelo de Von Bertalanffy puede escribirse también como:

$$\lim_{\Delta t \to 0} \frac{L(t+\Delta t) - L(t)}{\Delta t} = r\left(L_\infty - L(t)\right)$$

Así pues, para un valor dt suficientemente pequeño se tiene que:

$$\frac{L(t+dt) - L(t)}{dt} \simeq r\left(L_\infty - L(t)\right)$$

o, de manera equivalente:

$$L(t+dt) \simeq L(t) + dt \cdot r\left(L_\infty - L(t)\right)$$

Ello sugiere el siguiente algoritmo para obtener numéricamente los sucesivos valores de $L(t)$:

- **Paso 1.** *Inicialización:* fijar los valores de los parámetros t_0, L_∞, r, la longitud del *paso* dt*,* y el tiempo máximo t_{max} de observación. Hacer $k = 0$, $t = t_0$ y fijar $L_0 = L(t_0) = 0$.
- **Paso 2.** Mientras $t < t_{max}$:
  - Hacer $k = k + 1$.
  - Actualizar $t = t + dt$
  - calcular $L_k = L_{k-1} + dt \cdot r(L_\infty - L_{k-1})$
- **Paso 3.** *Fin:* representar gráficamente L frente a t.

Algoritmo 1: algoritmo de discretización para aproximar el modelo de Von Bertalanffy

Tarea 2: Implementar el algoritmo anterior en una función que reciba como argumentos los valores de t_0, L_∞, r, dt y t_{max}. Ejecutar dicho algoritmo con los valores $L_\infty = 300$, $r = 0{,}10$, $t_0 = -1$. $dt = 0{,}8$ y $t_{max} = 12$.

Tarea 3: Superponer al gráfico generado en la tarea anterior la curva calculada en la tarea 1, utilizando los mismos valores de los parámetros L_∞, t_0, r y t_{max}.

Tarea 4: Explora el efecto de variar el valor de dt en el algoritmo elaborado en la tarea 2. ¿Cuando se consigue el mejor ajuste entre la figura obtenida en la tarea 2 y la obtenida en la tarea 3?.

Ajuste de los parámetros del modelo de Von Bertalanffy.

En la práctica los parámetros L_∞, r y t_0 se desconocen, y es preciso aproximarlos a partir de los datos proporcionados por una muestra de peces para los que se dispone de su edad (calculada a partir de los otolitos) y su talla. La tabla y la figura siguientes (figura 2) muestran un ejemplo de los datos disponibles habitualmente, en este caso edades y tallas de una muestra de 20 peces.

La distribución de la nube de puntos indica claramente que el modelo de Von Bertalanffy puede describir bien la relación entre edad y talla en estos peces, pero ¿cuáles serían los valores adecuados de L_∞, r y t_0?

Método de mínimos cuadrados.

Este método permite responder a la pregunta anterior. La idea es encontrar los valores de L_∞, r y t_0 que hacen que la función $L(t) = L_\infty\left(1 - e^{-r(t-t_0)}\right)$ pase lo más cerca posible del centro de la nube de

	edad	longitud
1	11.00	328.50
2	3.90	284.30
3	6.90	329.80
4	11.00	338.60
5	7.30	326.80
6	9.70	332.60
7	3.10	270.30
8	3.10	281.60
9	1.20	218.30
10	2.60	253.80
11	8.50	336.00
12	11.90	331.30
13	12.40	345.70
14	11.70	339.30
15	13.90	334.30
16	8.30	323.30
17	5.80	324.20
18	4.80	305.60
19	4.90	301.90
20	7.90	317.60

puntos, minimizando la suma de los cuadrados de las distancias de los puntos observados a la función que los representa en el modelo. Así, si denotamos por (t_i, y_i) el $i-$ésimo pez observado, el modelo de Von Bertalanffy predice para un pez con esa edad t_i la longitud $\hat{y}_i = L(t_i) = L_\infty \left(1 - e^{-r(t_i - t_0)}\right)$. Minimizar la suma de cuadrados de las distancias entre las observaciones y las predicciones del modelo es equivalente entonces a minimizar la función:

$$\sum_{i=1}^{20} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{20} \left(y_i - L_\infty \left(1 - e^{-r(t_i - t_0)}\right)\right)^2$$

Para minimizar esta función debemos calcular sus derivadas e igualar a cero:

$$\begin{aligned}
\frac{\partial}{\partial L_\infty} \left(y_i - L_\infty \cdot \left(1 - e^{-r(t_i - t_0)}\right)\right)^2 &= 2 \sum_{i=1}^{20} \left(e^{-(t_i - t_0) r} - 1\right) \left(y_i - L_\infty \left(1 - e^{-(t_i - t_0) r}\right)\right) = 0 \\
\frac{\partial}{\partial t_0} \left(y_i - L_\infty \cdot \left(1 - e^{-r(t_i - t_0)}\right)\right)^2 &= 2\, r\, L_\infty \sum_{i=1}^{20} e^{-(t_i - t_0)\, r} \left(y_i - L_\infty \left(1 - e^{-(t_i - t_0) r}\right)\right) = 0 \\
\frac{\partial}{\partial r} \left(y_i - L_\infty \cdot \left(1 - e^{-r \cdot (t_i - t_0)}\right)\right)^2 &= 2\, L_\infty \sum_{i=1}^{20} (t_0 - t_i)\; e^{-(t_i - t_0)\, r} \left(y_i - L_\infty \left(1 - e^{-(t_i - t_0) r}\right)\right) = 0
\end{aligned}$$

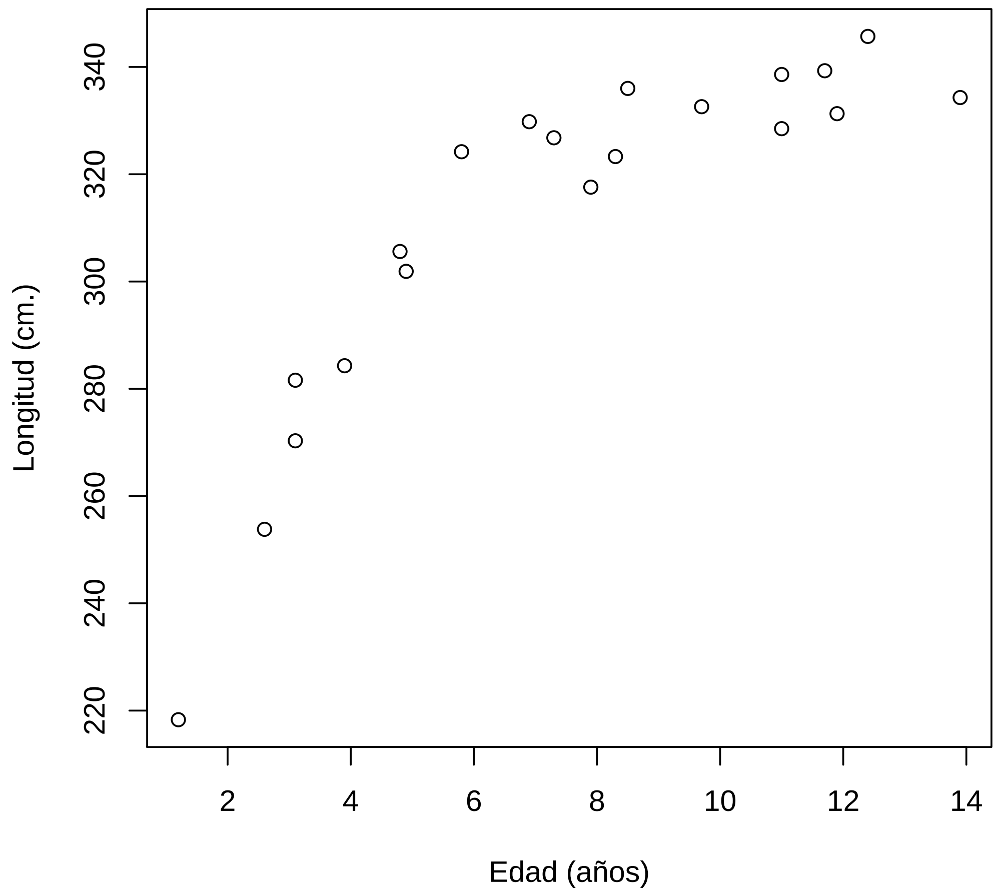

Figura 2: Nube de puntos *edad-talla*

De la segunda ecuación se sigue que:

$$\sum_{i=1}^{20} e^{-(t_i-t_0)\,r}\ \left(y_i - L_\infty\left(1-e^{-(t_i-t_0)r}\right)\right) = 0 \qquad (3)$$

Sustituyendo en la primera ecuación se tiene que:

$$\sum_{i=1}^{20} \left(y_i - L_\infty\left(1-e^{-(t_i-t_0)r}\right)\right) = 0 \qquad (4)$$

y sustituyendo en la tercera:

$$\sum t_i\, e^{-(t_i-t_0)\,r}\ \left(y_i - L_\infty\left(1-e^{-(t_i-t_0)r}\right)\right) = 0 \qquad (5)$$

¡¡ IMPOSIBLE OBTENER UNA SOLUCIÓN EXPLÍCITA ANALÍTICAMENTE!!!

$$\Downarrow$$

¡¡¡ES NECESARIO UTILIZAR MÉTODOS NUMÉRICOS!!!

Tarea 5: *Inventar* un método para obtener una aproximación inicial de los valores de L_∞, t_0 y r a partir de la observación de la nube de puntos de la figura 2.

Aproximación 1: Resolver directamente las ecuaciones normales mediante fsolve.

Podemos desarrollar la ecuación 4 del siguiente modo (sustituimos el número de observaciones, 20, por n para dar un carácter más general a la solución):

$$\begin{aligned}\sum_{i=1}^{n} y_i - L_\infty \sum_{i=1}^{n}\left(1-e^{-(t_i-t_0)r}\right) &= 0\\ \sum_{i=1}^{n} y_i - L_\infty \left(n-\sum_{i=1}^{n} e^{-(t_i-t_0)r}\right) &= 0\end{aligned}$$

de donde:

$$L_\infty = \frac{\sum_{i=1}^{n} y_i}{n-\sum_{i=1}^{n} e^{-(t_i-t_0)r}} \tag{6}$$

Sustituyendo el valor de L_∞ en la ecuación 3

$$\sum_{i=1}^{n} e^{-(t_i-t_0)\,r}\left(y_i - \frac{\sum_{i=1}^{n} y_i}{n-\sum_{i=1}^{n} e^{-(t_i-t_0)r}}\left(1-e^{-(t_i-t_0)r}\right)\right)=0 \tag{7}$$

y en la ecuación 5:

$$\sum t_i\, e^{-(t_i-t_0)\,r}\left(y_i - \frac{\sum_{i=1}^{n} y_i}{n-\sum_{i=1}^{n} e^{-(t_i-t_0)r}}\left(1-e^{-(t_i-t_0)r}\right)\right)=0 \tag{8}$$

se obtiene un sistema no lineal de dos ecuaciones y dos incógnitas (t_0 y r), ya que los valores de t_i e y_i son conocidos (se han obtenido experimentalmente). Pueden resolverse utilizando fsolve(), implementando previamente el sistema de ecuaciones anterior:

```
function f = sistemaBertal(x)
    t0=x(1);
    r=x(2) ;
    f= zeros (2, 1);
```

```
    [ t y ]=textread ( 'bertalan.csv' , '%f %f' , 'headerlines' , 1 ) ;
    aux=exp(-r*(t-t0));
    sumy=sum(y);
    sumaux=sum(aux);
    n=length(t);
    Linf=sumy/(n-sumaux);
    eq1=aux.*(y-Linf*(1-aux));
    eq2=t.*eq1;
    f(1) =sum(eq1);
    f(2) = sum(eq2);
end %function
```

Ahora, ejecutando fsolve() encontramos la solución del sistema:

```
>>> x=fsolve(@sistemaBertal,[-1.5,0.5])
x =
  -1.74433    0.33910
```

Para calcular el valor de L_∞ implementamos una nueva función:

```
function Linf = calculaLinf(x)
    t0=x(1);
    r=x(2) ;
    [ t y ]=textread ( 'bertalan.csv' , '%f %f' , 'headerlines' , 1 ) ;
    aux=exp(-r*(t-t0));
    sumy=sum(y);
    sumaux=sum(aux);
    n=length(t);
    Linf=sumy/(n-sumaux);
end %function
```

y la empleamos con llos valores de t_0 y r que se acaban de hallar:

```
>>> calculaLinf(x)
ans =   340.21
```

Utilizando otra aproximación inicial encontramos otra solución:

```
>>> x=fsolve(@sistemaBertal,[-2,1])
x =
  -22.59721     0.84028
>>> calculaLinf(x)
```

```
ans =  311.19
```

Podemos representar estas soluciones gráficamente, junto con los puntos observados:

```
[ edad longitud]=textread ( 'bertalan.csv' , '%f  %f' , 'headerlines' , 1 ↩
    ) ;
plot(edad ,longitud,'o3')
hold on;
 x=fsolve(@sistemaBertal,[−1.5  0.5]);
Linf=calculaLinf(x);
t0=x(1);
r=x(2);
t=0:0.1:14;
y=Linf*(1−exp(−r*(t−t0)));
plot(t,y);
```

o utilizando el otro valor inicial:

```
[ edad longitud]=textread ( 'bertalan.csv' , '%f  %f' , 'headerlines' , 1 ↩
    ) ;
plot(edad ,longitud,'o3')
hold on;
 x=fsolve(@sistemaBertal,[−2  1]);
Linf=calculaLinf(x);
t0=x(1);
r=x(2);
t=0:0.1:14;
y=Linf*(1−exp(−r*(t−t0)));
plot(t,y);
```

La gráfica nos muestra que la segunda solución no muestra un buen ajuste a la gráfica.

Aproximación 2: minimizar la suma de cuadrados utilizando directamente la función de minimización sqp en Octave.

Para emplear esta función hemos de definir primero la función a minimizar:

```
function sc=sumCuad(x)
    Linf=x(1);
    t0=x(2);
    r=x(3) ;
```

```
    [ t y ]=textread ( 'bertalan.csv' , '%f %f' , 'headerlines' , 1 ) ;
    fit=Linf*(1-exp(-r*(t-t0)));
    sc=sum((y-fit).^2);
end %function
```

Nótese que esta función recibe como argumentos los valores de L_∞, t_0 y r. Para llamar a la función minimizadora sqp hemos de proporcionar un valor inicial $x_0 = [L_\infty;\ t_0;\ r]$:

```
>>> x0=[340 ;-1.5; 0.5]
x0 =
   340.00000
    -1.50000
     0.50000
>>> sqp(x0,@sumCuad)
ans =
   340.20984
    -1.74433
     0.33910
```