Lotka - Volterra model in biomatematics Predator-prey system solution

Profa. Hazem Álvarez R.

¿Qué es?

El modelo presa-depredador, también conocido como el modelo **Lotka-Volterra**, se ocupa de la interacción entre dos especies, donde *una* de ellas (*presa*) tiene abundante comida, y la *segunda* especie (*depredador*) tiene suministro de alimentos exclusivamente a la población de presas (Figueirero, 2014).

Descripción

Supongamos también que, durante el proceso, en un intervalo de tiempo t, el medio no debería cambiar favoreciendo a ninguna de las especies y que cualquier adaptación genética es lo suficientemente lenta.

Las variaciones están dadas por las siguientes ecuaciones:

$$\begin{cases} \begin{pmatrix} \text{variación} \\ \text{de número de} \\ \text{presas} \end{pmatrix} = \begin{pmatrix} \text{aumento} \\ \text{natural} \end{pmatrix} - \begin{pmatrix} \text{destrucción por} \\ \text{depredadores} \end{pmatrix} \\ \begin{pmatrix} \text{variación} \\ \text{de número de} \\ \text{depredadores} \end{pmatrix} = -\begin{pmatrix} \text{muerte en} \\ \text{ausencia} \\ \text{de presa} \end{pmatrix} + \begin{pmatrix} \text{aumento} \\ \text{causado por} \\ \text{alimentación} \\ \text{disponible} \end{pmatrix} \end{cases}$$

Donde:

$x = x(t)$: la densidad poblacional de las presas, y

$y = y(t)$: la densidad de población de depredadores de presas,

en cada momento t.

Modelo

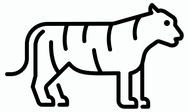
En pocas palabras, el modelo de Lotka-Volterra *supone que las presas crecen exponencialmente en ausencia de depredadores (modelo de Malthus)* y que la tasa de mortalidad de depredadores en ausencia de presa es proporcional a su población **y(t)** en cada momento **t** (muerte por falta de alimentos) (Puchuri, 2018).

La tasa de natalidad del depredador depende exclusivamente de este modelo en la cantidad de presas devoradas en cada encuentro.

Al modelar los posibles encuentros por el término bilineal **xy**, entonces el sistema presa-depredador, es:

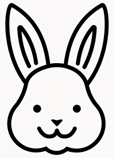
$$\begin{cases} \dfrac{dx}{dt} = ax - bxy \\ \dfrac{dy}{dt} = -cy + dxy \end{cases}$$

Recordemos



El modelo clásico de Lotka-Volterra se considera un hábitat en donde coexisten dos especies que interaccionan entre ellas.

- La especie $\mathbf{x(t)}$ se le llamará presa y tiene una fuente de alimentación por la que no compite;
- La otra especie $\mathbf{y(t)}$, a la que llamaremos depredador, tiene a $\mathbf{x(t)}$ en su dieta.
- De esta forma, $\mathbf{x(t)}$ representa el número de presas en el instante $\mathbf{t}$, mientras que $\mathbf{y(t)}$ indica la cantidad de depredadores en ese mismo momento.



Hipótesis

Para entender el problema depredador presa, fácilmente:

$$\frac{dx}{dt} = ax(t) : \text{Las presas, en ausencia de cualquier depredador.}$$

$$\frac{dx}{dt} = ax(t) - bx(t)y(t) : \text{Si hay presencia de depredador.}$$

$$\frac{dy}{dt} = -c\,y(t) : \text{Si no hay presas}$$

$$\frac{dy}{dt} = -cx(t) - dx(t)y(t) :$$

Prey $\boxed{\dfrac{dx}{dt}} = \alpha x - \beta xy$

Predator $\boxed{\dfrac{dy}{dt}} = \delta xy - \gamma y$

Resolviendo el modelo

Ecuaciones iniciales

$$\frac{dx}{dt} = \alpha x - \beta xy \qquad \frac{dy}{dt} = \delta xy - \gamma y$$

1. Igualamos a 0 para indicar un punto de equilibrio o estado estacionario

$$0 = \alpha x - \beta xy \qquad 0 = \delta xy - \gamma y$$

Es decir, $\text{Steady States } (\hat{x}, \hat{y})\text{: } (0,0)$

2. Resolvemos

$$0 = \alpha x - \beta xy$$
$$\beta xy = \alpha x$$
$$y = \frac{\alpha x}{\beta x}$$
$$\hat{y} = \frac{\alpha}{\beta}$$

$$0 = \delta xy - \gamma y$$
$$\gamma y = \delta xy$$
$$\frac{\gamma y}{\delta y} = x$$
$$\frac{\gamma}{\delta} = \hat{x}$$

Ejemplo en Python

```
% matplotlib en línea
import matplotlib.pyplot as plt
de importación aleatoria *
de importación de gruñón *
importaciones de sís

- parámetros del modelo
a = 0,7 ; b = 0,5 ; c = 0,3 ; e = 0,2
dt = 0,001 ; máximo de tiempo = 100

- tiempo inicial y poblaciones
t = 0 ; x = 1,0 ; y = 0,5

- listas vacías en las que almacenar tiempo y poblaciones
t_list=x-list = []; x-list = []; y-list = []

- listas de inicialización
t.list.append(t); x-list.append(x); y-list.append (y)

mientras que t - máximo tiempo::
    3 nuevos valores para t, x, y
    t = t +dt
    x = x (+(a * x - b * x * * y)* dt
    y = y +. ( c * y . e * * * *y)* dt

    - almacenar nuevos valores en las listas
    t.list.append (t)
    x-list.append(x)
    y-list.append (y)

- Tramar los resultados
p = plt.plot(t-list, x-list, 'r', t-list, y-list, 'g', linewidth = 22)
```

Recuperado de **Predator-Prey Relationships - The Lotka-Volterra Model en** https://aubreymoore.github.io/ALBI345F17/pdfs/Lotka-Volterra-Model.html#Predator-Prey-Relationships---The-Lotka-Volterra-Model

Fuentes

1. 99

1. Modelo de Lotka - Volterra en la biomatemática Solución de sistema depredador-presa en https://revistas.unjbg.edu.pe/index.php/cs/article/view/991/1116

2. Predator-Prey Model (Lotka-Volterra) en https://www.youtube.com/watch?v=Tc05IbqTsFM

3. Python Code for Predator-Prey Model
En https://www.youtube.com/watch?v=2f5aRTBmm10

Gracias