UNIVERSIDAD NACIONAL DE INGENIERIA
FACULTAD DE CIENCIAS

Tema:
Problema de LoktaVolterra

Apellidos: Moreno Vera
Nombres: Felipe Adrian
Código: 20120354I
Curso: Física Computacional
Codigo Curso: CC063

2016-II

Problema de Lokta-Volterra

Realiza la simulación de LoktaVolterra, para 6 poblaciones de depredador-presa.
Se sabe que las ecuaciones de presa-depredaror son:

Presa:

$$\frac{dx}{dt}=(a-by)x$$

Depredador:

$$\frac{dy}{dt}=(cx-d)y$$

de estas 2 se sabe: $\frac{dy}{dx}=-\frac{y}{x}\frac{(cx-d)}{(by-a)}$, resolviendo por variable separables:

$\frac{1}{y}(by-a)dy=-\frac{1}{x}(cx-d)dx$

se tiene:

$$(by-a)dln(y)+(cx-d)dln(x)=0$$

Se construye la ecuación:

Solución general:

Constante de integración:

$$V(x,y)=-cx+d\ln(x)-by+a\ln(y)$$

Ahora haciendo un cambio de variable:

$u_1=\frac{c}{d}x$, $u_2=\frac{b}{a}y$, $d\tau=a\,dt$ y $\alpha=\frac{d}{a}$

Se tiene:

Presa:

$$\frac{dx}{dt}=(a-by)x$$

Depredador:

$$\frac{du_2}{d\tau}=\alpha(u_1-1)u_2$$

de estas 2 se sabe: $\frac{du_2}{du_1}=\alpha\frac{u_2}{u_1}\frac{(u_1-1)}{(1-u_2)}$

Solución general:

Constante de integración:

$$V(u_1,u_2)=-\alpha u_1+\alpha\ln(u_1)-u_2+\ln(u_2)$$

Sistema de ecuaciones LoktaVolterra

Para la solución del sistema, se tiene una matriz A, y vectores de la siguiente forma:

$\dot{x}_i = x_i \left(r_i + \sum_{j=1}^{n} a_{ij} x_j \right)$, donde: a_{ij} son los componentes de la matriz A.

Ejemplo de 2 dimensiones:

Sistema de ecuaciones presa-depredador:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} x & y \end{pmatrix} \left(\begin{pmatrix} a \\ -d \end{pmatrix} + \begin{pmatrix} 0 & -b \\ c & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \right)$$

esto nos da las ecuaciones:

Presa:

$$\frac{dx}{dt} = (a - by)\, x$$

Depredador:

$$\frac{dy}{dt} = (cx - d)\, y$$

Entonces llevemóslo a nuestra problemática con 6 poblaciones (3presas y 3 depredadores)

$$\dot{x}_i = x_i \left(r_i + \sum_{j=1}^{n} a_{ij} x_j \right)$$

Es equivalente a:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} \left(r + \begin{pmatrix} 0 & 0 & 0 & -20 & -30 & -5 \\ 0 & 0 & 0 & -1 & -3 & -7 \\ 0 & 0 & 0 & -4 & -10 & -20 \\ 20 & 30 & 35 & 0 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 & 0 \\ 7 & 8 & 20 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} \right)$$

Donde $r = \begin{pmatrix} a' \\ d' \end{pmatrix}$ donde a y d son vectores, donde sus componentes son:

$a_i = \sum_{j=1}^{n} b_{ij} x_j(0)$ y $d_i = \sum_{j=n+1}^{m} c_{ij} x_j(0)$, donde “n” es el número de presas y “m” es el número de depredadores, donde los x_i son los valores iniciales de cada componente.

Cálculo de los valores iniciales:

Se sabe que el sistema solución de LoktaVolterra es:

$$X(t) = \sum_{i=1}^{n} c_i e^{\lambda_i t} v_i$$

Donde v_i son los autovectores, λ_i son los autovalores y c_i son los coeficientes para la solución. (combinación lineal de las soluciones).

Solución:

Nos dan de dato la matriz A (ya definida arriba) y los coeficientes c_i .

Entonces nuestros X(0) son:

$$X(0)=\sum_{i=1}^{n} c_i v_i$$

Calculando los autovectores de A usando Octave, con la función eig(A) se obtiene:

$$\begin{pmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\\ \lambda_4\\ \lambda_5\\ \lambda_6\end{pmatrix}=\begin{pmatrix}0.00000-33.62930\,i\\ 0.00000+33.62930\,i\\ -0.00000-7.69575\,i\\ -0.00000+7.69575\,i\\ -0.39310-0.00000\,i\\ 0.39310-0.00000\,i\end{pmatrix}$$

Esto es igual a un vector:

$$\begin{pmatrix}x_1(0)\\ x_2(0)\\ x_3(0)\\ x_4(0)\\ x_5(0)\\ x_6(0)\end{pmatrix}=\begin{pmatrix}0.61968\\ 0.33702\\ -0.52611\\ 0.24196\\ 3.26462\\ 1.28524\end{pmatrix}$$

Entonces calculando nuestro r, vendría a ser:

Vector de presas:

$$a_i=\sum_{j=1}^{3} b_{ij} x_j(0) \quad \text{, entonces : } \quad a=\begin{pmatrix}-20 & -30 & -5\\ -1 & -3 & -7\\ -4 & -10 & -20\end{pmatrix}\begin{pmatrix}x_1(0)\\ x_2(0)\\ x_3(0)\end{pmatrix}$$

$$\text{resulta: } \quad a=\begin{pmatrix}-19.8737\\ 2.0520\\ 4.6733\end{pmatrix}$$

Vector de depredadores:

$$d_i=\sum_{j=n+1}^{m} c_{ij} x_j(0) \quad \text{, entonces: } \quad d=\begin{pmatrix}20 & 30 & 35\\ 3 & 3 & 3\\ 7 & 8 & 20\end{pmatrix}\begin{pmatrix}x_4(0)\\ x_5(0)\\ x_6(0)\end{pmatrix}$$

$$\text{resulta: } \quad d=\begin{pmatrix}147.761\\ 14.375\\ 53.515\end{pmatrix}$$

Entonces nuestro vector r es igual a:

$$r=\begin{pmatrix}-19.8737\\ 2.0520\\ 4.6733\\ -147.761\\ -14.375\\ -53.515\end{pmatrix}$$

Entonces nuestro sistema a resolver queda de la forma:

$$\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\\ \dot{x}_5\\ \dot{x}_6\end{pmatrix}=\begin{pmatrix}x_1\\ x_2\\ x_3\\ x_4\\ x_5\\ x_6\end{pmatrix}\left(\begin{pmatrix}-19.8737\\ 2.0520\\ 4.6733\\ -147.761\\ -14.375\\ -53.515\end{pmatrix}+\begin{pmatrix}0 & 0 & 0 & -20 & -30 & -5\\ 0 & 0 & 0 & -1 & -3 & -7\\ 0 & 0 & 0 & -4 & -10 & -20\\ 20 & 30 & 35 & 0 & 0 & 0\\ 3 & 3 & 3 & 0 & 0 & 0\\ 7 & 8 & 20 & 0 & 0 & 0\end{pmatrix}\begin{pmatrix}x_1\\ x_2\\ x_3\\ x_4\\ x_5\\ x_6\end{pmatrix}\right)$$

Usando el método de euler para aproximar la solución del sistema de ecuaciones de grado 1 con valores iniciales:

$$x_{i+1}(t)=x_i(t)+\dot{x}_i(t)\,dt$$

Se hace la aproximación para calcular el valor de los X(t) usando Octave.
Y usando la matriz anterior se tiene:

$$x_{i+1}(t)=x_i(t)+x_i(t)\left(r_i+\sum_{j=1}^{n} a_{ij}x_j(t)\right)dt$$

Como ecuación para calcular cada valor por variación del tiempo.

Usando Octave:
Inicializamos valores:

```
A=[ 0  0  0 -20 -30 -5;
     0  0  0 -1  -3  -7;
    0  0  0 -4  -10 -20;
    20 30 35 0  0   0;
    3  3  3  0  0   0;
    7  8  20 0  0   0];

b = [-20 -30 -5;
      -1  -3 -7;
      -4 -10 -20];

c = [20 30 35;
     3  3  3;
     7  8  20];
```

Autovalores y Autovectores:

```
[autovectores,v]=eig(A);
# devuelve los autovectores de manera vertical, y los autovalores en matriz diagonal
autovectores
autovalores = [];
k=1;
len1 = length(v);
len2 = length(v(1,:));
for i=1:len1
  for j=1:len2
    if i==j
      autovalores(k) = v(i,j);
      k++;
    end
  end
end

autovalores=autovalores'
```

Valores Iniciales:

```
x0 = zeros(lenu,1);
lenu = length(autovectores);
lamb = [3,3,1,1,5,0.1];
```

Solucion Analitica

```
t=0;
for i=1:lenu
  x0 = x0 + lamb(i)*exp(autovalores(i)*t)*autovectores(:,i);
endfor
```

Solucion Numerica

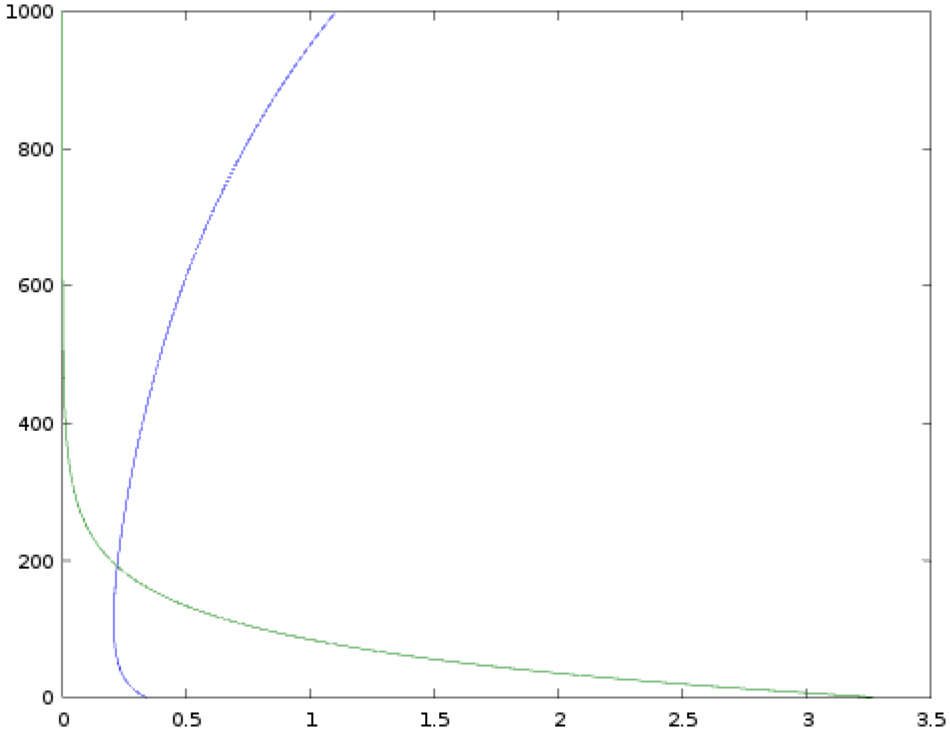
```
iters=10;
```

Calculamos el vector r:

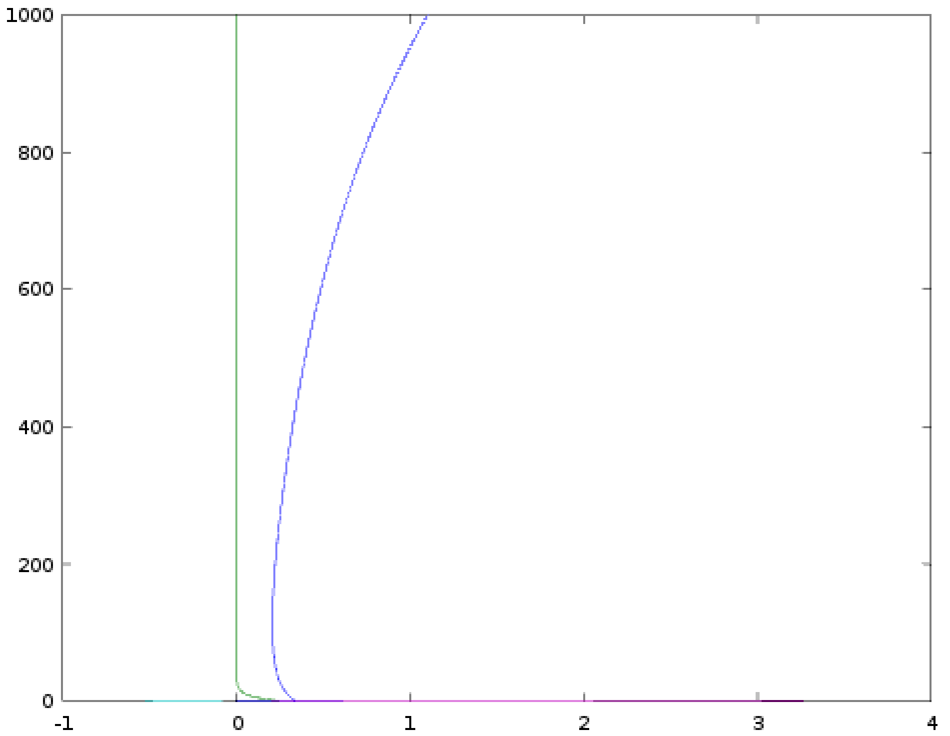
```
a = b*x0(1:lenu/2);
d = c*x0(lenu/2+1:6);
r = a;
r(4:6)=d
x0
xt = zeros(lenu,iters); # lenu columnas y iters filas
xt(:,1)=x0;
```

Solución Numérica I: (loktaVolterra_numerico_Real.m):

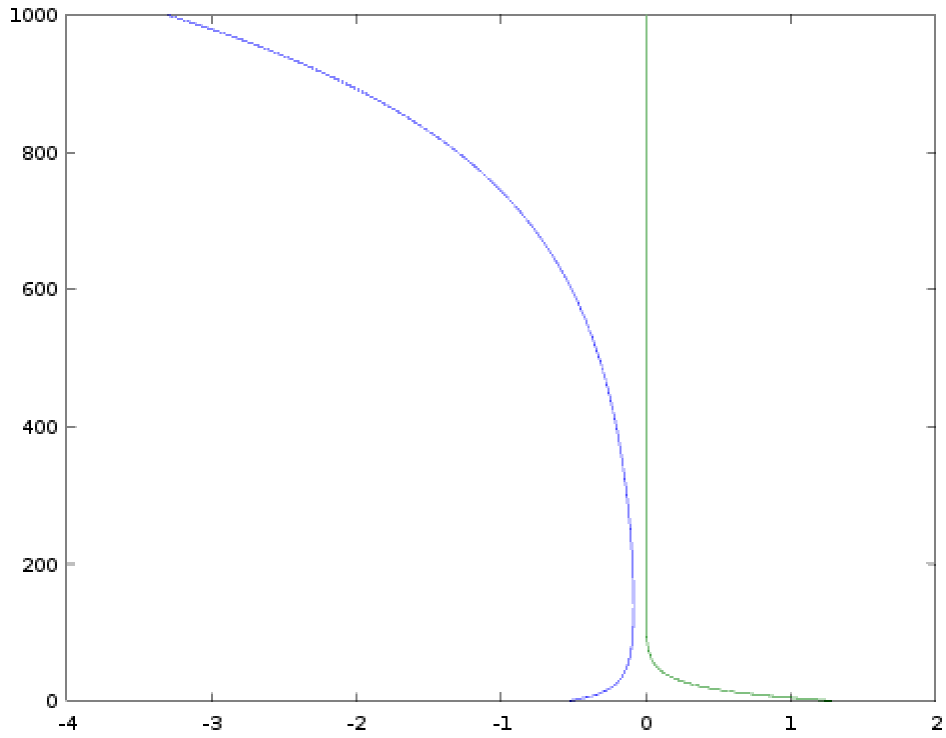
Problación 1 y 3:



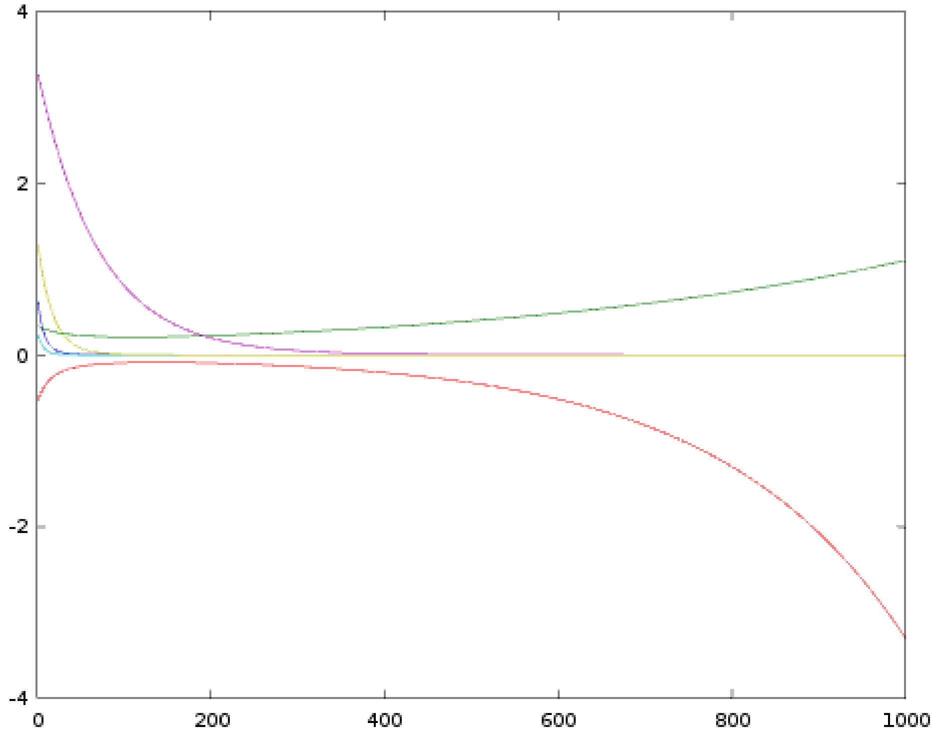
Población 2 y 4:



Población 3 y 6:



Interacción de las 6 poblaciones (presa vs t y depredador vs t):



Metodo de Euler para la aproximacion de las poblaciones:

```
dt=0.0001;
for i=2:iters
  for j=1:lenu
    xt(j,i)= xt(j,i-1) + xt(j,i-1)*(r(j)+A(j,:)*xt(:,i-1))*dt;
  end
end
```

Resolviendo y viendo los resultados, obtenemos lo siguiente:

```
Columns 1 through 5:

   6.1968e-01    6.1169e-01    6.0377e-01    5.9595e-01    5.8821e-01

Columns 6 through 10:

   5.8056e-01    5.7299e-01    5.6550e-01    5.5810e-01    5.5078e-01

Columns 11 through 15:

   5.4353e-01    5.3638e-01    5.2929e-01    5.2229e-01    5.1537e-01

Columns 16 through 20:

   5.0852e-01    5.0176e-01    4.9506e-01    4.8844e-01    4.8190e-01

Columns 21 through 25:

   4.7543e-01    4.6904e-01    4.6271e-01    4.5646e-01    4.5028e-01

Columns 26 through 30:

   4.4417e-01    4.3813e-01    4.3216e-01    4.2626e-01    4.2042e-01

Columns 31 through 35:

   4.1465e-01    4.0895e-01    4.0331e-01    3.9774e-01    3.9224e-01

Columns 36 through 40:

   3.8680e-01    3.8142e-01    3.7610e-01    3.7085e-01    3.6565e-01

Columns 41 through 45:

   3.6052e-01    3.5545e-01    3.5044e-01    3.4549e-01    3.4059e-01
```

```
Columns 461 through 465:

   2.6702e-20    1.3821e-20    7.0596e-21    3.5574e-21    1.7678e-21

Columns 466 through 470:

   8.6595e-22    4.1795e-22    1.9868e-22    9.2974e-23    4.2810e-23

Columns 471 through 475:

   1.9386e-23    8.6285e-24    3.7728e-24    1.6197e-24    6.8222e-25

Columns 476 through 480:

   2.8177e-25    1.1404e-25    4.5189e-26    1.7520e-26    6.6405e-27

Columns 481 through 485:

   2.4584e-27    8.8813e-28    3.1279e-28    1.0728e-28    3.5794e-29

Columns 486 through 490:

   1.1603e-29    3.6497e-30    1.1124e-30    3.2802e-31    9.3431e-32

Columns 491 through 495:

   2.5660e-32    6.7817e-33    1.7212e-33    4.1850e-34    9.7232e-35
```

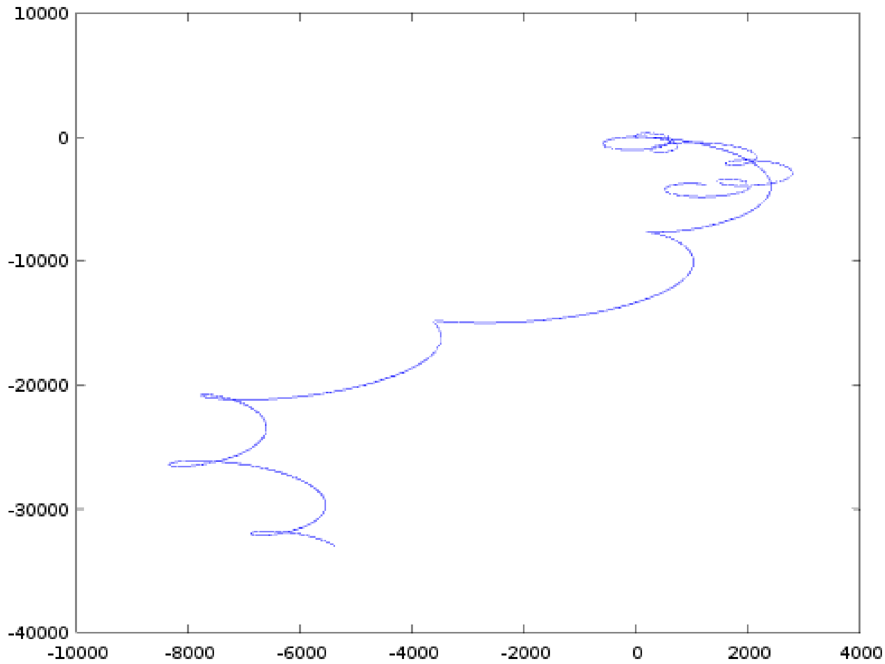
```
Columns 571 through 575:

   8.2016e-78   -1.0805e-77    1.4600e-77   -2.0229e-77    2.8732e-77

Columns 576 through 580:

  -4.1823e-77    6.2377e-77   -9.5300e-77    1.4912e-76   -2.3890e-76

Columns 581 through 585:

   3.9183e-76   -6.5775e-76    1.1299e-75   -1.9857e-75    3.5698e-75

Columns 586 through 590:

  -6.5636e-75    1.2340e-74   -2.3720e-74    4.6608e-74   -9.3599e-74
```

```
Columns 651 through 655:

   3.0369e-39  -1.9491e-38   1.2723e-37  -8.4453e-37   5.7008e-36

Columns 656 through 660:

  -3.9132e-35   2.7313e-34  -1.9385e-33   1.3989e-32  -1.0264e-31

Columns 661 through 665:

   7.6566e-31  -5.8069e-30   4.4774e-29  -3.5097e-28   2.7967e-27

Columns 666 through 670:

  -2.2655e-26   1.8655e-25  -1.5615e-24   1.3286e-23  -1.1490e-22

Columns 671 through 675:

   1.0100e-21  -9.0232e-21   8.1934e-20  -7.5615e-19   7.0922e-18

Columns 676 through 680:

  -6.7603e-17   6.5488e-16  -6.4469e-15   6.4496e-14  -6.5567e-13

Columns 681 through 685:

   6.7734e-12  -7.1102e-11   7.5842e-10  -8.2200e-09   9.0524e-08

Columns 686 through 690:

  -1.0129e-06   1.1516e-05  -1.3302e-04   1.5612e-03  -1.8615e-02

Columns 691 through 695:

   2.2550e-01  -2.7752e+00   3.4714e+01  -4.3854e+02   6.0419e+03
```

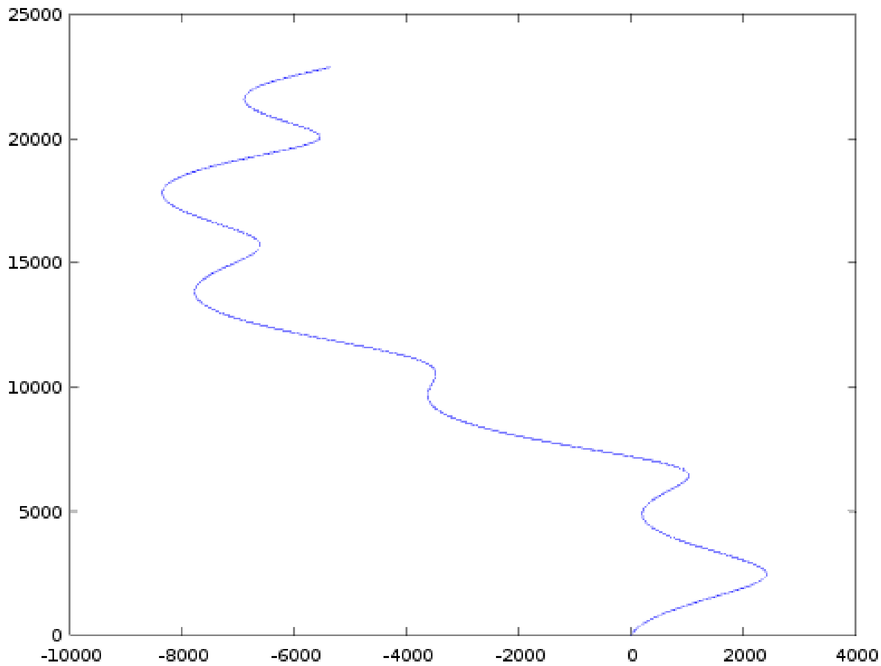
```
Columns 686 through 690:

  -1.0129e-06   1.1516e-05  -1.3302e-04   1.5612e-03  -1.8615e-02

Columns 691 through 695:

   2.2550e-01  -2.7752e+00   3.4714e+01  -4.3854e+02   6.0419e+03

Columns 696 through 700:
  -6.7603e-17   6.5488e-16  -6.4469e-15   6.4496e-14  -6.5567e-13

Columns 681 through 685:

   6.7734e-12  -7.1102e-11   7.5842e-10  -8.2200e-09   9.0524e-08

Columns 686 through 690:

  -1.0129e-06   1.1516e-05  -1.3302e-04   1.5612e-03  -1.8615e-02

Columns 691 through 695:

   2.2550e-01  -2.7752e+00   3.4714e+01  -4.3854e+02   6.0419e+03

Columns 696 through 700:

  -6.4702e+03   1.6640e+05   5.2926e+07   5.5971e+12   6.2639e+22

Columns 701 through 705:

   7.8467e+42   1.2314e+83  3.0326e+163          Inf          NaN
```

Con estas imágenes de resultados, se ve que los resultados de las poblaciones decrecen y crecen hasta cierto punto y vuelven a decrecer … pero al momento de crecer de nuevo diverge.

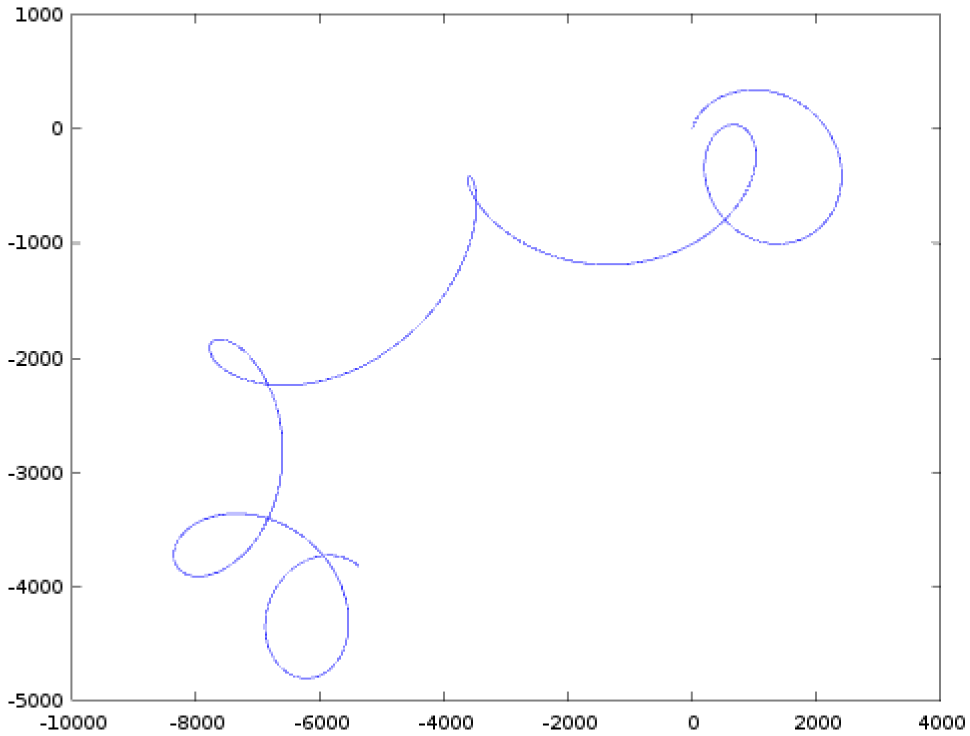
Solución Analítica (loktaVolterra_analitico.m):
Se presenta las gráficas de poblaciones:
Población 1 y 4:



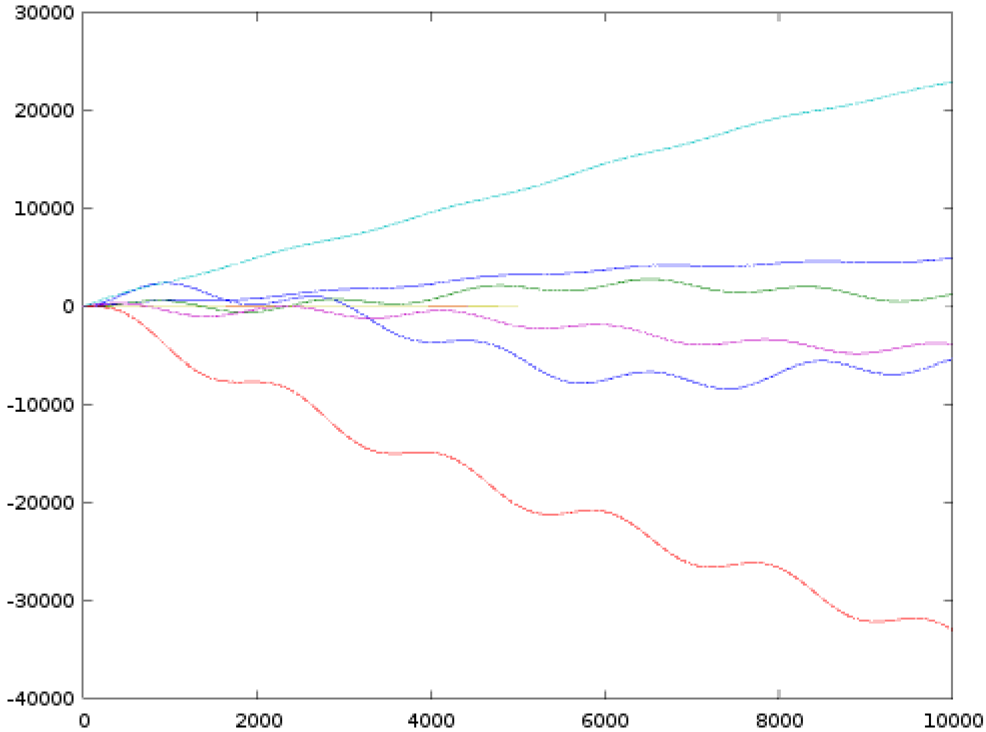
Población 2 y 5:



Población 3 y 6:



Interacción de las 6 poblaciones:



Se observa la divergencia.

Explicación de la divergencia:
Se ve que los autovalores de la matriz A, también llamada estabilidad tiene autovalores imaginarios y reales, de los cuales 4 son imaginarios y 2 reales.
Por tener un autovalor real positivo y otro negativo, el sistema se hace inestable.
Es por eso que el sistema (solución real) diverge.

Solución Numérica II:
Cambio de variable (loktaVolterra_numerico_Ideal.m):

Es decir, probando con el vector

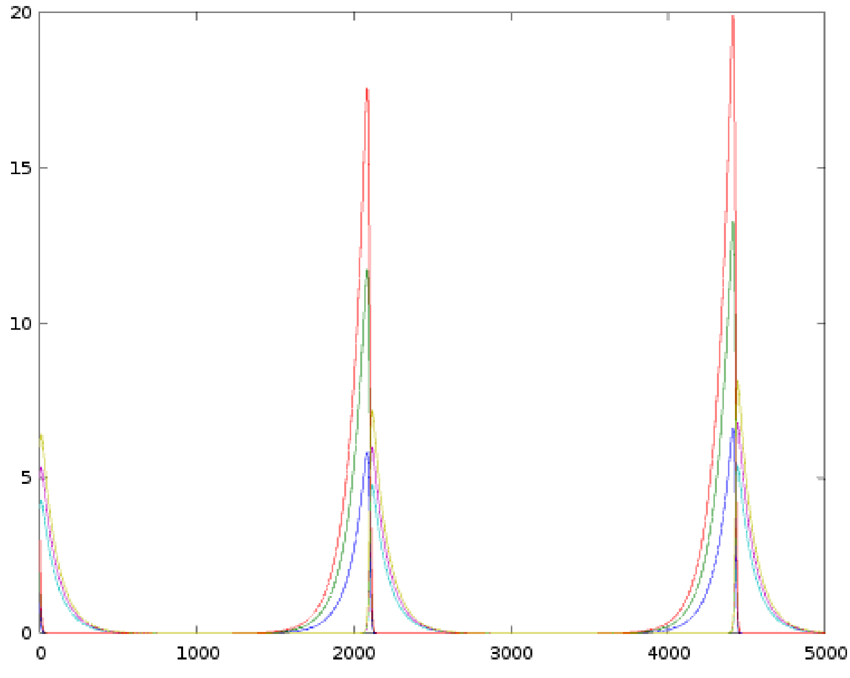
$$r=\begin{pmatrix}1\\1\\1\\-1\\-1\\-1\end{pmatrix}$$

y los mismos valores iniciales, con $\alpha=0.5$.
Es decir, la nueva ecuación a tomar será:

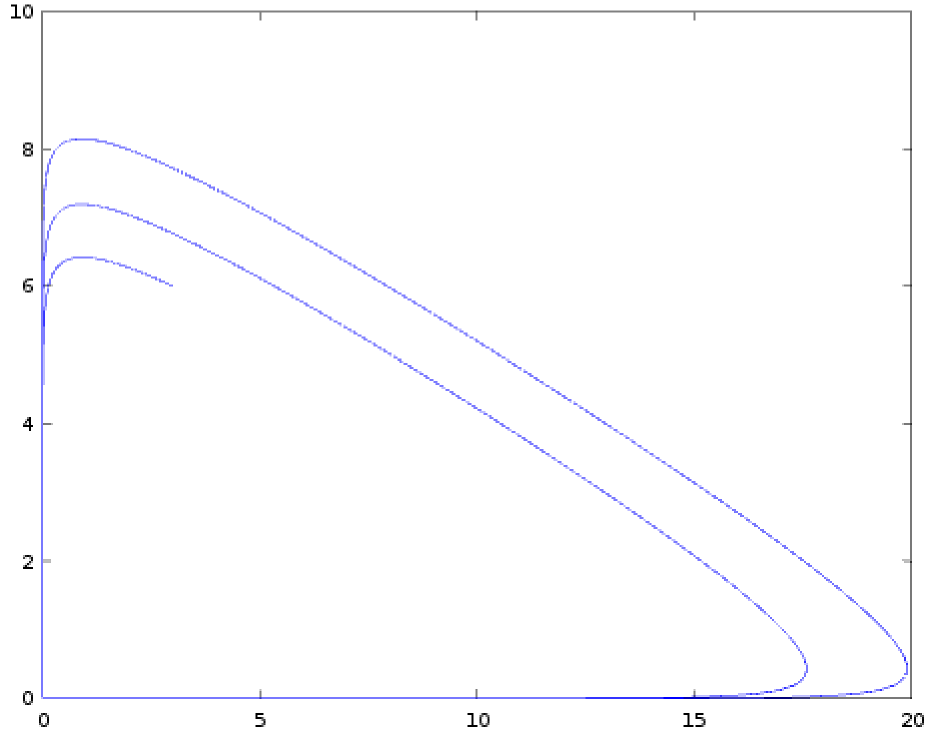
$$\begin{pmatrix}\dot{x}_1\\\dot{x}_2\\\dot{x}_3\\\dot{x}_4\\\dot{x}_5\\\dot{x}_6\end{pmatrix}=\begin{pmatrix}x_1\\x_2\\x_3\\x_4\\x_5\\x_6\end{pmatrix}\left(\begin{pmatrix}1\\1\\1\\-1\\-1\\-1\end{pmatrix}+\begin{pmatrix}0&0&0&-1&-1&-1\\0&0&0&-1&-1&-1\\0&0&0&-1&-1&-1\\\alpha&\alpha&\alpha&0&0&0\\\alpha&\alpha&\alpha&0&0&0\\\alpha&\alpha&\alpha&0&0&0\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\\x_4\\x_5\\x_6\end{pmatrix}\right)$$

Se observa que los valores no divergen tan rápido como de la anterior manera.

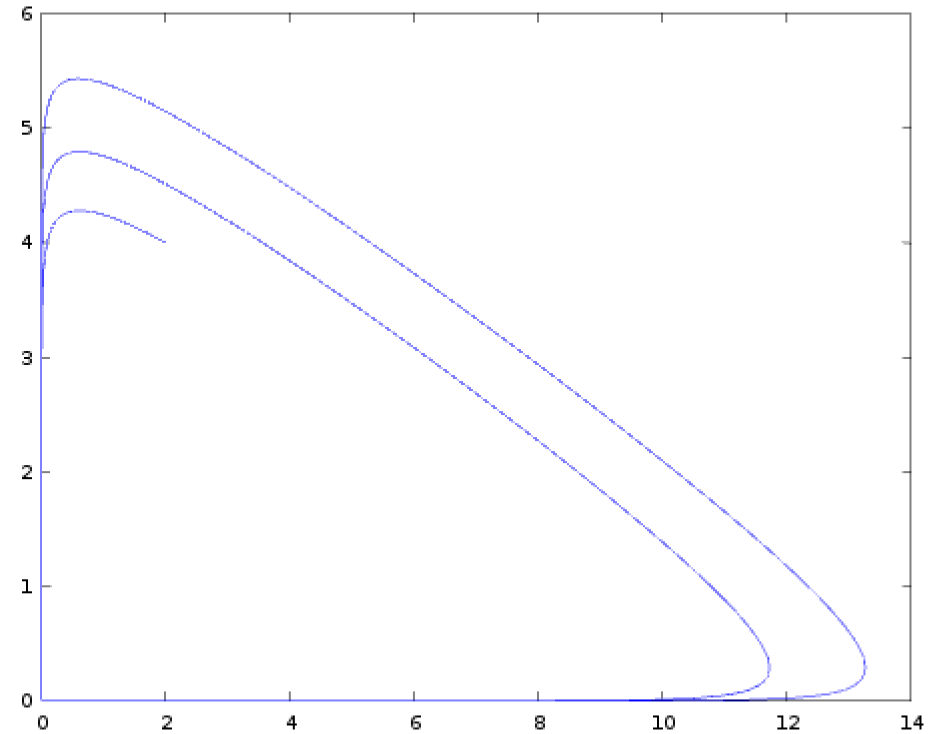
Solución general: (Presas vs t y Depredadores vs t)



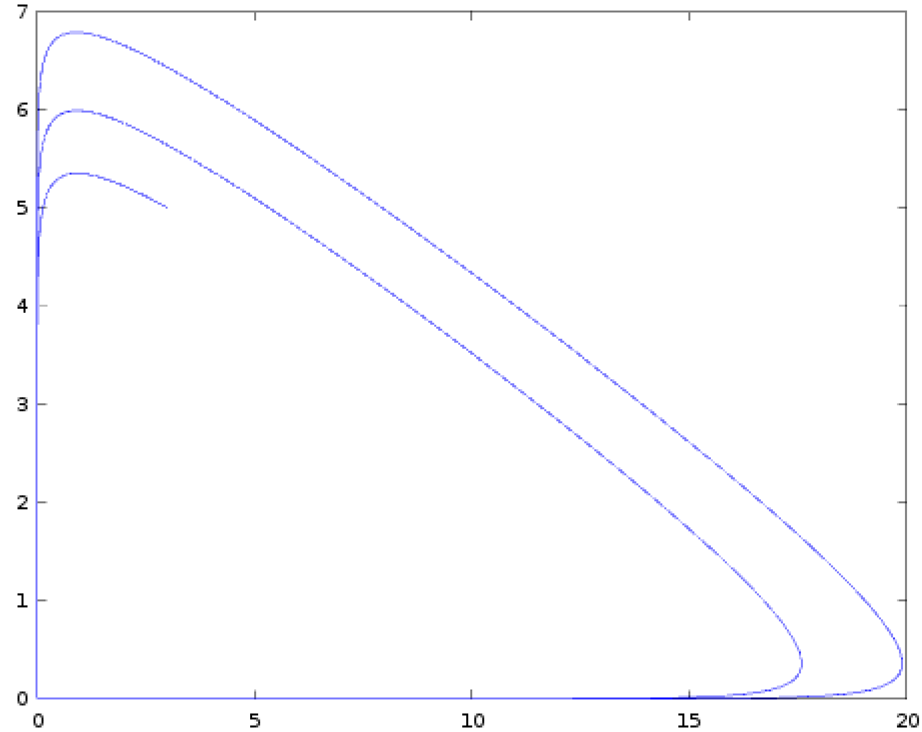
Población 1 y 4:



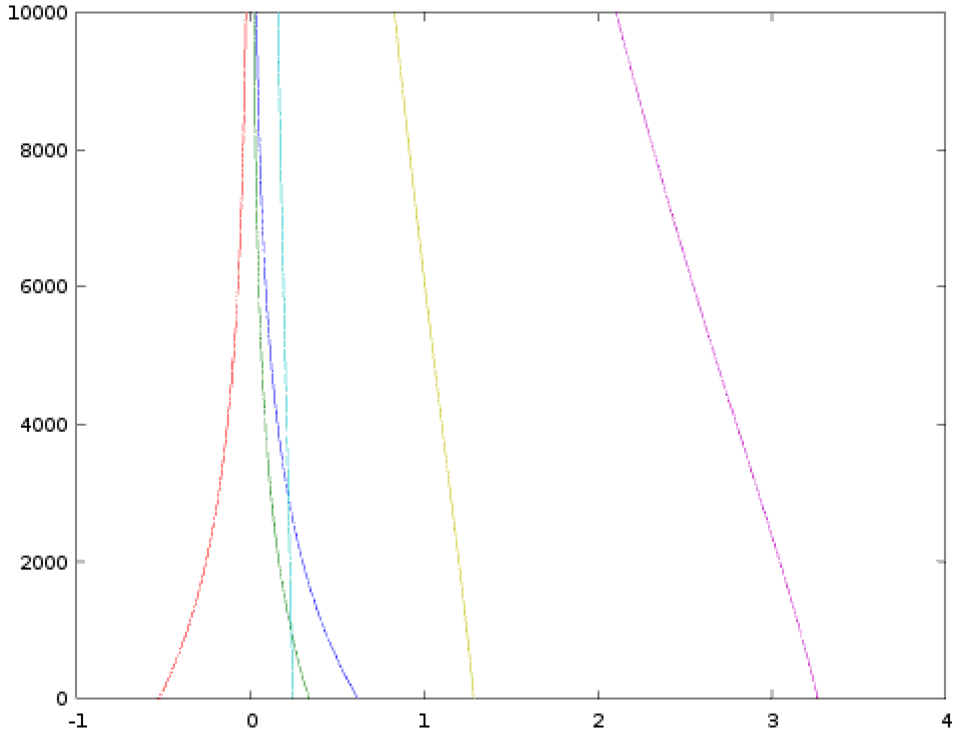
Población 2 y 5:



Población 3 y 6:



Interacción entre las 6 poblaciones:



Se observa cierta estabilidad luego del cambio de variable.

Analisando par en par la población (loktaVolterra_pares.m):

Escogemos una presa y un depredador y aplicamos la misma técnica, pero esta vez en un sistema de 2 dimensiones, es decir:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \left(\begin{pmatrix} 1 \\ -1 \end{pmatrix} + \begin{pmatrix} 0 & -1 \\ \alpha & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right)$$

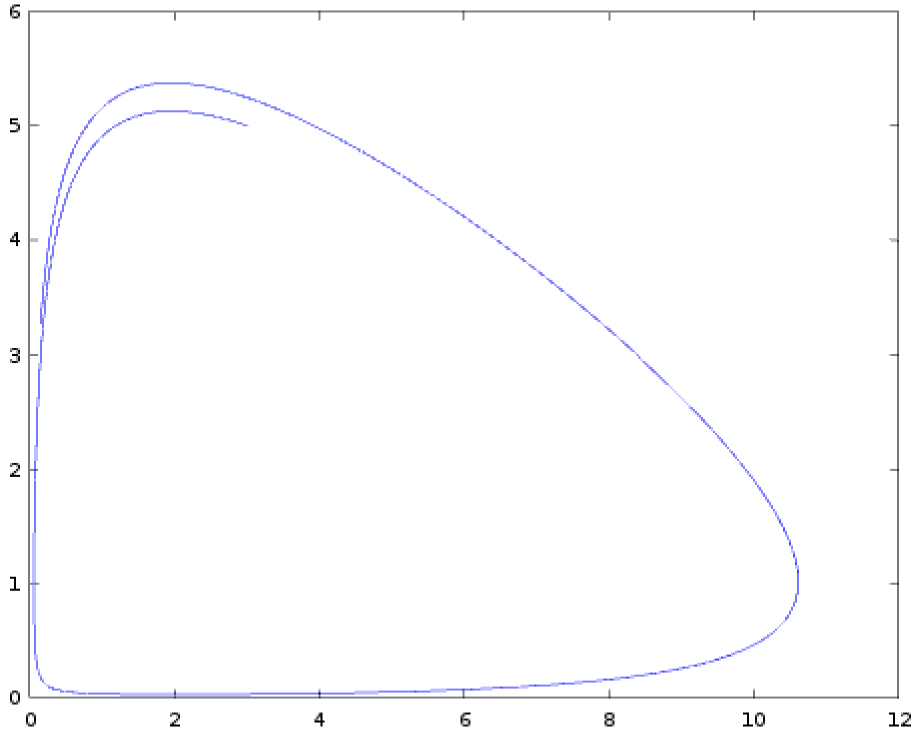
Por lo que se analiza con las condiciones iniciales similares al anterior, pero esta vez con $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix}$ y tomamos de par en par.

El script loktaVolterra_pares.m es el que ejecuta este método cogiendo por pares, el cual nos muestra (a diferencia de las anteriores gráficas donde obviamente la interacción entre 6 poblaciones divergen) que las poblaciones cumplen los condicionamientos de Lokta-Volterra.

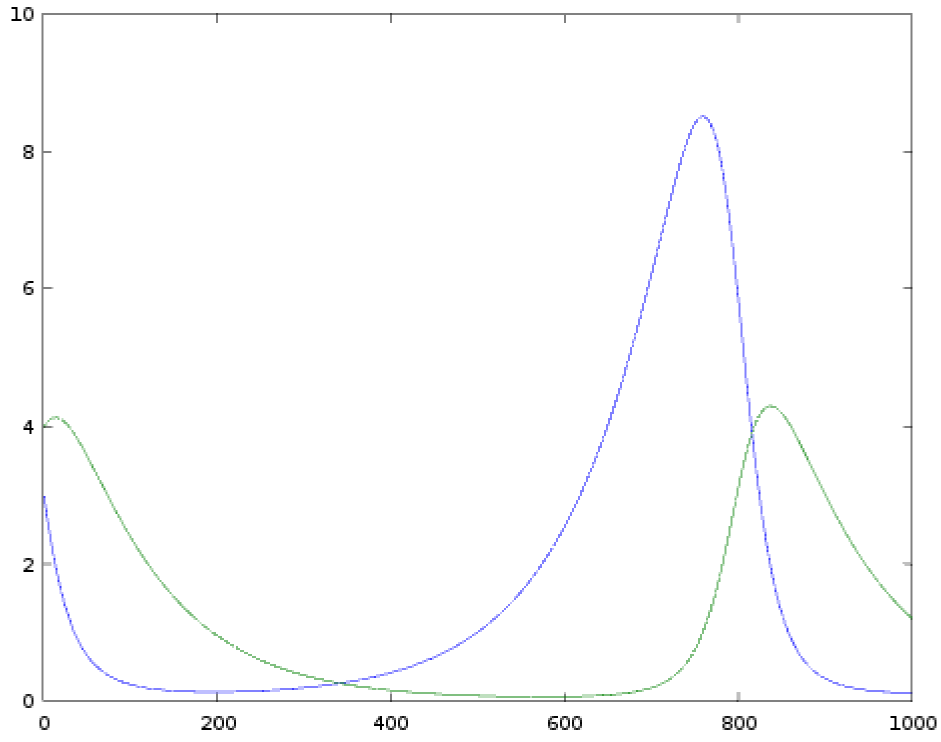
Ejemplo:

Interacción de l apoblación 1 (presa) y la 3 (depredador)

Gráfica Presa vs Depredador:



Gráfica Presa vs t y Depredador vs t:



Podemos apreciar que la convergencia entre 2 poblaciones es mucho más estable que la interacción entre 3 o más especies.

<u>Resumen:</u>

Se tienen 4 scripts:
loktaVolterra_analitico.m:
Presenta resultados mediante la solución analítica del sistema loktaVolterra para 6 poblaciones interactuantes al mismo tiempo, usando:

$$X(t)=\sum_{i=1}^{n} c_i e^{\lambda_i t} v_i$$

loktaVolterra_numerico_Real.m:
Presenta resultados mediante la solución numérica del sistema loktaVolterra para 6 poblaciones interactuantes al mismo tiempo usando el método de Euler, usando:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} \left(\begin{pmatrix} -19.8737 \\ 2.0520 \\ 4.6733 \\ -147.761 \\ -14.375 \\ -53.515 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & -20 & -30 & -5 \\ 0 & 0 & 0 & -1 & -3 & -7 \\ 0 & 0 & 0 & -4 & -10 & -20 \\ 20 & 30 & 35 & 0 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 & 0 \\ 7 & 8 & 20 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} \right)$$

loktaVolterra_numerico_Ideal:
Presenta resultados mediante la solución numérica del sistema loktaVolterra para 6 poblaciones interactuantes al mismo tiempo usando el método de Euler, haciendo el cambio de variable de x, y a u1 y u2 usando:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} \left(\begin{pmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & -1 & -1 & -1 \\ 0 & 0 & 0 & -1 & -1 & -1 \\ 0 & 0 & 0 & -1 & -1 & -1 \\ \alpha & \alpha & \alpha & 0 & 0 & 0 \\ \alpha & \alpha & \alpha & 0 & 0 & 0 \\ \alpha & \alpha & \alpha & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} \right)$$

loktaVolterra_pares.m:
Presenta resultados mediante la solución numérica del sistema loktaVolterra tomando de par en par presa-depredador, se sabe que del sistema de 6 variables, 3 son presas y 3 son depredadores. Entonces este script esta diseñado para analizar cada presa con cada depredador mediante un sistema de 2 variables, de la forma:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \left(\begin{pmatrix} 1 \\ -1 \end{pmatrix} + \begin{pmatrix} 0 & -1 \\ \alpha & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right)$$

Ejecución:
En consola ejecutar:

octave loktaVolterra_pares.m o también entrar al shell de octave y ejecutar el nombre del programa.

Referencias:

http://mathfaculty.fullerton.edu/mathews/n2003/lotkavoltera/Lotka-VolterraMod/Links/Lotka-VolterraMod_lnk_3.html
https://en.wikipedia.org/wiki/Competitive_Lotka%E2%80%93Volterra_equations
https://en.wikipedia.org/wiki/Lotka%E2%80%93Volterra_equations