Universidad Rey Juan Carlos

Práctica número 1. Curso 2012-2013

Las prácticas propuestas aquí están realizadas con la ayuda del programa de simulación Matlab. Las prácticas también se podrán realizar con el programa libre Octave disponible para los sistemas operativos Linux y Windows.

Las prácticas consisten en la elaboración de ficheros de comando Matlab para la resolución de problemas. Los “scripts” tienen que ir acompañados por una memoria explicativa y por las figuras correspondientes a cada problema.

En esta práctica se van a estudiar varias propiedades de las aplicaciones discretas.

1. Iteraciones de aplicaciones discretas

Proponemos estudiar aplicaciones discretas con Matlab de la forma:

$$\mathbf{x}_{n+1} = f(\mathbf{x}_n, \beta), \tag{1}$$

donde $\mathbf{x}$ es un vector de dimensión k. El dominio de definición de $\mathbf{x}$ depende de la función f. β representa un parámetro o un conjunto de parámetros del sistema. Para empezar elaboramos un programa en Matlab para iterar la función f, el pseudocódigo es el siguiente:

```
Función: Trayectoria(condiciones iniciales)

% Definición de parámetros y reserva de memoria
número de iteraciones=100
parámetro de la función=1
variable=vector vacío de dimensiones [k x número de iteraciones]

% Bucle de iteración
Para (contador=1) hasta (número de iteraciones-1)
variable(contador+1)=función mapa(variable(contador), parámetro)

Fin
```

Este programa toma como entrada un vector de valores iniciales y devuelve la trayectoria. Para que esté completo debemos escribir otra función con el mapa a iterar. Resulta interesante poder utilizar diferentes parámetros, por lo que es conveniente introducir el parámetro como una variable más.

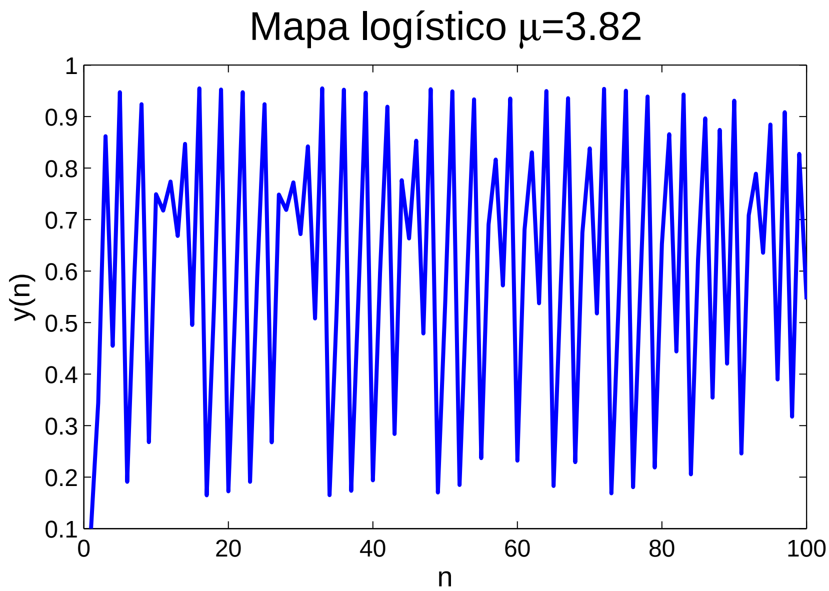

Figura 1: Ejemplo de trayectoria caótica en el mapa logístico con $\mu = 3{,}82$ y $x_0 = 0{,}1$.

Ejercicio 1 : Programar la función Trayectoria y cada uno de los mapas que se proponen a continuación. Obtener trayectorias con condiciones iniciales en [0,1] y representarlas del mismo modo que se hace en la figura 1.

1. Mapa triángulo (Tent Map) con los parámetros $\mu = 0{,}5$, $\mu = 1$ y $\mu = 1{,}5$:

$$f(x) = \begin{cases} \mu x & 0 \leq x \leq 0{,}5, \\ \mu(1-x) & 0{,}5 \leq x \leq 1, \end{cases} \tag{2}$$

2. Mapa logístico con los parámetros $\mu = 0$, $\mu = 2$,$\mu = 3{,}82$:

$$x_{n+1} = \mu(1 - x_n)x_n, \tag{3}$$

3. Mapa gaussiano con los parámetros $\alpha = 4$ y $\beta = -1$, $\beta = -0{,}5$, $\beta = 0$:

$$x_{n+1} = e^{-\alpha x_n^2} + \beta \tag{4}$$

2. Representación de las iteraciones

Para entender mejor el proceso de iterar mapas unidimensionales vamos a representar en una figura las iteraciones superpuestas al mapa. Primero vamos a representar la aplicación junto a la primera bisectriz para ayudarnos a visualizar las sucesivas iteraciones. Sobre este dibujo representaremos las iteraciones sucesivas gracias a la función `stairs` de Matlab, tal y como se muestra en la figura 2.

Ejercicio 2 : Realizar las siguientes tareas con el mapa logístico:

1. Representar en una misma figura las iteraciones en el mapa y en función del tiempo (Ayuda: usar función `subplot` de Matlab).

2. Representar una trayectoria para $\mu = 0{,}5$, $\mu = 1$ y $\mu = 1{,}5$. La condición inicial debe encontrarse en [0,1].

3. Mapas bidimensionales

Ahora veremos cómo simular y representar órbitas de mapas de dimensión dos. Los programas para la integración son idénticos al caso unidemensioal, teniendo cuidado al poner un vector ahora en la función f. Cogemos por ejemplo el mapa de Lozi:

$$\begin{aligned} x_{n+1} &= 1 - \alpha|x_n| + y_n \\ y_{n+1} &= \beta x_n \end{aligned} \tag{5}$$

La función f del mapa recibe un vector y devuelve otro vector. Podemos representar las trayectorias por separado en dos ventanas separadas pero también podemos representar la órbita con puntos discretos en un plano. Las coordenadas de cada punto corresponden a una iteración del mapa. Aunque se pierde el aspecto de la dinámica temporal podemos fácilmente localizar órbitas periódicas y caóticas si las hay. Con la ayuda de Matlab representar una órbita del mapa anterior para los parámetros $\alpha = 1{,}4$; $\beta = 0{,}3$ usando la función `plot` y sus opciones (como 'MarkerSize') a fin de obtener resultados como los expuestos en la figura 3.

Repetir lo anterior con el mapa de Hénon:

$$\begin{aligned} x_{n+1} &= 1 + y_n - \alpha x_n^2 \\ y_{n+1} &= \beta x_n \end{aligned} \tag{6}$$

con los parámetros: $\beta = 0{,}4$, $\alpha = 1$ y $\alpha = 1{,}2$

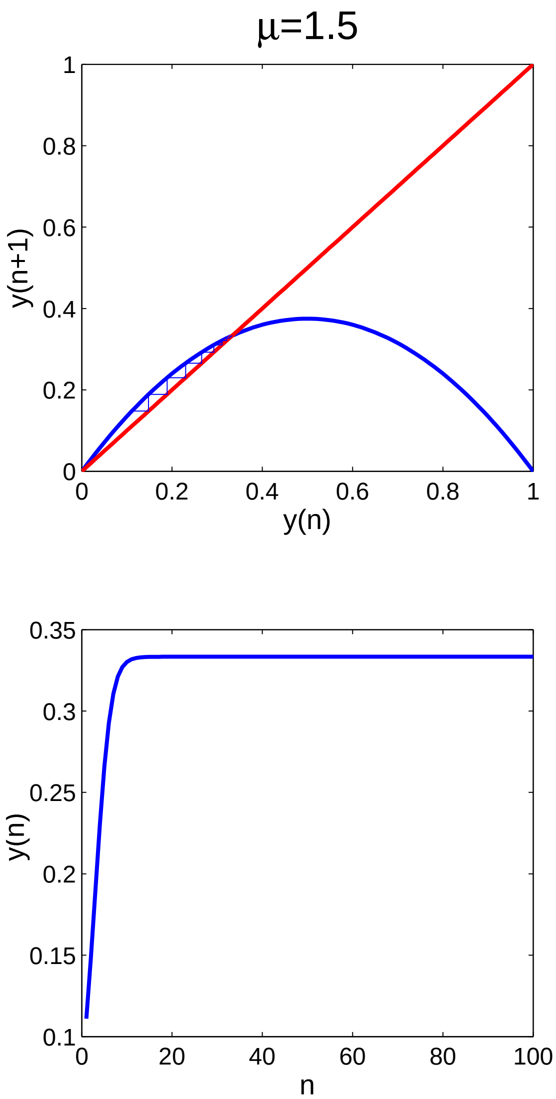

Figura 2: Ejemplo de mapa tela de araña para la aplicación logística.

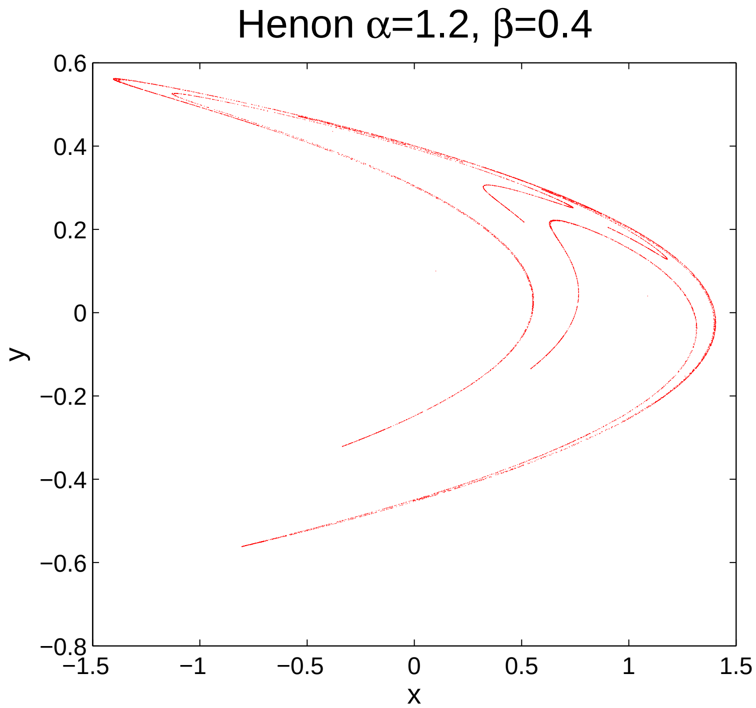

Figura 3: Mapa de Henon con $\alpha = 1{,}2$, $\beta = 0{,}4$.