

# Sistemas complejos

## The Nonlinear Workbook II

Carlos Manuel Rodríguez Martínez

Universidad Veracruzana

26 de febrero de 2016

# Volvemos con sistemas Dinámicos

- Regla determinista para describir la evolución de un sistema.
- Sea un sistema descrito por un conjunto finito de variables dinámicas  $\vec{x} = (x^{(1)}, \dots, x^{(n)}) \in \mathbb{R}$ .
- La evolución temporal del sistema estará descrita por:

$$\frac{d}{dt}\vec{x}(t) = \vec{F}(\vec{x}(t), t).$$

# Ecuación logística

- Aplicación: Es un modelo para describir el crecimiento poblacional

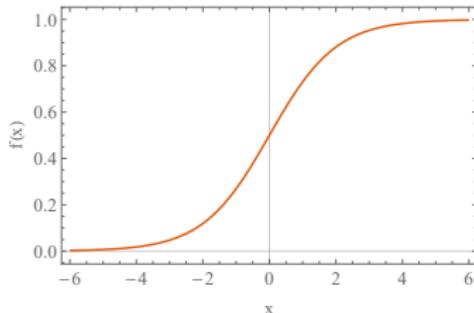
$$\frac{dP}{dt} = rP \left(1 - \frac{P}{k}\right),$$

donde  $r$  es la razón de crecimiento y  $k$  es la capacidad de carga.

- Su solución: La función sigmoide

$$P(t) = \frac{KP_0 e^{rt}}{K + P_0 (e^{rt} - 1)}.$$

- Su gráfica:



# Ecuación logística

- En general la ecuación logística es

$$\frac{dx}{dt} = rx(1 - x).$$

- Por medio de la aproximación de Euler

$$\frac{x(t+h) - x(t)}{h} = rx(1 - x),$$

haciendo  $x(t+h) = x_{n+1}$ ,  $x(t) = x_n$ ,

$$x_{n+1} = x_n + hr x_n (1 - x_n).$$

- Con el cambio de variable

$$x_n = \frac{1 + hr}{hr} y_n,$$

queda

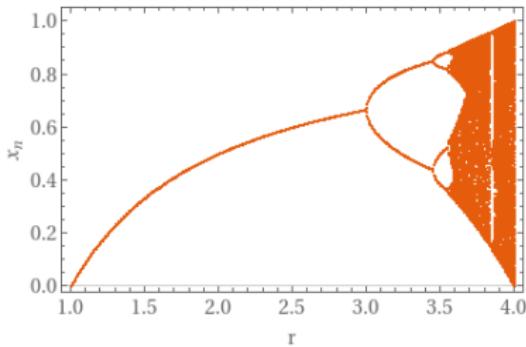
$$y_{n+1} = (1 + hr)y_n(1 - y_n).$$

# Ecuación logística

- Renombrando  $y_{n+1} = x'_{n+1}$ ,  $y_n = x'_n$  y  $1 + hr = r'$  queda

$$x'_{n+1} = r' x'_n (1 - x'_n).$$

- Este es el mapa logístico con el que se ha estado trabajando.
- Su diagrama de bifurcación es



# Diagrama Cobweb

- Es una herramienta visual para investigar las características de las órbitas de un sistema dinámico.

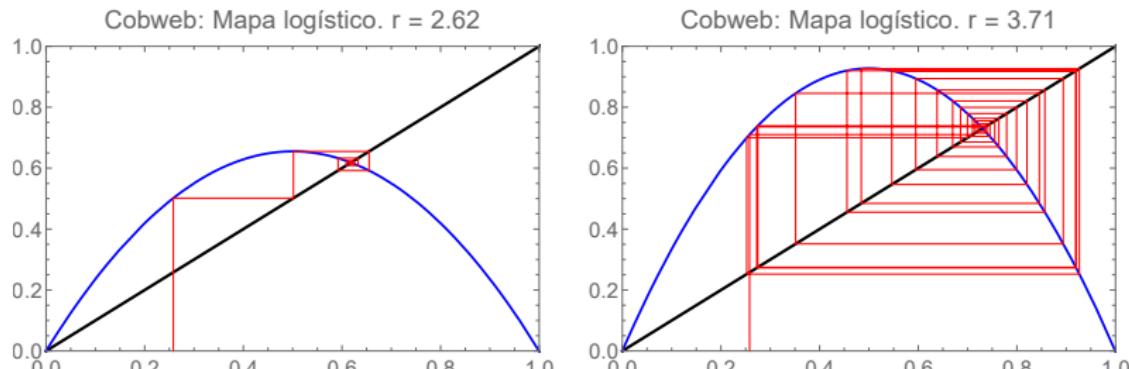


Figura: Diagrama Cobweb para el mapa logístico.

# Exponente de Lyapunov

- Es una medida de esta divergencia entre dos trayectorias infinitesimalmente cercanas.
- Aquí se supone que las trayectorias divergen exponencialmente, de manera que

$$\epsilon_n = \epsilon_0 e^{n\lambda},$$

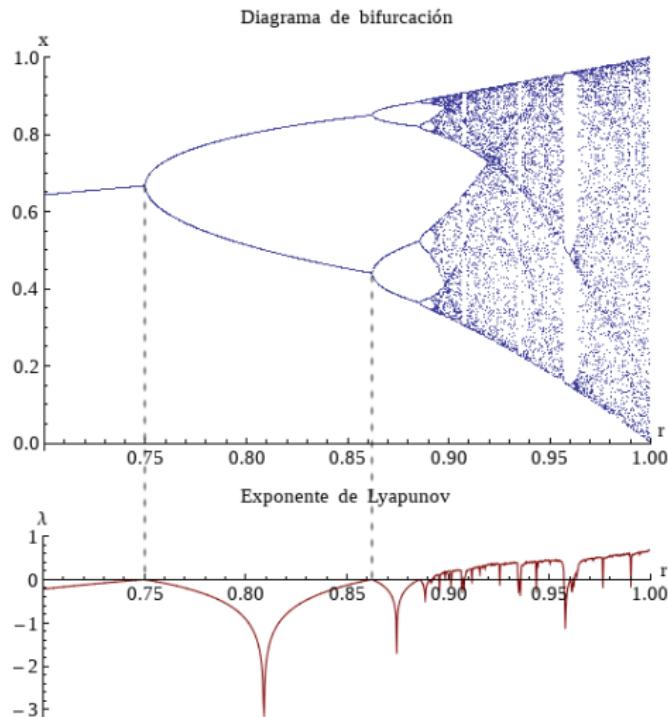
- El *exponente de Lyapunov* queda definido como

$$\lambda = \frac{1}{n} \sum_{k=0}^{n-1} \ln |F'(x_k)|.$$

- Si  $\lambda < 0$  la órbita tiende hacia algún valor, si  $\lambda > 0$  no tiende hacia ningún valor. Cuando  $\lambda = 0$  la trayectoria no diverge.

# Exponente de Lyapunov: Mapa logístico

- En el mapa logístico se observa que el exponente de Lyapunov es mayor que 0 en la zona caótica.

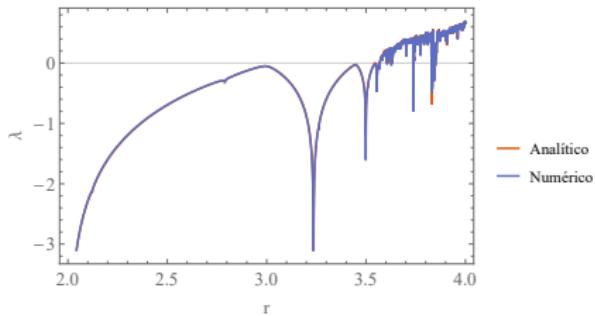


# Exponente de Lyapunov: Numérico

- Si no se conoce la derivada de la función que define el mapeo se puede calcular numéricamente.
- Sea  $x_{n+1}(x_n) = f(r, x_n)$ ,  $\tilde{x}_{n+1}(x_n - \epsilon) = f(r, x_n - \epsilon)$ .
- El exponente de Lyapunov es

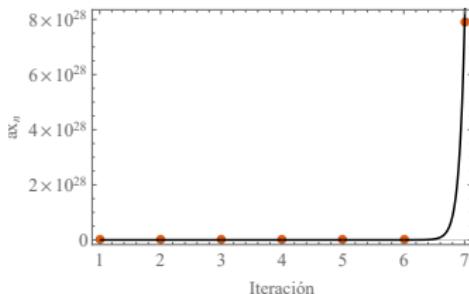
$$\lambda = \frac{1}{n} \sum_{k=0}^{n-1} \log \left| \frac{x_{n+1}(x_n) - \tilde{x}_{n+1}(x_n - \epsilon)}{\epsilon} \right|.$$

- Para el mapa logístico



# Generador de números aleatorios

- Dado que un sistema computacional es uno completamente determinista, es decir que se puede conocer su estado siguiente a partir del estado actual, es técnicamente imposible para este sistema producir números totalmente aleatorios.
- A partir de la función  $f(x) = a^x$ .
- Esta función crece muy rápidamente, a una velocidad  $f'(x) = a^x \log a$ .
- Por lo tanto los puntos  $f(x)$  y  $f(x + i)$  donde  $i \in \mathbb{Z}^+$  son muy distantes.



# Generador de números aleatorios

- Aprovechando esto resulta que la función

$$f(x) = a^x \bmod m$$

es caótica.

- Restringiéndonos a  $x \in \mathbb{Z}^+$  se puede representar de forma recurrente.

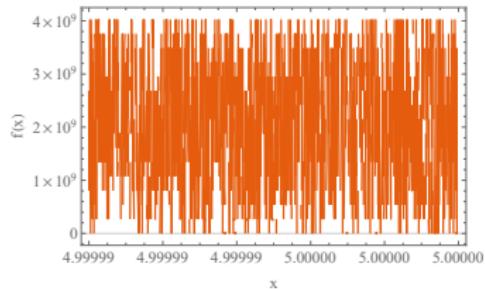
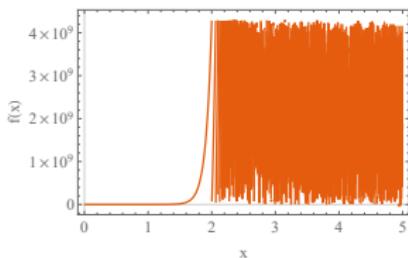
$$x_{n+1} = ax_n \bmod m.$$

- En general

$$x_{n+1} = (ax_n + c) \bmod m$$

también es caótico.

Su gráfica es



# Generador de números aleatorios

- A la relación de recurrencia

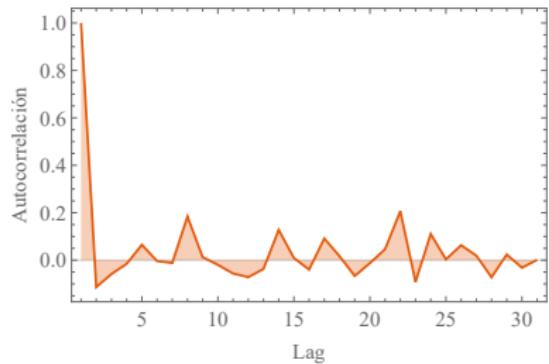
$$x_{n+1} = (ax_n + c) \bmod m$$

se le conoce como *Lineal Congruential Generator* (LCG).

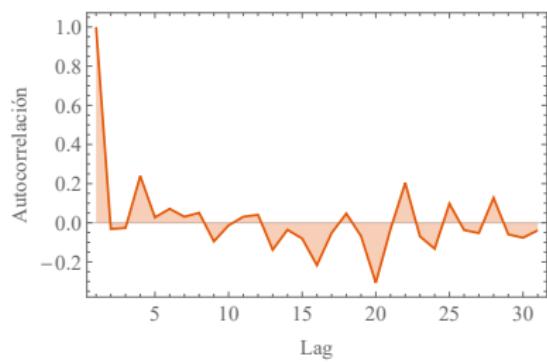
- $m$ : Módulo  $m > 0$ .
- $a$ : Multiplicador  $0 \leq a < m$ .
- $c$ : Incremento  $0 \leq c < m$ .
- $x_0$ : Valor inicial  $0 \leq x_0 < m$ .
- Ej: Si  $m = 10$ ,  $x_0 = a = c = 7$  la secuencia generada es:  
 $7, 6, 9, 0, 7, 6, 9, 0, 7, 6, 9, 0, 7, 6, 9, 0, 7, 6, 9, 0, 7$ .
- La mayoría de los compiladores de C++ usan una versión del LCG en su función `rand()`. Por ejemplo, el compilador GNU usa:  $m = 2^{31}$ ,  $a = 1103515245$ ,  $c = 0$ . Randu usó:  $m = 2^{31}$ ,  $a = 65539$ ,  $c = 0$ .

# Generador de números aleatorios: RANDU

- “Si tuviéramos que eliminar todos los artículos científicos cuyas conclusiones han quedado invalidadas por culpa de usar RANDU, quedaría en cada estantería un hueco del tamaño de un puño.” William H. Press, Saul Teukolsky, Numerical Recipes.



(a) GNU.

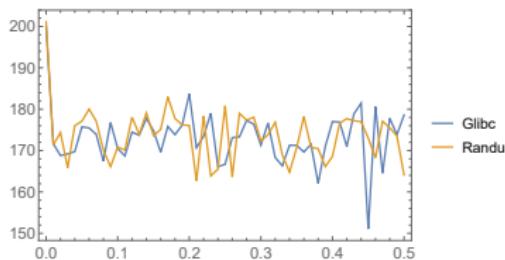


(b) Randu.

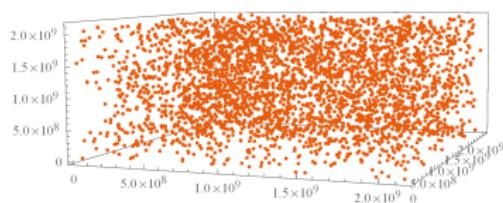
Figura: Correlograma de LCG.

# Generador de números aleatorios: RANDU

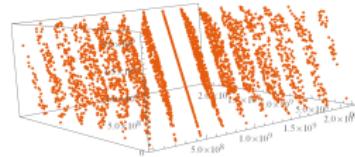
- La transformada de Fourier tampoco dice mucho



- Agrupando los puntos de 3 en 3 y graficándolos se observa el patrón.



(a) GNU.

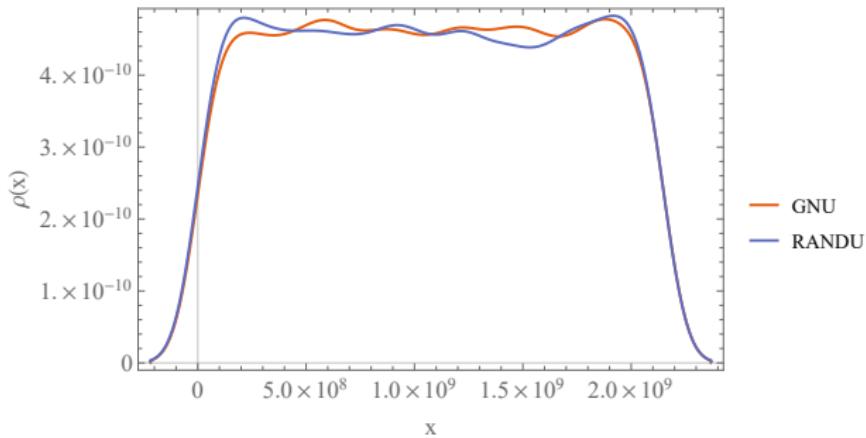


(b) Randu.

Figura: Gráfica de puntos de LCG.

# Generador de números aleatorios: Densidad

Tanto la densidad del generador de GNU como RANDU es aparentemente uniforme.



# Generador de números aleatorios: Lyapunov

- Se puede analizar la calidad del generador de números aleatorios evaluando el exponente de Lyapunov. Con  $c = 0$  se evalúa el exponente de Lyapunov respecto al valor de  $a$ .

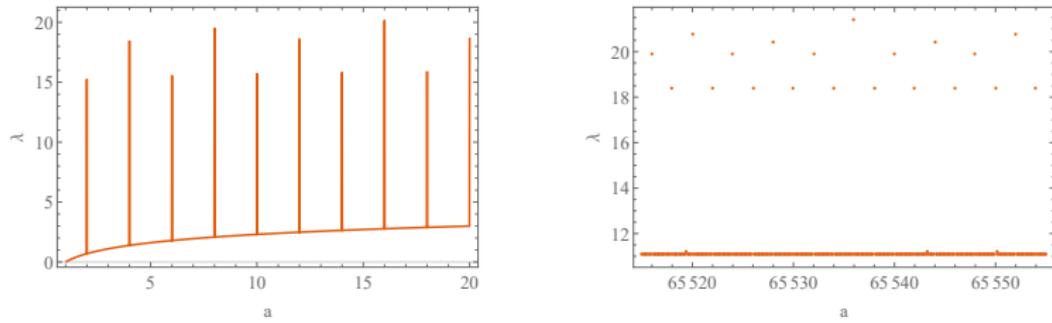
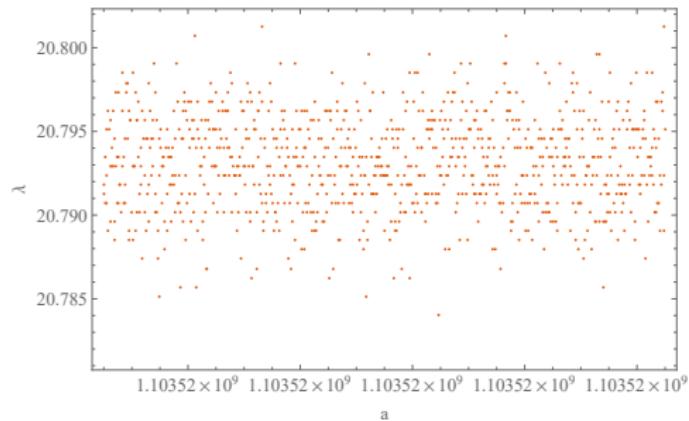


Figura: Exponente de Lyapunov para diferentes valores de  $a$ .

- El exponente de Lyapunov de Randu es: 11.0904.

# Generador de números aleatorios: Lyapunov

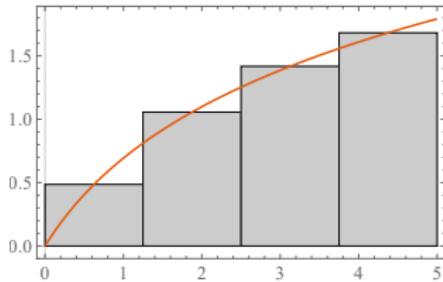
- El exponente de Lyapunov de GNU es: 20.794.



# Métodos de integración

- Con la integral  $I = \int_a^b f(x) dx$  se quiere obtener el área bajo la curva  $f(x)$ .
- Una forma de obtener esto es colocando rectángulos verticales bajo la curva y sumar áreas.

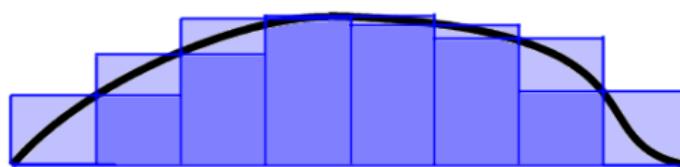
Área estimada = 5.8005  
Área real = 5.7506



A esto se le conoce como suma de Riemann.

# Métodos de integración

- Algunas funciones no se pueden aproximar bien mediante este método. Otra forma alternativa es colocar rectángulos horizontales.



A esto se le conoce como integral de Lebesgue.

## Métodos de integración: Monte-Carlo

- Una forma numérica de obtener el valor de una integral es a través de *método de Monte-Carlo*.
- Sea  $f$  una función que mapea de  $f : [0, 1] \rightarrow [0, 1]$ , para evaluar la integral

$$I = \int_0^1 f(x) dx$$

se puede aproximar como

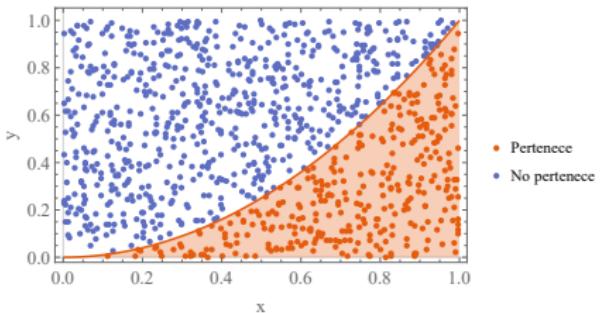
$$I = \frac{1}{N} \sum_j z_j + O(N^{1/2}),$$

donde

$$z_j = \begin{cases} 0 & y_j > f(x_j) \\ 1 & y_j \leq f(x_j) \end{cases}.$$

# Métodos de integración: Interpretación de Monte-Carlo

- Supongamos que queremos integrar la función  $f(x) = x^2$  en el intervalo  $[0, 1]$ .



- La razón entre los puntos que pertenecen y no pertenecen es una buena estimación de área.
- La solución analítica de la integral es  $I = \frac{1}{3}$ .
- Utilizando el LCG con los parámetros GNU se obtiene  $I = 0.333$ .
- Utilizando Randu se obtiene  $I = 0.358$ .

# Método para resolver ecuaciones algebraicas: Newton-Raphson

- Sea  $f(x)$  una función de la cual se desea conocer sus raíces (puntos en los que  $f(x) = 0$ ).
- Se desea construir un método iterativo con el cual nos podamos ir aproximando a la raíz en un número finito de pasos.
- Sea  $x_0$  un punto inicial donde se sospecha que la raíz está cerca. Quiero conocer cómo es la función en el punto  $x_0 + \epsilon$ .

$$f(x_0 + \epsilon) = f(x_0) + f'(x_0)\epsilon + \frac{1}{2}f''(x_0)\epsilon^2 + \dots$$

- Tomando los términos hasta a primer orden queda

$$f(x_0 + \epsilon) \approx f(x_0) + f'(x_0)\epsilon.$$

# Método para resolver ecuaciones algebraicas: Newton-Rhapson

- La raíz ocurre cuando  $f(x + \epsilon) = 0$ , entonces

$$f(x_0) + f'(x_0)\epsilon = 0.$$

El valor de  $\epsilon$  que satisface esto es

$$\epsilon = -\frac{f(x_0)}{f'(x_0)}.$$

- Esto indica que

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

es un punto más cercano a la raíz.

- Al realizar este procedimiento  $n$  veces se obtiene el método de Newton-Rhapson.

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

# Newton-Rhapson: Ejemplo

- Ejemplo de aproximación por el método de Newton-Rhapson.

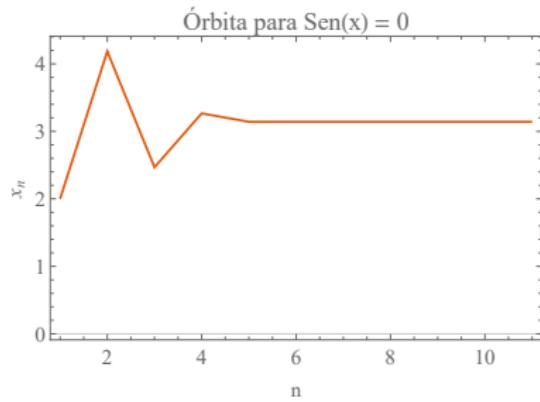
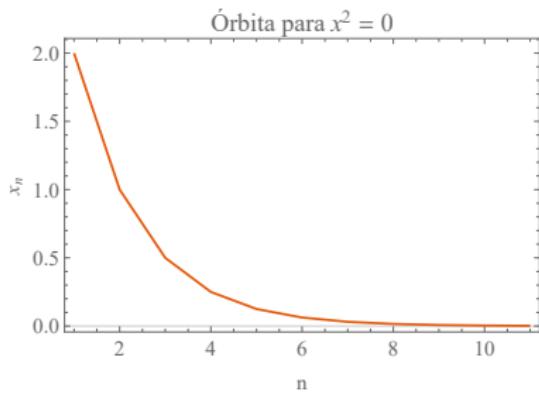


Figura: Aproximación a la solución por el método de Newton-Rhapson.

# Newton-Rhapson: Diagrama Cobweb

- El mapeo de Newton-Rhapson está definido por la función

$$g(x) = x - \frac{f(x)}{f'(x)},$$

con la condición de convergencia  $|g'(x)| < 1$ .

- Se puede utilizar el diagrama Cobweb para analizar la convergencia de las órbitas.

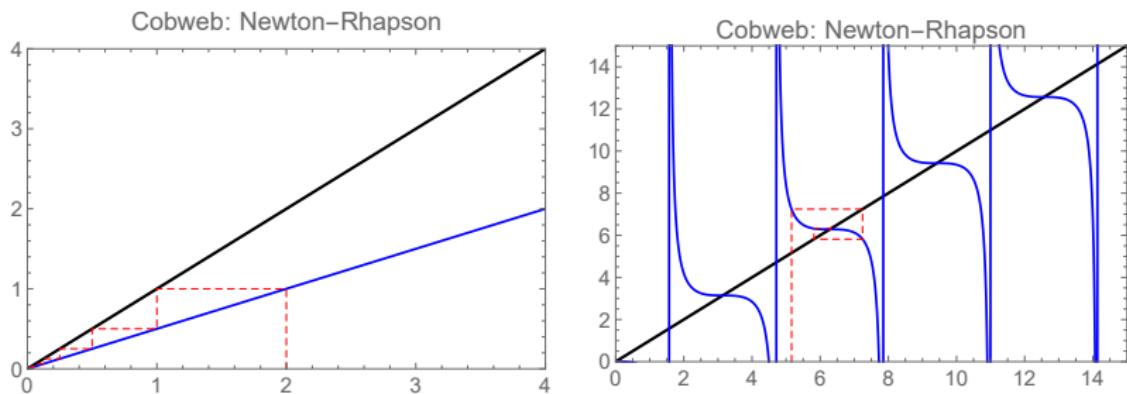


Figura: Diagrama Cobweb para el mapeo de Newton-Rhapson.

# Newton-Raphson: Convergencia

Calculando  $|g'(x)|$  podemos saber si determinado punto convergerá a una solución.

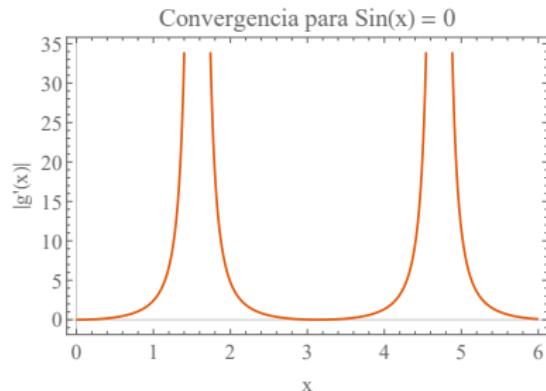
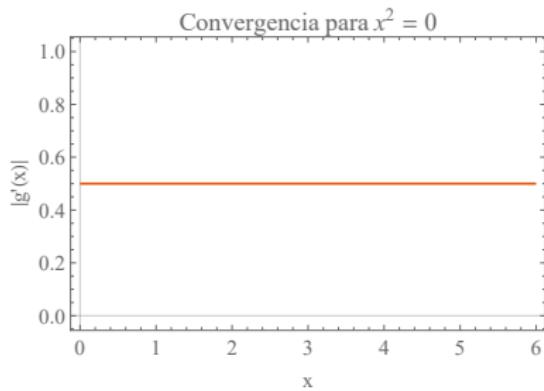


Figura: Convergencia para el mapeo de Newton-Raphson.

## Newton-Rhapson: Convergencia

- En el diagrama Cobweb para la función  $f(x) = \sin(x)$  se observa que el punto al cual converge el mapeo depende del punto inicial.
- Si se aplica el método de Newton-Rhapson para encontrar una solución en números complejos, la convergencia es fractal.

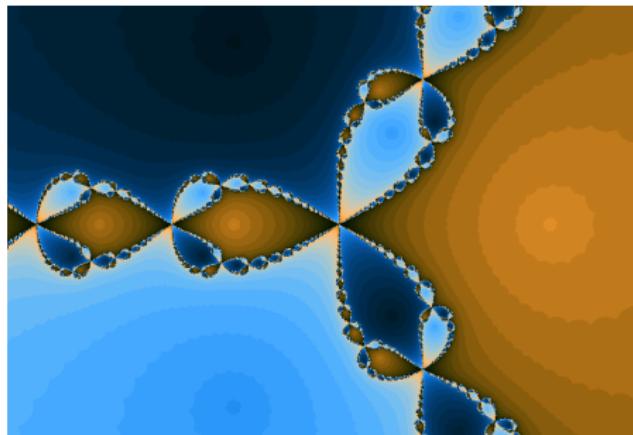


Figura: Convergencia para la función  $z^3 - 1 = 0$ .

# Mapa de rotacioness

- Se tienen dos sistemas oscilando a frecuencias  $\nu$  y  $\nu'$ .
- El estado de cada sistema se puede describir por una variable angular

$$\theta(t) = \nu t \bmod 1.$$

- Se va a registrar el valor de  $\theta$  a una frecuencia  $\nu'$ , es decir se va a medir

$$\theta_n = \theta \left( t_0 + \frac{n}{\nu'} \right).$$

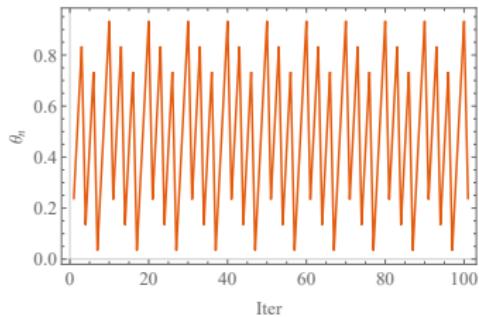
- La forma recurrente es

$$\theta_{n+1} = (\theta_n + \Omega) \bmod 1,$$

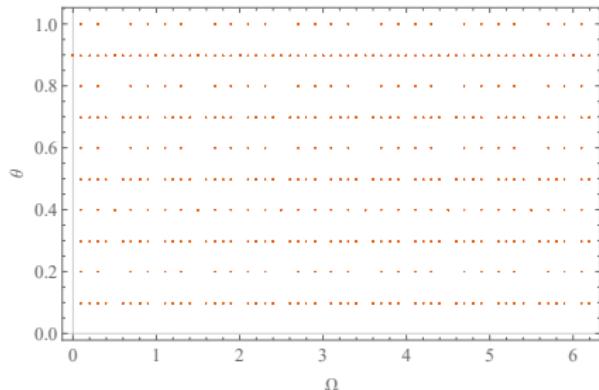
donde  $\Omega = \frac{\nu}{\nu'}$ .

# Mapa de rotaciones

- Con  $\Omega = 0.3$  la órbita de  $\theta_n$  es



- Si se grafican los puntos de la órbita para varios valores de  $\Omega$  se obtiene



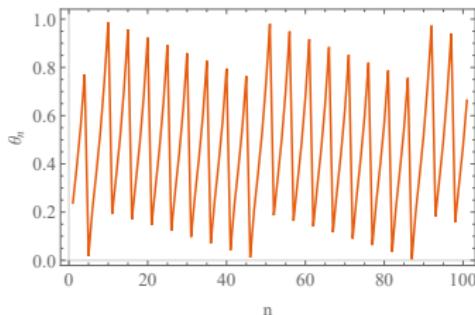
# Circle Map

- Si los osciladores están acoplados

$$\theta_{n+1} = \theta_n + \Omega - \frac{k}{2\pi} \sin(2\pi\theta_n) \bmod 1,$$

donde  $k$  es la constante de acoplamiento.

- Su órbita para  $\Omega = 0.2$  y  $k = 0.3$



- Nos interesa conocer si los dos osciladores entran en resonancia. Si ocurre esto la órbita converge hacia algún punto fijo.

# Número de rotación (Winding Number)

- Para investigar si la órbita converge

$$\theta_{n+1} = \theta_n + \Omega - \frac{k}{2\pi} \sin(2\pi\theta_n).$$

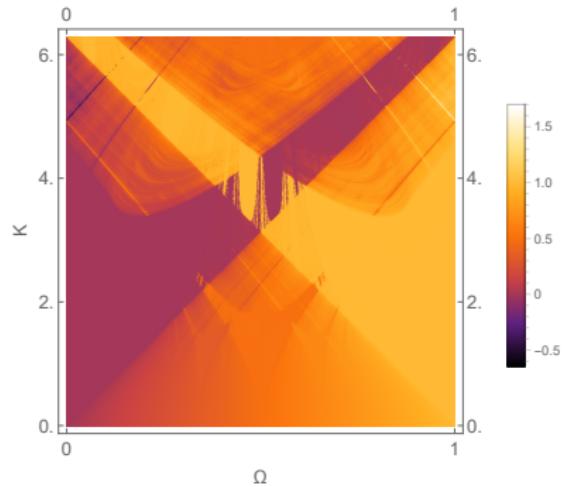
- El Winding Number está definido por

$$\rho = \lim_{T \rightarrow \infty} \frac{1}{T} \left( f^{(T)}(x) - x \right)$$

se interpreta como el número promedio de rotaciones por iteración.

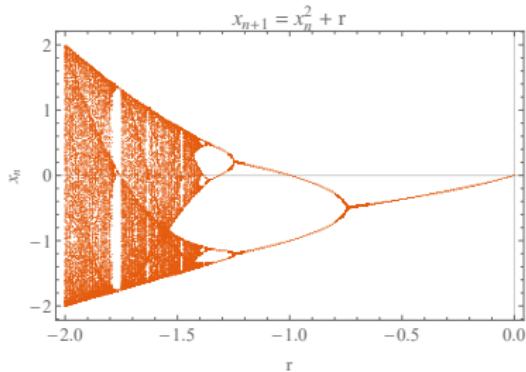
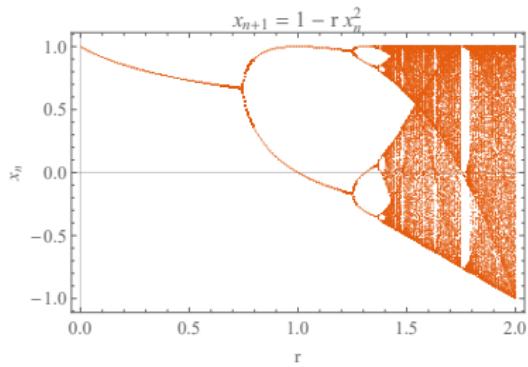
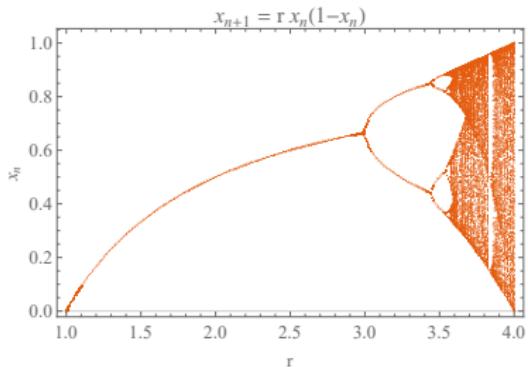
# Número de rotación (Winding Number)

- El mapa generado de graficar el Winding Number respecto a  $\Omega$  y  $k$



# Constante de Feigenbaum

Numerosos mapeos exhiben bifurcaciones.

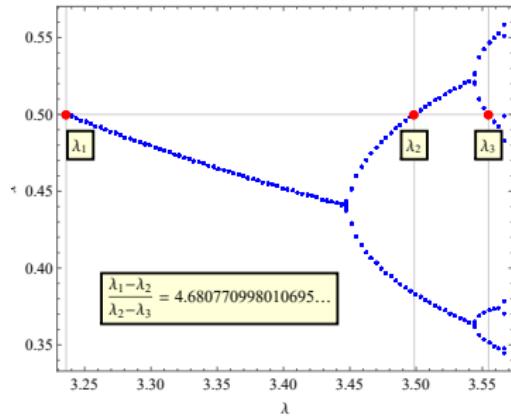
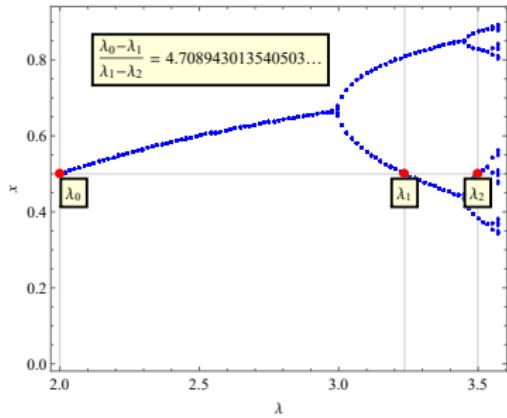


# Constante de Feigenbaum

- Sea  $\lambda_n$  el valor de  $r$  en el cual existe una bifurcación de un punto periódico con periodo  $n$ .
- Para muchos mapeos se encuentra que

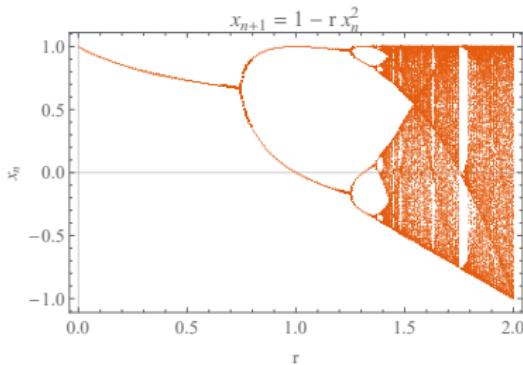
$$\delta = \lim_{n \rightarrow \infty} \frac{\lambda_{n+1} - \lambda_n}{\lambda_{n+2} - \lambda_{n+1}} = 4.669201609 \dots$$

- Ejemplo:



# Dinámica simbólica

- Se tiene el mapeo  $x_{n+1} = 1 - rx_n^2$ .



- Supongamos un  $r$  tal que la órbita es periódica con periodo  $n$ .
- A partir de un punto final  $x_n$  no puedo saber cuál fue el punto inicial  $x_0$  debido a que para cada punto  $x_n$  hay dos posibles valores de  $x_{n-1}$

$$x_{n-1} = \sqrt{\frac{(1-x_n)}{r}}, \quad x_{n-1} = -\sqrt{\frac{(1-x_n)}{r}}$$

## Dinámica simbólica

- Etiquetando los puntos según su raíz sí podría conocer el valor de  $x_0$ , de manera que

$$x_0 = f_{\sigma_0}^{-1} \circ f_{\sigma_1}^{-1} \circ \cdots \circ f_{\sigma_{n-1}}^{-1}(x_n),$$

donde los  $\sigma$  son las etiquetas.

- Para simplificar la notación puede definirse

$$\sigma(x_n) = f_\sigma^{-1}(x_n),$$

de manera que

$$x_0 = \sigma_0 \circ \sigma_1 \circ \cdots \circ \sigma_{n-1}(x_n).$$

- Para el mapa  $x_{n+1} = 1 - rx_n^2$  como tiene dos raíces se pueden nombrar

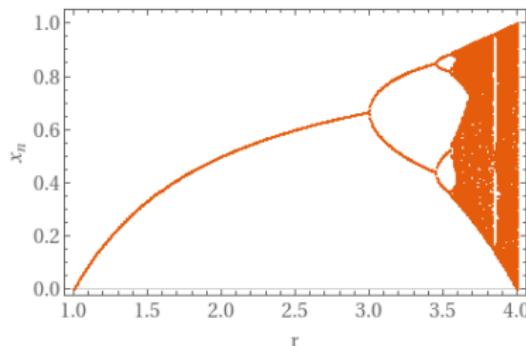
$$R = \sqrt{\frac{(1-x_n)}{r}}, \quad L = -\sqrt{\frac{(1-x_n)}{r}}$$

## Dinámica simbólica

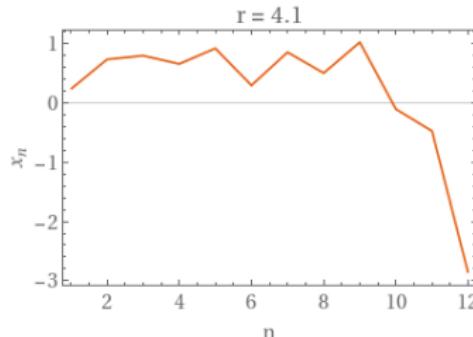
- De manera que se puede describir su dinámica por la secuencia de  $R$  y  $L$ . A esto se le conoce como *kneading sequence*, se denota por  $K$ .
- Por ejemplo, para el punto  $r = 1.394$ , con 30 iteraciones,  
 $K = RLRRRLRLRLRRRLRLRLRRRLRLRLRRRL.$
- Para  $r = 1.48$ ,  
 $K = RLRRRRRLRRRRRLRRRRRLRRRRRLRRRRR.$

# Repulsor caótico

- Hasta ahora siempre se ha visto el mapa logístico con  $r \leq 4$ .



- ¿Qué pasa con la órbita cuando  $r > 4$ ? Por lo general tiende a  $x_n \rightarrow -\infty$ .



## Repulsor caótico

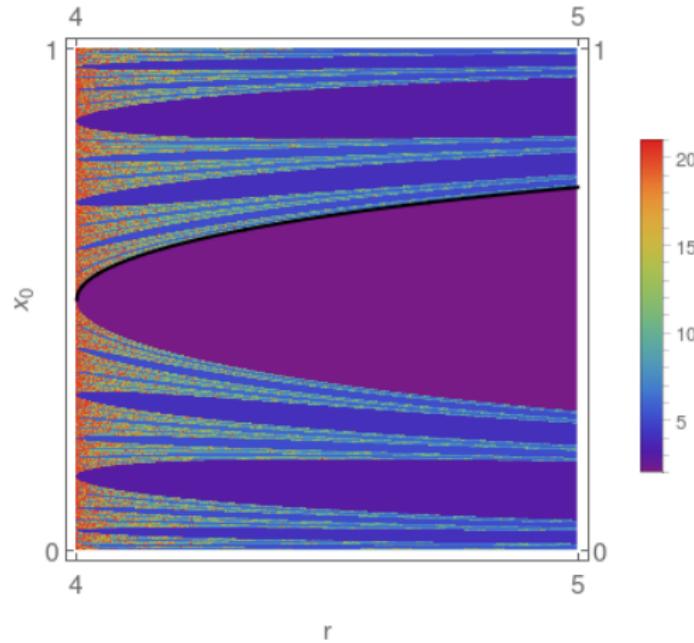
- Sea  $s(r) = \frac{r}{4} - 1$ , se define el gap como

$$\text{gap}(r) = \sqrt{\frac{s(r)}{1+s(r)}}, \quad \text{o} \quad \text{gap}(r) = \frac{1}{2} \left( \sqrt{\frac{s(r)}{1+s(r)}} + 1 \right),$$

- Los puntos  $x < \text{gap}(r)$  siempre divergen a  $x_n \rightarrow -\infty$ . En los puntos  $x > \text{gap}(r)$  la divergencia no está garantizada.
- Ej: Para  $r = 5$ ,  $\text{gap}(5) = 0.723607$ , entonces la órbita del mapa logístico con  $r = 5$ ,  $x_0 = 0.5$  es  
 $0.5, 1.25, -1.5625, -20.0195, -2104.01, -2.21447 \times 10^7, -2.45194 \times 10^{15}, -3.00601 \times 10^{31}, -4.51806 \times 10^{63}, -1.02064 \times 10^{128}, -5.20857 \times 10^{256}$ .

# Repulsor caótico

- Se graficará el número de iteraciones que tarda en divergir cada punto.



# Conclusión

Gracias por su atención.