Programación 2024

Guía 8: Matplotlib: Graficación

23 de Octubre de 2024

Antes de comenzar los problemas genere un nuevo directorio `guia8` donde trabajará y guardará todos los programas y archivos que se producirán en este práctico.

Asegúrense de que todos los gráficos se guarden en archivos .png y luego transfieran a su computadora para verlos.

Problema 1: Realice la graficación de la función $f(x) = sin(x)/|x|$ entre -2π y 2π usando arrays para x y y. Busque la resolución adecuada. Ponga de título la función, aclare los ejes como x y $f(x)$. Superponga con la función $f_2(x) = x/6$. Guarde la salida del gráfico en un archivo .png.

Problema 2: Se requiere realizar evaluaciones de la función de Besel de orden 0 a través de la serie

$$J_0(x) = 1 - \frac{x^2}{2^2} + \frac{x^4}{2^2 4^2} - \frac{x^6}{2^2 4^2 6^2} + \cdots \tag{1}$$

(a) Diseñar una rutina que dados n (número de términos) y x calcule la serie de la función de Besel.

(b) Realice una función para determinar cuantos términos son necesarios para obtener una precisión de δ_p.

(c) Grafique entre 0 y 20 la función de Bessel aproximada y la función de Bessel provista por scipy.

(d) Determinar cual es la primera raíz positiva de la función de Bessel de orden 0 con una resolución de $delta_p = 0.5, 0.1, 0.01$.

(e) Determine la raíz de la función de Bessel de orden 0 en el intervalo [0,5] usando el método de bisección.

Problema 3: Grafique la función $z = x^2 + y^2 - 4$. Utilice los comandos de graficación *contour* y $plot\_surface$. Utilice también pyplot con axes(projection='3d'), ax.contour3D(X, Y, Z, 50, cmap='binary')

Problema 4: Una función iterada a orden n se define de la siguiente manera :

$$f^n(x) = f(f(f(...(x)...))) = f \circ f \circ f \circ ...f \tag{2}$$

esto es; como la composición n-esima de la función consigo misma.

Esta secuencia puede expresarse tambi'en de la siguiente manera :

$$\begin{aligned} f^0 &= x \\ f^n &= f(f^{n-1}(x)) = f \circ f^{n-1} \end{aligned} \tag{3}$$

- Realice una función con la función iterada de orden n que tenga de entrada el nombre de la función a iterar y el orden. Reutilice los códigos desarrollados en guías anteriores.

- Diseñe un programa que calcule la función iterada de orden n-ésima para la función.

$$f(x) = x^2 - 0.75 \tag{4}$$

- Evaluar la función de orden $n = 3 - 6$ en el intervalo $|x| \leq 1.5$ con 100 puntos usando *linspace*.
- Graficar las evaluaciones de la función. Describir a que tiende la función a medida que $n \to \infty$.

Problema 5: La segunda ley de Newton en 1D se expresa como :

$$\frac{dv}{dt} = \frac{f(x,v)}{m} \tag{5}$$

$$v = \frac{dx}{dt} \tag{6}$$

donde $f(x,v)$ es la fuerza que actúa sobre la partícula, m es la masa, x es la posición y v la velocidad de la misma. La fuerza puede variar según la posición y velocidad de la partícula.

Usando la derivada como el cociente incremental podemos expresar como :

$$\frac{v(t+dt) - v(t)}{dt} = \frac{f(x(t), v(t))}{m} \tag{7}$$

$$\frac{x(t+dt) - x(t)}{dt} = v(t) \tag{8}$$

Si los tiempos estan ordenados de manera que $t_i = i \cdot dt$ $(i = 1, ..., N)$ entonces podemos definir $x(t_i) = x_i$ y $v(t_i) = v_i$ y las ecuaciones serían :

$$\frac{v_{i+1} - v_i}{dt} = \frac{f(x_i, v_i)}{m} \tag{9}$$

$$\frac{x_{i+1} - x_i}{dt} = v_i \tag{10}$$

despejando la velocidad y la posición en el tiempo $i + 1$, obtenemos un par de ecuaciones recursivas:

$$v_{i+1} = v_i + dt \cdot \frac{f(x_i, v_i)}{m} \tag{11}$$

$$x_{i+1} = x_i + dt \cdot v_i \tag{12}$$

Esta últimas ecuaciones permiten calcular conociendo los valores iniciales x_0 y v_0 en el tiempo $t = 0$ los valores de x_i, v_i a cualquier tiempo $t_i = i \cdot dt$.

Implemente un programa que permita resolver la segunda ley de Newton para el caso de un oscilador armónico amortiguado, donde la fuerza tiene la expresión :

$$f(x,v) = -kx - bv \tag{13}$$

donde b es la constante de amortiguamiento y k es la constante del resorte.
El valor de $dt = 0.1$ puede ser adecuado para el caso particular de este ejercicio.
Asumiendo $k = 1$ en unidades arbitrarias, encuentre el efecto de aumentar b (0.1, 0.2, 0.3 ...) graficando $x(t)$ y observando el amortiguamiento de las oscilaciones.

Problema 6: Grafique la trayectoria de una partícula en movimiento rectilíneo y circular uniformes en tres dimensiones. Reutilice los códigos desarrollados en la Guía anterior. Utilice pyplot con gca(projection='3d').

Problema 7: Caos en el mapa logístico. El mapa logístico es definido por la ecuación iterativa

$$x_{i+1} = rx_i(1 - x_i) \tag{14}$$

Este es un mapa iterativo que puede tener tres comportamientos límites

- Puntos fijos. Es decir una vez que se obtiene el valor del punto fijo todas las iteraciones siguientes daran el mismo valor. Por ejemplo el $x = 0$.
- Ciclo límite. Se los valores son repetitivos y periódicos (secuencia de valores que se repiten).
- La evolución se vuelve totalmente loca, parece que fueran números aleatorios sin no ni son. Sin embargo no son aleatorio es un fenómeno puramente deterministico denominado caos. Aun cuando tengamos la ecuación de evolución la predicción se vuelve impredecible, pequeñas perturbaciones de la situación actual conllevan a grandes desviaciones en pocas iteraciones.

(a) Implemente una función que simule la evolución de la población de la especie en funcion de la generación $i = 1, 2, ...T$.

(b) Estudie las soluciones de este problema en función del parámetro r (r entre 0 y 5). Para un dado r comience con $x_0 = 1/2$ y resuelva 1000 iteraciones.

(c) Grafique para $r = 0.5, 1.5, 3.2, 3.53.7$ los valores de las poblaciones x posibles entre i=100 a i=400 como función del tiempo i.

(d) Grafique para cada r los valores de las poblaciones x posibles entre i=100 a i=400 como función de r 3 y 4. Grafique con resoluciones de 10, 100, 1000 puntos.

(e) Grafique en un gráfico (r, x) donde x esta en el eje vertical integraciones comenzando por x=1/2 (conviene desechar las primeras 1000 iteraciones) y con valores de r entre 2 y 4 con pasos de 0.01 grafique con puntos 'k.' en la función *plot* o utilizando la función `scatter`.

Lo que debería obtener es un gráfico que se parece a un "árbol", conocido como gráfico de Feigenbaum (Buscar en internet el gráfico y corroborar que les ha dado lo mismo).

Problema 8: Interferencia de ondas.

Suponga que se tiran un par de piedras en una pileta las cuales cada una genera ondas circulares que se desparraman con el tiempo produciendo interferencia entre las dos fuentes.

Cada onda se puede representar por $\zeta_i = \zeta_0 \sin(kr_i)$ con $i = 1, 2$ donde asumimos que ambas ondas tienen la misma longitud de onda y la misma amplitud y difieren en su centro (lugar donde cayó la piedra); $r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$, luego el patrón de la pileta es la suma $\zeta = \zeta_1 + \zeta_2$. Notar que ζ representa el desplazamiento vertical de la superficie del agua.

(a) Realice una función que determine el patrón ondulatorio de una piedra.

(b) Realice una función que permita evaluar el patrón ondulatorio de una piedra en un dominio o grilla cuadrada dados un δx y δy y un largo del estanque de $L_x = L_y$.

(c) Evalue el patrón de interferencia asumiendo que las ondas tienen una longitud de onda de $5cm$; $k = 2\pi/\lambda$; una amplitud de 1cm y los centros estan separados por 20cm. El estanque tiene un metro de largo y requerimos una resolución.

(d) Realice un gráfico de densidad del patrón de interferencia.

Problema 9: Curvas paramétricas en un gráfico.

(a) Haga un gráfico con la curva deltoide que se define paramétricamente con las ecuaciones

$$x = 2\cos\theta + \cos 2\theta \tag{15}$$
$$y = 2\sin\theta - \sin 2\theta \tag{16}$$

donde $0 \leq \theta < 2\pi$.

(b) Grafique la espiral de Galileo la cual viene definida en coordenadas polares por $r = \theta^2$ para $0 \leq \theta \leq 10\pi$.

(c) Considerando coordenadas polares en el rango de a $0 \leq \theta \leq 24\pi$ grafique la función:

$$r = e^{\cos\theta} - 2\cos(4\theta) + \sin^5 \frac{\theta}{12}. \tag{17}$$