

Tarea 01. Graficador de funciones

José Miguel Saavedra Aguilar

20 de agosto de 2022

Resumen

En esta tarea se presenta un graficador de funciones reales en un intervalo $[a, b] \subset \mathbb{R}$. Dicho graficador fue programado en Python utilizando la librería gráfica **Graphics.py** de John Zelle. Con esta librería, podemos graficar puntos, líneas, texto, y figuras geométricas básicas. El graficador funciona para funciones definidas para arreglos numéricos de **Numpy**, incluye un marco, malla, ejes coordenados y zoom sobre la función. Así mismo, se estudian los detalles de la construcción estética del graficador, como es la malla, los ejes coordenados y la gráfica por sí misma. Finalmente, se hace una muestra del funcionamiento del graficador utilizando múltiples funciones evaluadas en distintos intervalos, lo que nos permite estudiar la función a detalle.

1. Introducción

La librería gráfica de Python **Graphics.py** fue creada por John Zelle para su libro [Zelle(2004)] para trabajar con gráficos por computadora de forma orientada a objetos. En la versión 5, la más actual al día de hoy, podemos utilizar los siguientes objetos: Punto, línea, círculo, óvalo, rectángulo, polígono, texto e imágenes. La función principal de **Graphics.py** es **GraphWin(título, ancho, alto)**, que crea la ventana gráfica **título** de tamaño **ancho** \times **alto**.

2. Metodología

Una vez que conocemos la librería gráfica, queda utilizarla para graficar una función $f : \mathbb{R} \mapsto \mathbb{R}$ en el intervalo $[a, b]$.

Para esto, iniciamos con un vector de n puntos equiespaciados entre a y b , que llamaremos x_i para $i = 1, \dots, n$. Así mismo, llamamos $y_i = f(x_i)$ a la función f evaluada en x_i . Para evitar los errores numéricos con $x \approx x$ tales que f no está definida en x , en el caso que $|y_i| > y_{max}$, no se tomará en cuenta este punto para el resto de los cálculos ni para graficar. Ahora que tenemos todos los puntos de la gráfica, debemos reescalarlos para que sean visibles en la gráfica. En el graficador, se introduce el ancho P_x y alto P_y , en píxeles, de la gráfica, así como si se desea un marco alrededor de ésta. El marco es de $w_{marco} = 50$ px, pero es posible modificarlo a cualquier valor positivo. En caso de no desear marco, se toma como un marco de valor $w_{marco} = 0$. Así, la ventana gráfica tendrá $P_y + 2w_{marco}$ px de alto y $P_x + 2w_{marco}$ px de ancho. Si llamamos y_{max}, y_{min} al máximo y mínimo valor de todas las

y_i respectivamente, definimos las siguientes cantidades:

$$\begin{aligned}d_y &:= y_{max} - y_{min} \\m_y &:= -\frac{P_y}{d_y} \\b_y &:= w_{marco} - y_{max}m_y \\d_x &:= b - a \\m_x &:= \frac{P_x}{d_x} \\b_x &:= w_{marco} - am_x\end{aligned}$$

Así, si tomamos $\hat{x}_i := m_x x_i + b_x$, $\hat{y}_i := m_y y_i + b_y$, tenemos

$$\begin{aligned}\hat{x}_1 &= w_{marco} \\ \hat{x}_n &= P_x + w_{marco} \\ \min_{i=1,\dots,n} \hat{y}_i &= w_{marco} \\ \max_{i=1,\dots,n} \hat{y}_i &= P_y + w_{marco}\end{aligned}$$

Es decir, \hat{x}_i, \hat{y}_i es un reescalamiento de la gráfica al tamaño, en pixeles, de la ventana. Los siguientes apartados son enteramente opcionales, pero dan una mejor apariencia a la gráfica. El primero de estos es la malla sobre x, y . Pintarla es muy sencilla, pues son líneas rectas, sin embargo, la dificultad, si es que podemos llamarla así, es el intervalo de la malla. Una primera interpretación puede ser tomarla cada cierto intervalo, sin embargo esto puede resultar contraproducente con ciertas funciones, pues la escala de este intervalo no coincide con la escala de la gráfica. Otra opción es tomar un número fijo de intervalos a lo largo de la gráfico, sin embargo resta utilidad a la malla y es puramente estético. La forma en que se tomarán los intervalos de la gráfica es calculando la escala de la gráfica.

$$\begin{aligned}c_x &:= \left\lceil \log_{10} \left(\frac{d_x}{2} \right) \right\rceil - 1 \\ c_y &:= \left\lceil \log_{10} \left(\frac{d_y}{2} \right) \right\rceil - 1 \\ s_x &:= 10^{c_x} \\ s_y &:= 10^{c_y}\end{aligned}$$

Si colocamos la malla sobre los puntos que marcan js_x, js_y para $j = -10, \dots, 10$ tales que sean visibles, tendremos una malla de a lo más 21 cortes verticales y horizontales, que marca en escalas de magnitud de orden 10, y que sirve como referencia para comparaciones. A la malla, se le agregan los ejes coordenados, cuando se pueden ver, y la escala de la gráfica. Esta escala está en los ejes coordenados cuando se muestran, o sobre el marco si no se muestran. El último elemento que añadimos a la gráfica antes de la función misma, es el título de la gráfica, especificado por el usuario.

Finalmente, para graficar la función, dibujamos una línea entre el punto (\hat{x}_i, \hat{y}_i) y el punto $(\hat{x}_{i+1}, \hat{y}_{i+1})$ para $i = 1, \dots, n - 1$ siempre que este punto se tome en cuenta para graficar.

Una vez que tenemos el graficador, resta hacer zoom dentro de la gráfica. Como se espera que el zoom pueda ser sobre cualquier punto de la gráfica, se creó una función que hace zoom

de forma artificial. La forma en que lo logra es calculando el punto medio c del intervalo $[a, b]$, $c = \frac{a+b}{2}$, y la amplitud d_x del intervalo, $d_x = b - c$. Se procede a graficar f en $[a, b]$. Una vez graficado, **Graphics.py** nos permite saber si se presiona una tecla, y cuál tecla se presionó. Con esta información, podemos hacer una ampliación (zoom 2.0x) de la gráfica, el graficador solicita un nuevo centro c^{new} , si es que se quiere cambiar, y reduce la amplitud a la mitad, es decir $d_x^{new} = \frac{d_x}{2}$. Así mismo, podemos hacer una reducción (zoom 0.5 x) de la gráfica, nuevamente se solicita un nuevo centro c^{new} , pero se aumenta la amplitud al doble, es decir $d_x^{new} = 2d_x$. Una tercera opción es mover la gráfica sin hacer zoom, para esto se toma un nuevo centro c^{new} y $d_x^{new} = d_x$. Posteriormente, se grafica f en $[a^{new}, b^{new}]$, donde

$$\begin{aligned} a^{new} &= c^{new} - d_x^{new} \\ b^{new} &= c^{new} + d_x^{new} \end{aligned}$$

3. Resultados

A continuación, se consideran los problemas de la tarea.

3.1. Problema 1

Así, tenemos un algoritmo graficador con el pseudocódigo 1, un graficador con zoom con el pseudocódigo 2.

Para utilizar el graficador, debemos definir una función que pueda ser evaluada en un arreglo de NumPy, por ejemplo:

```
def seno(x):
y=np.sin(x)
return y
```

Nuestro graficador tiene la siguiente sintaxis:

Graficar($f, a, b, n, P_x, P_y, \text{enmarcar}, \text{showGrid}, \text{showAxis}, \text{Título}$)

donde f es la función definida previamente, a y b son los extremos del intervalo donde se graficará, n es el número de puntos a evaluar dentro de $[a, b]$, P_x y P_y son el tamaño en pixeles de la ventana gráfica, **enmarcar**, **showGrid** y **showAxis** son Booleanos que definen si se muestra el marco, si se muestra la malla y si se muestran los ejes respectivamente; **Título** es un string que contiene el título de la gráfica.

A continuación, se dará un ejemplo del uso del Graficador para graficar la función $f(x) = \sin(x)$ en el intervalo $[-5, 5]$, iniciando con la llamada del Graficador:

Graficar(np.sin, -5, 5, 800, 800, 800, True, True, True, "f(x)=sin(x)")

La gráfica resultante se observa en la figura 1. Ahora, veamos el graficador con zoom, que como dijimos antes, lo manejamos como otra función que llama dentro al graficador. Este graficador con zoom es una función con la siguiente sintaxis:

GraficadorConZoom($f, a, b, n, P_x, P_y, \text{enmarcar}, \text{showGrid}, \text{showAxis}, \text{Título}$)

Donde todos los argumentos son idénticos a los del graficador. por lo que está de más explicarlos. El funcionamiento de este algoritmo es más interactivo, como se puede ver en el pseudocódigo 2.

Algoritmo 1: Graficador de f en $[a, b]$

Entrada: f, a, b, n, P_x, P_y , enmarcar, mostrar malla, mostrar ejes, título
Salida: keyInput

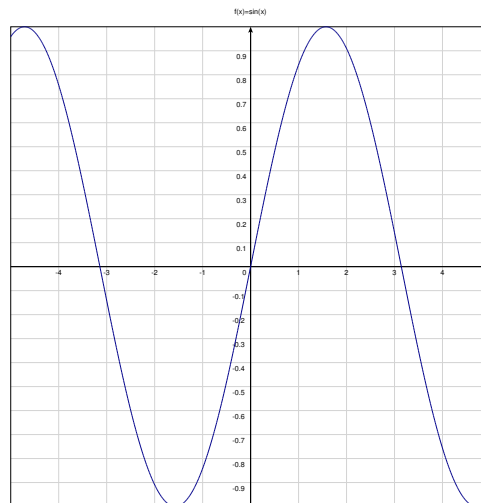
```

1 si enmarcar entonces  $w_{marco} \leftarrow 50$  en otro caso  $w_{marco} \leftarrow 0$ ;
2 para  $i = 1$  a  $n$  hacer
3    $x_i \leftarrow \frac{(b-a)i+na-b}{n-1}$ ;
4    $y_i \leftarrow f(x_i)$ ;
5    $graph_i \leftarrow \text{¿}y_i \text{ está bien definido?}$ 
6 fin
7  $y_{max} \leftarrow \max(y_i)$ ;  $y_{min} \leftarrow \min(y_i)$ ;
8  $d_y \leftarrow y_{max} - y_{min}$ ;  $m_y \leftarrow -\frac{P_y}{d_y}$ ;  $b_y \leftarrow w_{marco} - y_{max}m_y$ ;
9  $d_x \leftarrow b - a$ ;  $m_x \leftarrow \frac{P_x}{d_x}$ ;  $b_x \leftarrow w_{marco} - am_x$ ;
10  $\hat{x}_i \leftarrow m_x x_i + b_x$ ;  $\hat{y}_i \leftarrow m_y y_i + b_y$ ;
11 inicializarGráfico( $P_x + 2w_{marco}, P_y + 2w_{marco}$ );
12 si mostrar malla entonces
13    $c_x \leftarrow \lceil \log_{10} \left( \frac{d_x}{2} \right) \rceil - 1$ ;  $s_x \leftarrow 10^{c_x}$ ;
14    $c_y \leftarrow \lceil \log_{10} \left( \frac{d_y}{2} \right) \rceil - 1$ ;  $s_y \leftarrow 10^{c_y}$ ;
15   para  $j = \lfloor \frac{a}{s_x} \rfloor + 1$  a  $\lfloor \frac{b}{s_x} \rfloor$  hacer
16     Pintar línea de  $(s_x m_x j + b_x, w_{marco})$  a  $(s_x m_x j + b_x, P_y + w_{marco})$ ;
17   fin
18   para  $j = \lfloor \frac{y_{min}}{s_y} \rfloor + 1$  a  $\lfloor \frac{y_{max}}{s_y} \rfloor$  hacer
19     Pintar línea de  $(w_{marco}, s_y m_y j + b_y)$  a  $(P_x + w_{marco}, s_y m_y j + b_y)$ ;
20   fin
21 fin
22 si enmarcar entonces Pintar cuadro de  $(w_{marco}, w_{marco})$  a
    $(P_x + w_{marco}, P_y + w_{marco})$  ;
23 si mostrar ejes entonces
24   si  $y_{min} y_{max} < 0$  entonces
25     Pintar línea de  $(w_{marco}, b_y)$  a  $(P_x + w_{marco}, b_y)$ ;
26     Escribir escala de  $x$  debajo de la línea anterior;
27   en otro caso
28     Escribir escala de  $x$  debajo del marco;
29   fin
30   si  $ab < 0$  entonces
31     Pintar línea de  $(b_x, w_{marco})$  a  $(b_x, P_y + w_{marco})$ ;
32     Escribir escala de  $y$  a la izquierda de la línea anterior;
33   en otro caso
34     Escribir escala de  $y$  a la izquierda del marco;
35   fin
36 fin
37 si título no es vacío entonces Escribir título en  $(w_{marco} + \frac{P_x}{2}, \frac{w_{marco}}{2})$  ;
38 para  $i = 1$  a  $n - 1$  hacer
39   si  $graph_i$  entonces Pintar línea de  $(\hat{x}_i, \hat{y}_i)$  a  $(\hat{x}_{i+1}, \hat{y}_{i+1})$ ;
40 fin

```

Algoritmo 2: Graficador con zoom**Entrada:** f, a, b, n, P_x, P_y , enmarcar, mostrar malla, mostrar ejes, título**Salida:**

```
1  $c_x \leftarrow \frac{a+b}{2}$ ;  
2  $d_x \leftarrow b - c_x$ ;  
3 mientras True hacer  
4   Graficar( $f, a, b, n, P_x, P_y$ , enmarcar, mostrar malla, mostrar ejes, título);  
5    $\text{keyInput} \leftarrow$  Tecla presionada;  
6   seleccionar  $\text{keyInput}$  hacer  
7     caso Escape hacer  
8       break  
9     caso  $+$  hacer  
10       $c_x \leftarrow \text{float}(\text{Entrada de Usuario})$ ;  
11       $d_x \leftarrow \frac{d_x}{2}$ ;  
12     caso  $-$  hacer  
13       $c_x \leftarrow \text{float}(\text{Entrada de Usuario})$ ;  
14       $d_x \leftarrow 2d_x$ ;  
15     caso Enter hacer  
16       $c_x \leftarrow \text{float}(\text{Entrada de Usuario})$ ;  
17     fin  
18   fin  
19    $a \leftarrow c_x - d_x$ ;  
20    $b \leftarrow c_x + d_x$ ;  
21 fin
```

Figura 1: Función $y = \sin(x)$ graficada en $[-5, 5]$

Iniciamos llamando al Graficador con Zoom:

```
GraficadorConZoom(np.sin,-5,5,800,800,800,True,True,True,"f(x)=sin(x)")
```

Esto nos mostrará una gráfica idéntica a la obtenida con el graficador, sin embargo, el programa nos arroja el siguiente mensaje en la consola:

Presiona + para acercar (2X), - para alejar (0.5X), Enter para mover

Siguiendo las indicaciones de este mensaje, si hacemos clic en la gráfica, para asegurarnos estar trabajando ahí y no en otra ventana, y presionamos la tecla + en nuestro teclado, nos dará el siguiente mensaje:

Centro actual es 0.0

Si se desea un nuevo centro sobre x , se ingresa. De lo contrario, se presiona Enter para mantener el centro actual.

Por ejemplo, si elegimos hacer un acercamiento y mantener el mismo centro sobre x . tendremos la gráfica mostrada en la figura 2a.

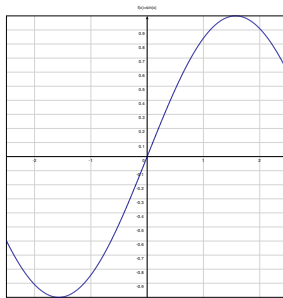
Si, en cambio, se presiona - en el teclado con la ventana de la gráfica activa, la gráfica se aleja, como se ve en la figura 2b.

Finalmente, si presionamos Enter para elegir un nuevo centro, lo hacemos de la siguiente forma:

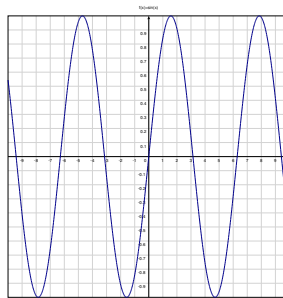
Centro actual es 0.0

Introduce nuevo centro de x: 1.5707963268

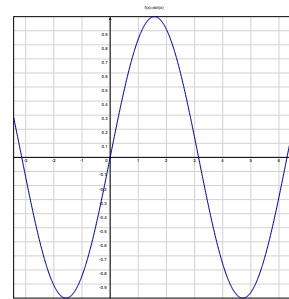
Lo que mueve el centro de la gráfica al punto $x = 1,5707963268 \approx \frac{\pi}{2}$, como se observa en la figura 2c.



(a) Ejemplo de zoom 2.0x



(b) Ejemplo de zoom 0.5x



(c) Ejemplo de desplazamiento

Figura 2: Ejemplos de uso del graficador con Zoom

3.2. Problema 2

La primera función a graficar es:

$$f(x) = x^3 + x^2 + 1 \quad (1)$$

Por lo que la definimos en Python y la graficamos:

```
def f(x):  
    a=x**3+x**2+1  
    return a  
Graficar(f,-20,20,800,800,800,True,True,True,"f(x)=x3+x2+1")
```

Ahora, la siguiente función está dada por:

$$g(x) = \sin(10x) \quad (2)$$

Igualmente, la definimos en Python y la graficamos:

```
def g(x):  
    y=np.sin(10*x)  
    return y  
Graficar(g,-3,3,1600,800,800,True,True,True,"g(x)=sen(10x)")
```

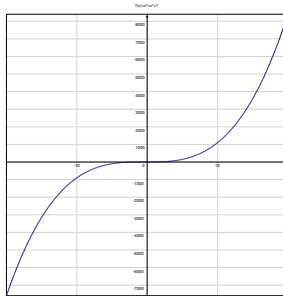
Obteniendo la gráfica de la figura 3b. Notamos que en este caso fue necesario hacer 1600 evaluaciones de la función debido a la naturaleza de la función, sin embargo esto no resulta en algún mayor problema. Finalmente, la última función está definida de la siguiente forma:

$$h(x) = \frac{1}{x+1} \quad (3)$$

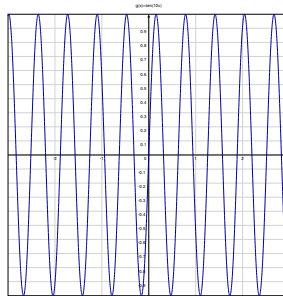
A continuación, la definimos y graficamos

```
def h(x):  
    y=1/(x+1)  
    return y  
Graficar(h,-20,20,800,800,800,True,True,True,"h(x)=1/(x+1)")
```

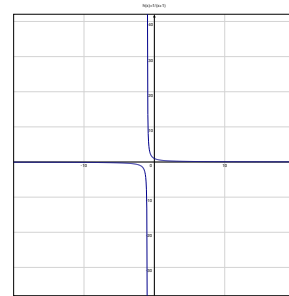
Esto resulta en la gráfica mostrada en la figura 3c.



(a) Gráfica de $f(x)$ evaluada en $[-20, 20]$



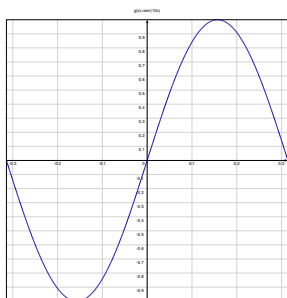
(b) Gráfica de $h(x)$ evaluada en $[-3, 3]$



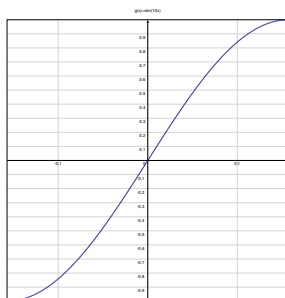
(c) Gráfica de $h(x)$ evaluada en $[-20, 20]$

3.3. Problema 3

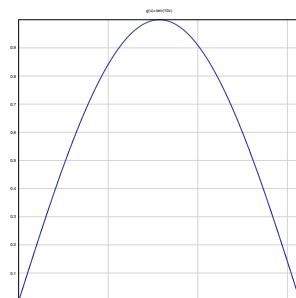
En la figura 4a se muestra el zoom en la gráfica de $g(x)$ en el punto $x = 0$, de forma que se muestra exactamente un ciclo de $g(x)$, es decir, evaluada en $[-\frac{\pi}{10}, \frac{\pi}{10}]$. Más detalladamente, la figura 4b muestra el intervalo de $g(x)$ centrado en el punto $x = 0$ en el que $g(x)$ es creciente, esto es, evaluada en $[-\frac{\pi}{20}, \frac{\pi}{20}]$. Finalmente, la figura 4c muestra el zoom en la gráfica de $g(x)$ centrada en el punto $x = \frac{\pi}{20}$, de forma que se muestra la parte cóncava de un ciclo de $g(x)$, es decir, evaluada en $[0, \frac{\pi}{10}]$.



(a) Gráfica de $g(x)$ evaluada en $[-\frac{\pi}{10}, \frac{\pi}{10}]$



(b) Gráfica de $g(x)$ evaluada en $[-\frac{\pi}{20}, \frac{\pi}{20}]$



(c) Gráfica de $g(x)$ evaluada en $[0, \frac{\pi}{10}]$

4. Conclusiones

Un graficador es fundamental en el cálculo numérico, y nuestro graficador es suficiente para graficar funciones $f : \mathbb{R} \mapsto \mathbb{R}$ que podamos evaluar numéricamente, como lo vimos en esta tarea. La función de zoom lo hace más interesante, pues incluso otros graficadores muy utilizados no tienen esa característica.

Referencias

[Zelle(2004)] J. Zelle, Python Programming, Franklin, Beedle & Associates, Wilsonville, OR, 2004.