

Tarea n°1

Integrante: Dante B. Cárcamo Ardiles
Profesor: Valentino González C.
Auxiliares: José Vines
Jou-Hui Ho

Fecha de realización: 27 de septiembre de 2018
Fecha de entrega: 27 de septiembre de 2018
Santiago, Chile

Índice de Contenidos

1. Pregunta 1	1
1.1. introducción	1
1.2. Procedimiento	1
1.3. Resultados	2
1.4. Conclusiones	2

Lista de Figuras

1.	2
------------	---

1. Pregunta 1

1.1. introducción

Se busca comparar y analizar dos métodos de estimación para la derivada de una función, en este caso $f(x) = -\cos(x)$ para $x = 1.705$ (últimos tres dígitos corresponden a los últimos tres dígitos del rut sin dígito verificador) en radianes, el primer método es el más simple con un error de orden $O(h)$:

$$f'(x) = \frac{f(x+h) - f(x)}{h} \quad (1)$$

El segundo método produce errores del orden $O(h^4)$:

$$f'(x) = \frac{-f(x+2h) + 8f(x+h) - 8f(x-h) + f(x-2h)}{12h} \quad (2)$$

En primera instancia se pide encontrar un rango apropiado de valores h a explorar para comparar la estimación numérica con el valor entregado por la función `math.sin(1,705)`, con números de tipo `float32`, finalmente se prueba con `float64` y se comparan los resultados.

Para hacer correr el programa, se debe iniciar el archivo `partel.py` en un comando, luego automáticamente muestra los gráficos correspondientes a medida de que se cierra una ventana (no se muestran simultáneamente).

1.2. Procedimiento

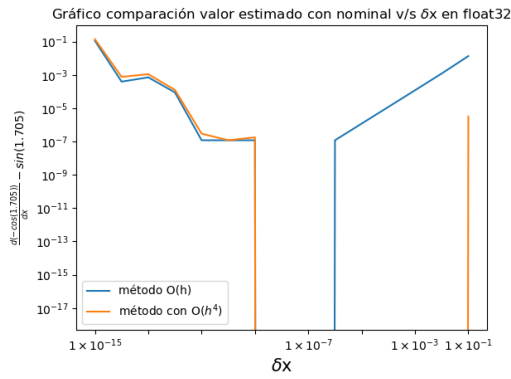
Para realizar lo pedido se crearon tres arreglos con `numpy`, uno contiene los valores de Δt o h mientras que los otros dos guardarán los valores entregados por cada método, cada uno se le definió `dtype="float32"` para así obtener todos los datos en este formato.

Para crear el arreglo de deltas se hace uso de la función `logspace` de `numpy` con inicio `-1` y final `-15` (sacados del ejemplo del profesor) con un paso `n` que es el tamaño de los arreglos, luego con un `while` se llenaron las listas `m_simple` (método más simple) y `m_h4` (método dado en el enunciado) con los respectivos valores obtenidos para cada valor de delta. Como se quiere observar la distancia respecto al valor dado por `math.sin(1.705)`, a cada lista se le resta este valor.

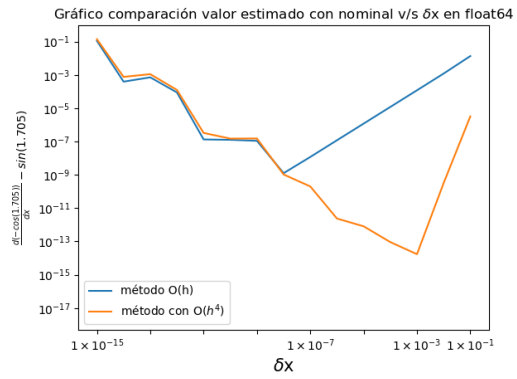
Para observar estos datos se procedió a plotear los elementos con la librería `pyplot`, plotando el valor absoluto de cada lista para obtener las distancias y usando los valores de delta para el eje `x`. El procedimiento anterior es análogo para el caso con `float64`, única diferencia es el uso de este tipo de dato. Se utilizaron escalas logarítmicas para facilidad de análisis.

finalmente se plotea la diferencia entre los datos obtenidos con `float32` y `float64`.

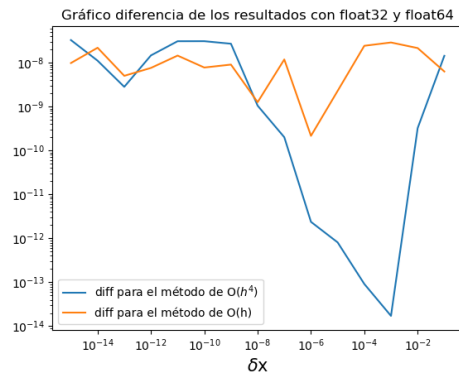
1.3. Resultados



(a) Gráfico de distancia para float32



(b) Gráfico de distancia para float64



(c) Gráfico de diferencia de valores entre float32 y float64

Figura 1:

1.4. Conclusiones