INSTITUTO TECNOLÓGICO SUPERIOR DE CHICONTEPEC

INGENIERÍA EN SISTEMAS COMPUTACIONALES

NOMBRE DE LA MATERIA:

Métodos Numéricos.

NOMBRE DEL TEMA:

Resumen de las unidades 3 y 4.

Uso de las librerías Numpy y Matplotlib.

UNIDADES:

3. Métodos De Solución De Sistemas De Ecuaciones.

4. Diferenciación e Integración Numérica.

NOMBRE DEL ALUMNO:

Miguel Angel Martinez Martinez.

NOMBRE DEL DOCENTE:

Ing. Efrén Flores Cruz.

Índice

Introducción

Numpy es un paquete fundamental para la computación científica en Python. Es una biblioteca de Python que proporciona un objeto de matriz multidimensional, varios objetos derivados (como matrices y matrices enmascaradas) y una variedad de rutinas para operaciones rápidas en matrices que incluyen matemáticas, lógica, manipulación de forma, clasificación, selección, E/S, transformada discretas de Fourier, algebra lineal básica, operaciones básicas, simulación aleatoria y muchas más.

Por otro lado, Matplotlib es una biblioteca de trazado para Python. Se utiliza junto con Numpy para proporcionar un entorno que sea una alternativa de código abierto para MatLab. También se puede usar con kits de herramienta grafica con PyQt y wxPython.

Unidad 3. Métodos de solución de sistemas de ecuaciones.

Unidad 3. Metodos de solucion de sistemas de ecuaciones.

3.1 Metodos interativos

El primer metodo iterativo para solucionar un sistema lineal aparece probablemente en una letra de Gauss a un estudiante suyo. El propuso el solucionar un sistema 4 by 4 de ecuaciones en varias ecuaciones solucionando el componente del cual la red residual es la mas grande.

La teoria de metodos iterativos inmoviles fue establecida solidamente con el trabajo de D.M. Un metodo iterativo es un metodo que progresivamente va calculando aproximaciones a la solucion de un problema. En Matematicas, en un metodo iterativo se repite un mismo proceso de mejora sobre una solucion aproximada: se espera que lo obtenido sea una solucion mas aproximada que la inicial. El proceso se repite sobre esta nueva solucion hasta que el resultado mas reciente satisface ciertos requisitos. A diferencia de los metodos directos, en los cuales se debe terminar el proceso para tener la respuesta, en los metodos iterativos se puede suspender al termino de una iteracion y se obtiene una aproximacion a la solucion.

3.2 Sistemas de ecuaciones no lineales

Un sistema de ecuaciones es no lineal, cuando al menos una de sus ecuaciones no es de primer grado.

$$x^2 + y^2 = 25$$
$$x + y = 7$$

La resolución de estos sistemas se suele hacer por el método de sustitución, para ello seguiremos los siguientes pasos:

1.- Se despeja una incógnita en una de las ecuaciones, preferentemente en la de primer grado

$y = 7 - x$

2.- Se sustituye el valor de la incógnita despejada en la otra ecuación

$x2 + (7 - x)2 = 25$

3.- Se resuelve la ecuación resultante

$x2 + 49 - 14x + x2 = 25$

$2x2 - 14x + 24 = 0$

$x2 - 7x + 12 = 0$

$$x = \frac{7 \pm \sqrt{49 - 48}}{2} = \frac{7 \pm 1}{2} = \quad x_1 = 4, \quad x_2 = 3$$

4.- Cada uno de los valores obtenidos se sustituye en la otra ecuación, se obtienen así los valores correspondientes de la otra incógnita.

$x = 3 \qquad y = 7 - 3 \qquad y = 4$

$x = 4 \qquad y = 7 - 4 \qquad y = 3$

3.3 Iteraciones y convergencia de sistemas de ecuaciones.

En general todo los procesos iterativos para resolver el sistema $Ax = b$ se recurre a una cierta matriz Q, llamada matriz descomposición, escogida de tal forma que el problema original adopte la forma equivalente

$$Qx = (Q-A)x + b$$

La ecuación sugiere un proceso iterativo que se concreta al escribir:

$$Qx^{(k)} = (Q-A)x^{(k-1)} + b \quad (k \geq 1)$$

El vector inicial $x^{(0)}$ puede ser arbitrario, aunque si se dispone de un buen candidato como solución este es el que debe emplearse. La aproximación inicial que se adopta, a no ser que se disponga de una mejor, es la idénticamente $x_1 = x_2 = \cdots = x_n = 0$ nula.

A partir de la ecuación se puede calcular una sucesión de vectores $x^1, x^2, \ldots$ Nuestro objetivo es escoger una matriz Q de manera que:

- Se pueda calcular fácilmente la sucesión $[x^{(k)}]$
- La sucesión $[x^{(k)}]$ converja rápidamente a la solución

Como en todo método iterativo, debemos especificar un criterio de convergencia δ y un número máximo de iteraciones M, para asegurar que el proceso se detiene si no se alcanza la convergencia. En este caso, puesto que x es un vector, emplearemos dos criterios de convergencia. En este caso, puesto que x es un vector, emplearemos dos criterios de convergencia que se deberán satisfacer simultáneamente:

1.- El modulo del vector diferencia $\|x^{(k)} - x^{(k-1)}\|$, partido por el modulo del vector x, $\|x^{(k)}\|$ debera ser menor que la convergencia deseada:

$$ABS\left(\frac{\|x^{(k)} - x^{(k-1)}\|}{\|x^{(k)}\|}\right) \leq \delta$$

2.- La diferencia relativa del mayor elemento en valor absoluto del vector $x^{(k)}$, $x_m = Max\{x_i\}$, debera ser diez veces menor que δ.

$$ABS\left(\frac{x_m^{(k)} - x_m^{(k-1)}}{x_m^{(k)}}\right) \leq \frac{\delta}{10}$$

Unidad 4. Diferenciación e integración numérica

Unidad 4. Diferenciación e integración numérica

4.1 Diferenciación numérica.

El cálculo de la derivada de una función puede hacer un proceso "difícil" y sea por lo complicado de la definición analítica de la función o porque esta se conoce únicamente en un número discreto de puntos. Este es el caso si la función representa el resultado de algún experimento. En esta lección.

Fórmulas para la primera derivada:

La definición de la derivada de una función $f(x)$ en el punto "x" está dada en ~~términos~~ en el punto "x" está dado en términos del límite:

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

$$D_h f(x) = \frac{f(x+h) - f(x)}{h}$$

Antes de ver algunos ejemplos donde usamos esta fórmula, tendríamos que contestar la pregunta de ¿Cuándo bueno es esta aproximación de la derivada? Por el teorema de Taylor sabemos que:

$$f'(x) = D_h f(x) - \frac{h}{2} f''(\xi_x)$$

Esta fórmula nos dice que aproximadamente a $f'(x)$ con un error ~~parcial~~ proporcional a "h", i.e., $O(h)$. Fórmula para la segunda derivada. El proceso de arriba se puede usar para obtener fórmulas para la derivada. Usando el teorema de Taylor podemos escribir las expansiones.

4.2 Integracion numerica.

En analisis numerico la integracion numerica constituye una amplia gama de algoritmos para calcular el valor numerico de una integral definida y, por extension, el termino se usa a veces para describir algoritmos numericos para resolver ecuaciones diferenciales. El termino cuadratura numerica (a menudo abreviado a cuadratura) es mas o menos sinonimo de integracion numerica, especialmente si se aplica a integrales de una dimension a pesar de que para el caso de dos o mas dimensiones (integrales multiples) tambien se utiliza.

El problema basico considerado por la integracion numerica es calcular una solucion aproximada a la integral definida:

$$\int_a^b f(x)\,dx$$

Este problema tambien puede ser enunciado como un problema de valor inicial para una ecuacion diferencial ordinaria, como sigue:

$$y'(x) = f(x), \quad y(a) = 0$$

Encontrar $y(b)$ es equivalente a calcular la integral. Los metodos desarrollados para ecuaciones diferenciales ordinarias, como el metodo de Runge-Kutta pueden ser aplicados al problema reformulado.

4.3 Integracion Multiple.

Las integrales multiples se utilizan a menudo en la ingenieria. Por ejemplo; una ecuacion general para calcular el promedio de una funcion bidimensional puede describirse como sigue:

$$\bar{f} = \frac{\int_c^b \left(\int_a^b (f(x,y)dx)\right) dy}{(d-c)(b-a)}$$

Al numerador se le llama integral doble.

Las tecnicas estudiadas en ... se utilizan para evaluar integrales multiples. Un ejemplo sencillo seria obtener la integral doble de una funcion sobre un area rectangular.

Recuerde del calculo de dichas integrales se pueden calcular como integrales iteradas

$$\int_a^b \left(\int_c^b f(x,y)dx\right) dy = \int_a^d \left(\int_a^b (x,y)\right) dx$$

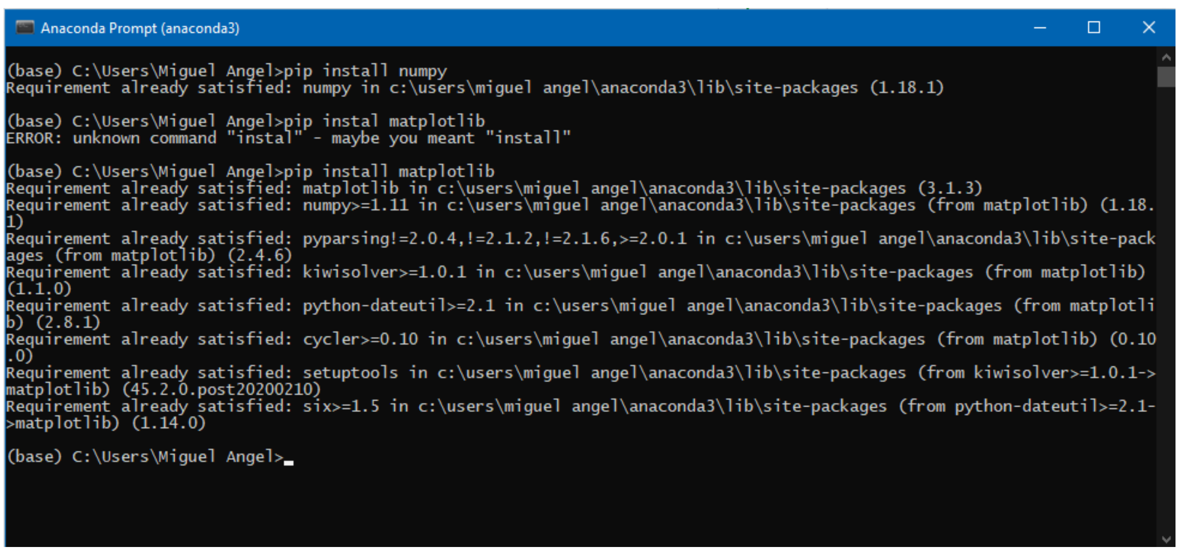
Primero se evalua la integral de una de las dimensiones y el resultado de esta primera integracion se incorpora en la segunda integracion. Una integral numerica doble estaria basada en la misma idea. Primero se aplican metodos, como la regla de Simpson o del trapecio para segmentos multiples, a la primera dimension manteniendo constantes los valores de la segunda dimension.

Librerías Numpy y Matplotlib

Para poder instalar las librerias en Python, lo podemos hacer mediante un IDE Anaconda que de tal forma podemos trabakar con Python, que Anaconda nos ofrece una variedad de herramientas y donde podemos descargar las librerias. Donde para eso nos tendremos que dirigirnos a la terminal de **Ancaconda Prompt (Anaconda3),** donde tendremos que escribir un par de codigos diciendole que quereos descargar las librerias que queremos descargar:

- Para eso escribimos en la terminal:
    - **(base) C:\Users\ Miguel Angel\>pip install numpy**

Le daremos Enter para que comience con las descarga de la librería Numpy. No demorara mucho.

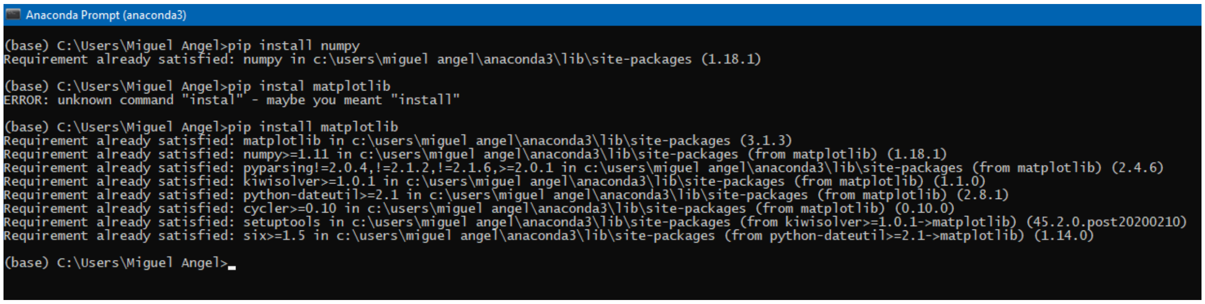


```
Anaconda Prompt (anaconda3)

(base) C:\Users\Miguel Angel>pip install numpy
Requirement already satisfied: numpy in c:\users\miguel angel\anaconda3\lib\site-packages (1.18.1)

(base) C:\Users\Miguel Angel>pip instal matplotlib
ERROR: unknown command "instal" - maybe you meant "install"

(base) C:\Users\Miguel Angel>pip install matplotlib
Requirement already satisfied: matplotlib in c:\users\miguel angel\anaconda3\lib\site-packages (3.1.3)
Requirement already satisfied: numpy>=1.11 in c:\users\miguel angel\anaconda3\lib\site-packages (from matplotlib) (1.18.
1)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in c:\users\miguel angel\anaconda3\lib\site-pack
ages (from matplotlib) (2.4.6)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\miguel angel\anaconda3\lib\site-packages (from matplotlib)
(1.1.0)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\miguel angel\anaconda3\lib\site-packages (from matplotli
b) (2.8.1)
Requirement already satisfied: cycler>=0.10 in c:\users\miguel angel\anaconda3\lib\site-packages (from matplotlib) (0.10
.0)
Requirement already satisfied: setuptools in c:\users\miguel angel\anaconda3\lib\site-packages (from kiwisolver>=1.0.1->
matplotlib) (45.2.0.post20200210)
Requirement already satisfied: six>=1.5 in c:\users\miguel angel\anaconda3\lib\site-packages (from python-dateutil>=2.1-
>matplotlib) (1.14.0)

(base) C:\Users\Miguel Angel>_
```

Imagen 1. Descarga de la librería Numpy para el lenguaje Python.

Para la segunda librería, haremos de la misma forma como lo hicimos con la primera librería, de tal forma haremos lo siguiente:

- Para eso escribimos en la terminal:
    - **(base) C:\Users\ Miguel Angel\>pip install matplotlib**



```
Anaconda Prompt (anaconda3)

(base) C:\Users\Miguel Angel>pip install numpy
Requirement already satisfied: numpy in c:\users\miguel angel\anaconda3\lib\site-packages (1.18.1)

(base) C:\Users\Miguel Angel>pip instal matplotlib
ERROR: unknown command "instal" - maybe you meant "install"

(base) C:\Users\Miguel Angel>pip install matplotlib
Requirement already satisfied: matplotlib in c:\users\miguel angel\anaconda3\lib\site-packages (3.1.3)
Requirement already satisfied: numpy>=1.11 in c:\users\miguel angel\anaconda3\lib\site-packages (from matplotlib) (1.18.1)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in c:\users\miguel angel\anaconda3\lib\site-packages (from matplotlib) (2.4.6)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\miguel angel\anaconda3\lib\site-packages (from matplotlib) (1.1.0)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\miguel angel\anaconda3\lib\site-packages (from matplotlib) (2.8.1)
Requirement already satisfied: cycler>=0.10 in c:\users\miguel angel\anaconda3\lib\site-packages (from matplotlib) (0.10.0)
Requirement already satisfied: setuptools in c:\users\miguel angel\anaconda3\lib\site-packages (from kiwisolver>=1.0.1->matplotlib) (45.2.0.post20200210)
Requirement already satisfied: six>=1.5 in c:\users\miguel angel\anaconda3\lib\site-packages (from python-dateutil>=2.1->matplotlib) (1.14.0)

(base) C:\Users\Miguel Angel>
```

Imagen 2. descarga de la segunda librería Matplotlib.

Ejemplos del uso de las librerías Numpy y Matplotlib

Para comprobar las funciones de las librerías que instalamos, haremos unos pequeños ejemplos:

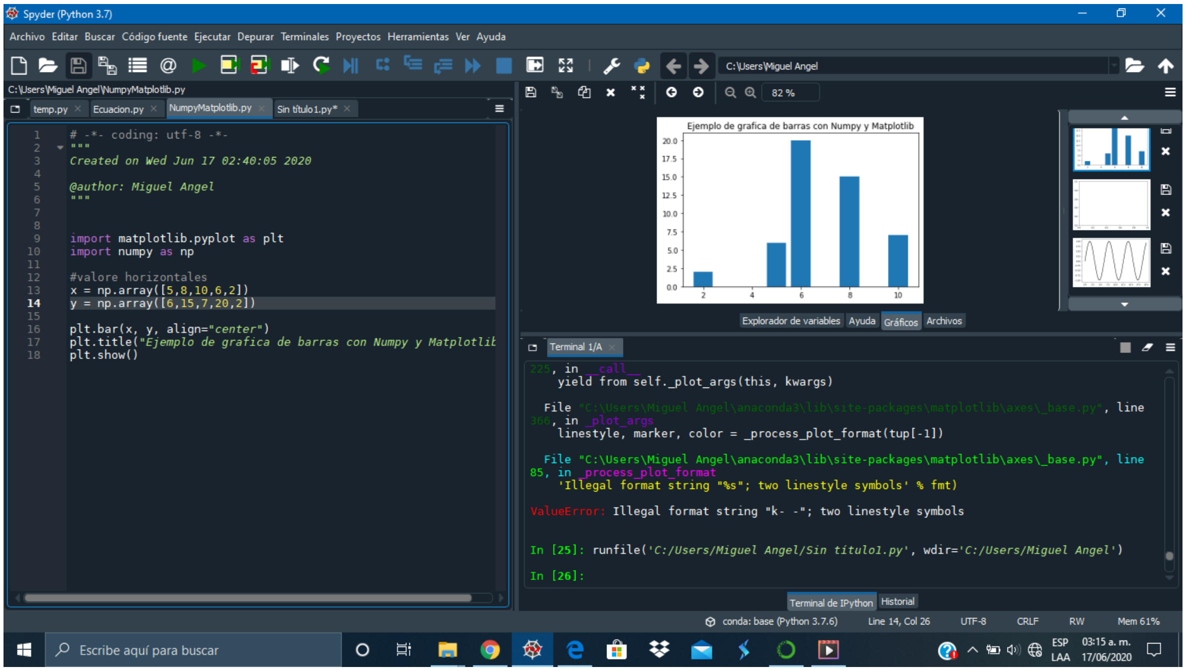


Imagen 3. Primer ejemplo de las funciones de las librerías.

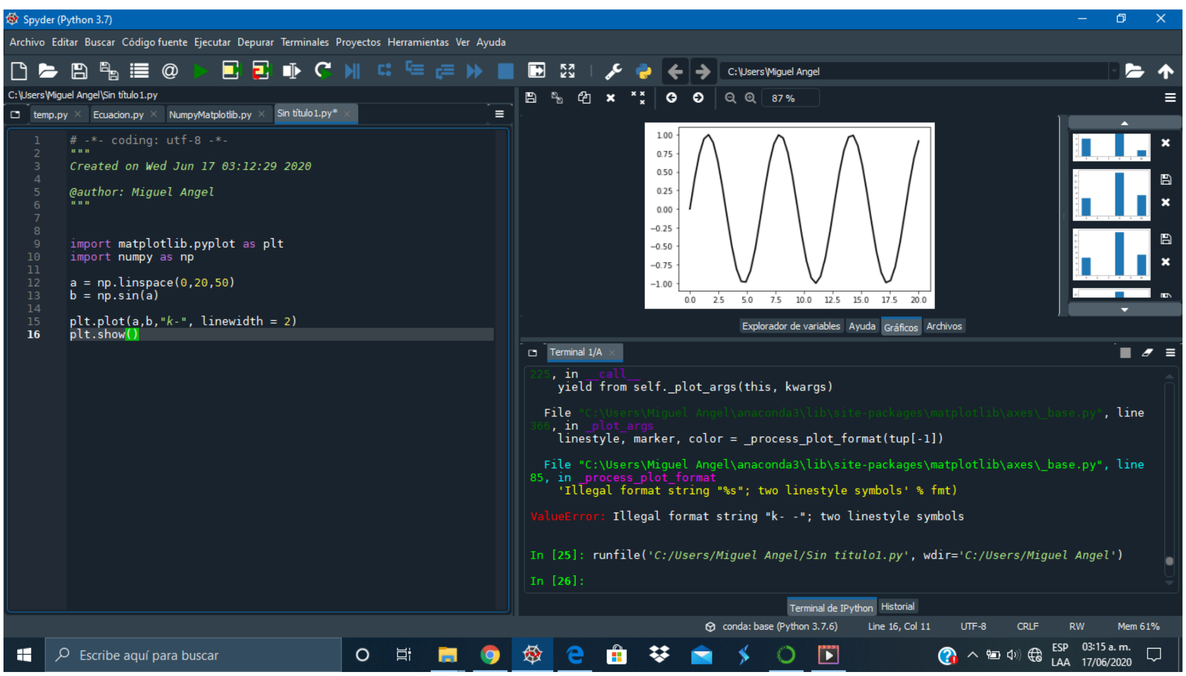


Imagen 4. Segundo ejemplo de las librerias.

Conclusión

Las librerías Numpy y Matplotlib son una de las herramientas que ocupa Python para realizar graficas por medio de código, donde hay diferentes tipos de graficas que podemos realizar, ademad de que también podemos realizar dibujos, pero es programación más avanzada. De tal forma que podemos jugar con las librerías, pero también tendremos que saber que tipo de grafica queremos realizar, porque dichas graficas tienen diferentes funciones tal como vimos con loe ejemplo. Aparte tendríamos que saber como ocuparlas, ya que se necesitan importarlas para poder trabajar con ellas.

Donde primero hay que descargar las librerías para poder usarlas, ya que al principio no contamos con dichas librerías, hay formas para descargarlos, uno es descargarlos para Windows y la segunda es por medio de Anaconda.

Bibliografía

Metodos Numericos. (s.f.). Recuperado el 24 de Junio de 2020, de Unidad 3: http://itpn.mx/recursosisc/4semestre/metodosnumericos/Unidad%20III.pdf

Metodos Numericos. (s.f.). Recuperado el 24 de Junio de 2020, de Unidad 4: http://itpn.mx/recursosisc/4semestre/metodosnumericos/Unidad%20IV.pdf