**Estudiantes**
Miguel Angel Avila Santos
Juan Andres Martinez Amado
Jorge Luis Esposito Albornoz
Juan Sebastian Herrera Guaitero

**Asignatura**
Análisis numérico

**Profesor**
Eddy Herrera Daza

Taller 2

21 de septiembre de 2021

3) Suponga que en el siguiente modelo $f(x)$ describe la cantidad de personas que son afectadas por un virus, en donde t es el tiempo en días $f(t) = k1t + k2t\ 2 + k3e\ 0.15t$ Se conocen los siguientes datos: $f(10) = 25$; $f(15) = 130$; $f(20) = 650$ Determine de forma aproximada el día más cercano donde la cantidad de personas infectadas supera los 1500; 1800; 2000.

Con los resultados dados, se reemplaza la ecuación en términos de T=10,T=15 y T=20 igualando la ecuación a los respectivos resultados dados 25,130 y 650. Con lo anterior se determinó un sistema de ecuaciones para poder obtener las incógnitas k1,k2 y k3.

$$\begin{bmatrix} 10a + 100b + ce^{0.15(10)} = 25 \\ 15a + 225b + ce^{0.15(15)} = 130 \\ 20a + 400b + ce^{0.15(20)} = 650 \end{bmatrix}$$

a =k1; b=k2; c=k3

Resolvemos la matriz 3x3 en con la librería numpy obtenemos el siguiente resultado:

```
[-17.32509437  -2.24216837  94.26530355]
```

Resultado implementación del grupo:

```
[[-20.98571752]
 [ -0.63594229]
 [ 65.9226225 ]]
[[-20.68504673]
 [ -0.82302545]
 [ 69.34896096]]
[[-20.3497935 ]

show more (open the raw output data in a text editor) ...

 [ -2.24216837]
 [ 94.26530355]]
[[-17.32509437]
 [ -2.24216837]
 [ 94.26530355]]
```

Donde obtuvimos como solución:

K1=-17.32509437
K2=2.24216837
K3=94.26530355

Función en términos de T

$F(t) =- 17.32509437t - 2.24216837t^(2) + 94.26530355e^(0.15t)$

Cantidad de personas infectadas:

1500 -> Dia mas cercano 24(Aproximado hacia arriba) con un total de 1742.65243322
1800 -> Dia mas cercano 25(Aproximado) con un total de 2173.78011151
2000 ->Dia mas cercano 25(Aproximado) con un total de 2173.78011151

Los días fueron aproximados, ya que cuando hablamos de días se maneja como un número entero

6) Dado el siguiente sistema:

$2x - z = 1$

$\boldsymbol{\beta}x + 2y - z = 2$

$-x + y + \boldsymbol{\alpha}z = 1$

a. Encuentre el valor de **α** y **β** para asegura la convergencia por el método de Jacobi

Si una matriz es diagonalmente dominante, entonces es convergente, para los valores de alpha y beta utilizando el siguiente análisis:

Para la primera fila 2 > |0| + |-1|
Para la segunda fila 2 > |B| + |-1|
Para la tercera fila A > |-1| + |1|

Por lo tanto se necesita Beta = 0 y Alpha > 2 para garantizar convergencia, de tal forma que Beta = 0 y Alpha = 3

b. Genere una tabla que tenga 10 iteraciones del método de Jacobi con vector inicial x{0}=[1,2,3]^t

| Iteración 0 | Vector Inicial: [1,2,3] | Error |
|---|---|---|
| Iteración 1 | Vector Actualizado: [2,2.5,0] | 3 |
| Iteración 2 | Vector Actualizado: [0.5,1,0.16666667] | 1.5 |
| Iteración 3 | Vector Actualizado: [0.58333333,1.08333333,0.16666667] | 0.08333333333333337 |
| Iteración 4 | Vector Actualizado: [0.58333333,1.08333333,0.16666667] | 8.32667268468867*10**-17 |
| Iteración 5 | Vector Actualizado: [0.58333333,1.08333333,0.16666667] | 2.220446049250313*10**-16 |
| Iteración 6 | Vector Actualizado: [0.58333333,1.08333333,0.16666667] | 8.32667268468867*10**-17 |
| Iteración 7 | Vector Actualizado: [0.58333333,1.08333333,0.16666667] | 2.220446049250313*10**-16 |
| Iteración 8 | Vector Actualizado: [0.58333333,1.08333333,0.16666667] | 8.32667268468867*10**-17 |
| Iteración 9 | Vector Actualizado: [0.58333333,1.08333333,0.16666667] | 2.220446049250313*10**-16 |
| Iteración 10 | Vector Actualizado: [0.58333333,1.08333333,0.16666667] | 8.32667268468867*10**-17 |

7) Dada la matriz A (del punto 1) verificar si:

$$u - 8v - 2w = 1$$
$$u + v + 5w = 4$$
$$3u - v + w = -2$$

i. Se puede descomponer de la forma LU, entonces utilice el resultado para resolver el sistema, teniendo en cuenta que la máquina admite cuatro dígitos significativos; ¿cómo afecta esto la respuesta?

Si se puede descomponer de la Forma LU, ya que cumple sus condiciones de no tener en su diagonal osea en el orden k-ésimo valores distintos a 0. EL resultado de la descomposición y solución de la matriz es:

```
Matriz L :
 [[1.     0.     0.    ]
 [1.     1.     0.    ]
 [3.     2.5556 1.    ]]
 Matriz U :
 [[  1.      -8.      -2.    ]
 [  0.       9.       7.    ]
 [  0.       0.     -10.8892]]
La solucion del sistema es = ['-1.2448', '-0.5714', '1.1632']
```

Al tener una precisión de 1^-4 esto no afecta el resultado obtenido ya que se utiliza la fórmula $Ax = LUx = b.$ dando por hecho que b es el vector solución de A. Siguiendo el primer paso $Ly = b$ para $y.$ y el segundo paso $Ux = y$ para $x.$ que al terminar da la solución del sistema

ii. Se puede descomponer utilizando el método de Cholesky?, entonces utilice el resultado para resolver el sistema y encuentre el número de operaciones que se necesita para aplicar el algoritmo.

A no se puede descomponer por el método Cholesky ya que la matriz no cumple con las condiciones para implementar este que son:

- Una matriz hermitiana es decir una matriz cuadrada de elementos complejos que tiene la característica de ser igual a su propia traspuesta conjugada. Es decir, el elemento en la i-ésima fila y j-ésima columna es igual al conjugado del elemento en la j-ésima fila e i-ésima columna, para todos los índices i y j
- Matriz definida positiva es una matriz hermitiana que en muchos aspectos es similar a un número real positivo, también puede tratarse de una matriz simétrica real cuyos menores principales son positivos

```
No se cumplen las condiciones de ser una matriz Herminitiana y definida positiva para la descomposicion de Cholesky
```

10) Dado un sistema de ecuaciones no lineales, implemente el método de Newton Multivariado (es decir para varias variables) para resolver el problema:
Determinar numéricamente la intersección entre la circunferencia $x^2 + y^2 = 1$ y la recta $y = x$.
Usamos la aproximación inicial (1,1).

Para encontrar la intersección de la circunferencia y la recta se resolvió el sistema de ecuaciones no lineal formado por las ecuaciones de ambas:

$$x^2 + y^2 - 1 = 0$$
$$y - x = 0$$

Utilizando el método de Newton Multivariado nos aproximamos a la solución del sistema no lineal dado por una aproximación inicial $x$ la cual como bien nos dice el enunciado será (1,1).
Ya que el problema no nos da un máximo de iteraciones en la implementación se definió 100 como el límite. Dentro de cada iteración, primero se calcula la matriz Jacobiana la cual está compuesta por las derivadas parciales de las $fi$ respecto a las variables $xj$, implementado dentro del código realizado. Posteriormente, resolvemos el sistema que está dado por la matriz Jacobiana multiplicada por el vector $y$ e igualado por menos el vector $f$, resolvemos este sistema con eliminación de Gauss con renglón pivote escalado, el cual está igualmente implementado en el método de Newton Multivariado utilizado, y lo igualamos a la variable $y$. Después, en base al vector $y$ resultado del sistema, se reescribe la aproximación inicial igualando el $x$ a $x$ más $y$. Luego, verificamos si la magnitud, calculada con ayuda de las funciones sqrt de la librería math y de la función dot de la librería numpy, es menor a la tolerancia, definida como 10E-8, si este es el caso se retorna la aproximación actual y finaliza el programa. Finalmente, si se excede el número de iteraciones dadas, finaliza el programa con un aviso de lo sucedido.

Resultados obtenidos:

```
Iteracion 0     [x,y]: [1. 1.]      Magnitud:  0.35355339059327373
Iteracion 1     [x,y]: [0.7500125 0.7500125]       Magnitud:  0.0589383305532884
Iteracion 2     [x,y]: [0.70833681 0.70833681]     Magnitud:  0.0017378850042399799
Iteracion 3     [x,y]: [0.70710794 0.70710794]     Magnitud:  1.6328910409753269e-06
Iteracion 4     [x,y]: [0.70710678 0.70710678]     Magnitud:  1.1678764813220112e-10
```

```
La interseccion del problema dado utilizando el metodo de Newton Multivariado y la aproximacion incial (1,1) es:
[0.70710678 0.70710678]
```