UNIVERSIDAD NACIONAL DE COLOMBIA
SEDE MEDELLÍN

MÉTODOS NUMÉRICOS 3006907
ALGO DE MATLAB, SEMESTRE 01–2019

GRAFICAS - RAICES - PUNTOS FIJOS – MATLAB

Considere la función

$$h(x) = \frac{1}{10}\cos(2x)(e^x - \tan^{-1}(x)) + e^{-x}$$

1. ¿Cuántos **ceros** tiene la función h en el intervalo $[-2,6]$?

   Para graficar podemos usar la instrucción `fplot` en la ventana de comandos de MATLAB, así:

```
>> h=@(x) 0.1*cos(2*x).*(exp(x)-atan(x))+exp(-x)
>> fplot(h,[-2,6])
>> grid on
>> axis([-2 6 -1 1])
```

   También se puede graficar, usando puntos

```
>> x = linspace(-2,6,100)
>> y = 0.1*cos(2*x).*(exp(x)-atan(x))+exp(-x)
>> plot(x,y)
```

   De la gráfica tenemos que h tiene 4 ceros en $[-2,6]$.

2. Aproximar el primer cero positivo de la función h empleando el método de disección hasta que se satisfaga una tolerancia `delta=1e-8`.

   De la gráfica se observa que el primer cero positivo de h esta en el intervalo $[1.2,1.4]$, usamos la rutina `bisect` así:

```
>> [c, err, yc] = bisect (h, 1.2, 1.4, 1e-8)
c = 1.343800488114357
err = 5.960464344312300e-09
yc = 1.688676254385513e-09
```

   el primer cero positivo de h en $[-2,6]$ es 1.343800488114357.

3. Aproximar el cuarto cero positivo de la función h empleando el método de Newton hasta que se satisfaga una tolerancia `delta=1e-10` o `epsilon=1e-10` .

   De la gráfica se observa que el cuarto cero positivo de h esta en el intervalo $[5,6]$, además, debemos definir la función derivada de h

$$h'(x) = \frac{1}{10}\left[\cos(2x)\left(e^x - \frac{1}{x^2+1}\right) - 2\,\text{sen}(2x)(e^x - \tan^{-1}(x))\right] - e^{-x}$$

   para poder emplear la rutina `newton`, así:

```
>> hp=@(x) 0.1*(cos(2*x).*(exp(x)-1./(x.^2+1))-2*sin(2*x).*(exp(x)-atan(x)))-exp(-x)
>> [p0, err, k, y] = newton (h, hp, 5.5, 1e-10, 1e-10,50)
p0 = 5.497702770021194
err = 2.804512178045115e-11
k = 3
y = 6.323067069935462e-15
```

   el cuarto cero positivo de h en $[-2,6]$ es 5.497702770021194, el cual se obtiene en la tercera iteración de Newton.

4. ¿Cuantos **puntos fijos** tiene la función h en el intervalo $[-2,6]$?

   Para identificar los puntos fijos de h, graficamos en la misma `figure` la gráfica de g y la recta $y = x$, así

```
>> fplot(h,[-2,6])
>> hold on
>> fplot(@(x)x,[-2,6])
```

la función h tiene 2 puntos fijos en $[-2,6]$.

5. Aproximar el primer punto fijo positivo de la función h empleando el método de punto fijo hasta que se satisfaga una tolerancia `delta=1e-8`.

   De la gráfica vemos que el primer punto fijo positivo de h esta en el intervalo $[0,1]$, utilizamos la rutina `fixpt`, así

```
>> [p, k, err, P] = fixpt (h, 0.5, 1e-8, 100)
p = 0.597286118665076
k = 60
err = 8.648844196379457e-09
P = ...
```

   el punto fijo de h en $[0,1]$ es 0.597286118665076.

SISTEMAS DE ECUACIONES LINEALES Y NO LINEALES – MATLAB

$\boxed{\text{Lineales}}$ Considere el sistema de ecuaciones lineales $\mathbf{A}\mathbf{x} = \mathbf{b}$ donde

$$\mathbf{A} = \begin{bmatrix} 3 & -1 & 0 & 0 & 0 \\ -1 & 5 & 4 & 0 & 0 \\ 0 & 4 & 6 & -2 & 0 \\ 0 & 0 & -2 & 8 & 1 \\ 0 & 0 & 0 & 1 & 11 \end{bmatrix}, \qquad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \qquad \text{y} \qquad \mathbf{b} = \begin{bmatrix} 3 \\ 0 \\ 1 \\ -1 \\ 3 \end{bmatrix}.$$

Analizar la convergencia de las sucesiones generadas por los métodos de Jacobi, Gauss-Seidel y SOR con parámetro $\omega \in (0,2)$.
Notemos de $\mathbf{A}$ es matriz simétrica, veamos como son sus valores propios

```
>> A=[3 -1 0 0 0;-1 5 4 0 0;0 4 6 -2 0;0 0 -2 8 1;0 0 0 1 11]
>> eig(A)
0.9435  3.1612  7.1747  10.2484  11.4721
```

todos son positivos, luego por teorema $\mathbf{A}$ es definida positiva (hasta acá podemos concluir convergencia de SOR para $\omega \in (0,2)$, en particular converge Gauss-Seidel). Además, $\mathbf{A}$ es tridiagonal, así que podemos concluir también convergencia de Jacobi.
En este caso podemos calcular la elección óptima de ω_{opt}, para ello necesitamos $\rho(\mathbf{T}_{\mathrm{J}})$. Hallemos la descomposición $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$ y así obtener $\mathbf{T}_{\mathrm{J}} = \mathbf{D}^{-1}(\mathbf{L}+\mathbf{U})$ y $\rho(\mathbf{T}_{\mathrm{J}})$

```
>> D=diag(diag(A))
>> L=triu(A)-A
>> U=tril(A)-A
>> Tj=inv(D)*(L+U)
>> radioTj=max(abs(eig(Tj)))
```

como $\rho(\mathbf{T}_{\mathrm{J}}) = \texttt{radioTj} = 0.8224$ entonces $\omega_{opt} = \dfrac{2}{1+\sqrt{1-[\rho(\mathbf{T}_{\mathrm{J}})]^2}} = \texttt{wopt} = 1.2748$ y podemos aproximar la solución del sistema usando la rutina `sor`, para ello supongamos una aproximación inicial $\mathbf{x}^{(0)} = [1,2,3,4,5]^T$

```
>> wopt=2/(1+sqrt(1-radioTj^2))
>> b=[3 0 1 -1 3]'
>> x0=[1 2 3 4 5]'
>> Y = sor (A,b,x0,wopt,1e-8,200)
```

necesito sólo 20 iteraciones, veamos que tan cerca estamos de la solución: calculemos la norma uno del residual

```
>> r=A*Y-b
>> norm(r,1)
```

$\|r\|_1 = 1.2014e-08$ estamos cerca!. Veamos cuántas iteraciones necesitan Jacobi y Gauss-Seidel

```
>> X = jacobi (A, b, x0, 1e-8, 200)
>> Z = gseid (A, b, x0, 1e-8, 200)
```

101 y 50 iteraciones, respectivamente. Calcule la norma uno de los residuales ¿cuál esta mas cerca?
Finalmente, veamos que se cumple lo que dice el teorema $\rho(Tgs) = [\rho(Tj)]^2$ y $\rho(T_{\omega_{opt}}) = \omega_{opt} - 1$

```
>> Tgs=inv(D-L)*U
>> radioTgs=max(abs(eig(Tgs)))
>> Tw=inv(D-wopt*L)*((1-wopt)*D+wopt*U)
>> radioTw=max(abs(eig(Tw)))
```

$\boxed{\text{No lineales}}$ Considere el sistema no lineal

$$\begin{aligned} \operatorname{sen}(x+y) - e^{x-y} &= 0 \\ \cos(x+6) - x^2y^2 &= -1 \end{aligned}$$

★ ¿Cuántas soluciones tiene este sistema en la región $[-4,4] \times [-4,4]$.

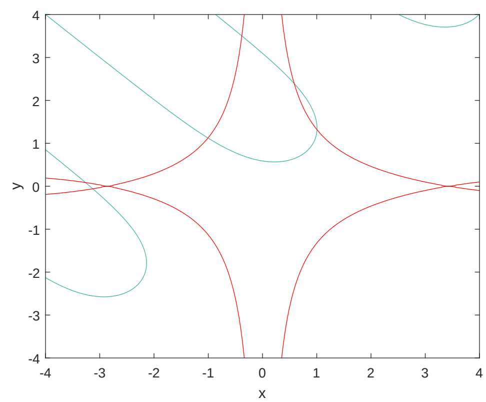
Para graficar funciones de dos variables podemos usar la instrucción `ezplot`

```
>> ezplot('sin(x+y)-exp(x-y)',[-4 4 -4 4])
>> hold on
>> ezplot('cos(x+6)-x^2*y^2+1',[-4 4 -4 4])
```

si queremos modificar el color de alguna de las funciones utilizamos las siguientes instrucciones

```
>> hh=ezplot('cos(x+6)-x^2*y^2+1',[-4 4 -4 4]);
>> set(hh,'color',[1 0 0])
```

el sistema tiene 6 soluciones



★ Aproximar las soluciones empleando el método de Newton hasta que se satisfaga una condición de error con `delta=1e-8` o `epsilon=1e-8`.
Para aproximar las soluciones definimos el campo vectorial $\mathbf{F}$ de tal forma que el sistema no lineal se pueda escribir de la forma $\mathbf{F}(\mathbf{X}) = 0$

$$\mathbf{F}(\mathbf{X}) := \mathbf{F}\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) = \begin{bmatrix} \operatorname{sen}(x_1+x_2) - e^{x_1-x_2} \\ \cos(x_1+6) - x_1^2x_2^2 + 1 \end{bmatrix}$$

y calculamos la matriz Jacobiana asociada a $\mathbf{F}$

$$\mathbf{JF}(\mathbf{X}) = \begin{bmatrix} \cos(x_1+x_2) - e^{x_1-x_2} & \cos(x_1+x_2) + e^{x_1-x_2} \\ -\operatorname{sen}(x_1+6) - 2x_1x_2^2 & -2x_1^2x_2 \end{bmatrix}.$$

Ahora vamos a definir a $\mathbf{F}$ y $\mathbf{J}$ en MATLAB

```
>> F=@(x) [sin(x(1)+x(2))-exp(x(1)-x(2)) cos(x(1)+6)-x(1)^2*x(2)^2+1]
>> J=@(x) [cos(x(1)+x(2))-exp(x(1)-x(2)) cos(x(1)+x(2))+exp(x(1)-x(2)); ...
           -sin(x(1)+6)-2*x(1)*x(2)^2 -2*x(1)^2*x(2)]
```

para aproximar las soluciones, consideremos por ejemplo $x^{(0)} = [1 -1]^T$

```
>> [P, iter, err] = newdim (F, J, [1 -1], 1e-8, 1e-8, 100)
P = 1.0042    1.3177
iter = 7
>> F(P)
1.0e-11 * [-0.0217  -0.1156]
```

Calcular las otras soluciones ...