Universidad Nacional de San Juan

Facultad de Ingeniería

MÉTODOS NUMÉRICOS

Ingeniería CIVIL

Equipo de Cátedra

Profesor Titular	Prof. Beatriz Morales
Profesor Asociado	Ing. Agustina Garcés
Profesor Adjunto	Ing. Marión Castro.
Jefe de Trabajos Prácticos	Ing. Pablo Marcuzzi

Año 2020

(Segunda clase de Ecuaciones No Lineales)

SISTEMAS DE ECUACIONES NO LINEALES

Se llama Sistema de Ecuaciones No lineales cuando tenemos más de una ecuación no lineal, o sea n-ecuaciones no lineales con n- incógnitas. La solución será la n-upla $(x_1, x_2, \dots, x_n)$ que verifica todas y cada una de las ecuaciones del sistema

$$\begin{cases} f_1(x_1, x_2, \dots, x_n) = 0 \\ f_2(x_1, x_2, \dots, x_n) = 0 \\ \dots \dots \dots \dots \dots \dots \dots \dots . \\ f_n(x_1, x_2, \dots, x_n) = 0 \end{cases}$$

Expresado en forma compacta (vectorial) será

$$\vec{F}(\vec{x}) = \vec{0}$$

Donde

$$\vec{F} = \begin{bmatrix} f_1 \\ f_2 \\ \dots . \\ f_n \end{bmatrix} \qquad \qquad \vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \dots . \\ x_n \end{bmatrix}$$

Ejemplo1: sea el sistema

$$\begin{cases} x^2 + y^2 - 4 = 0 & \longrightarrow \quad f_1(x, y) = 0 \\ e^x - y = 0 & \longrightarrow \quad f_2(x, y) = 0 \end{cases}$$

Estudiaremos dos métodos:

- Método Gráfico
- Método de Newton Raphson

MÉTODO GRÁFICO

Consiste en graficar cada una de las funciones en una misma ventana gráfica, y la solución será los valores del vector $\vec{x}$ que verifican todas las ecuaciones simultáneamente, que gráficamente significa el punto que es tienen en común (intersección) las gráficas.

Este método solo podemos aplicarlo para sistemas de dos ecuaciones con dos incógnitas, de allí que su uso es muy limitado.

Veamos el Ejemplo 1. Lo hacemos aplicando Matlab u Octave, para ello se debe graficar cada una de las ecuaciones.

$$\begin{cases} x^2 + y^2 - 4 = 0 & \longrightarrow \quad x^2 + y^2 = 4 \\ e^x - y = 0 & \longrightarrow \quad y = e^x \end{cases}$$

¿Se puede despejar la variable “**y**”de cada una de las ecuaciones y graficarlas-?

En el caso de la circunferencia, al despejar la variable "y" $y = \pm\sqrt{4 - x^2}$ por el doble signo de la raíz, no se puede expresar en una **sola función,** entonces conviene trabajarlo de otra manera.

Por lo general en las ecuaciones no lineales no se puede despejar una variable en función de la otra, entonces la manera de trabajar es graficar las curvas de nivel $z = 0$, correspondientes a las superficie, $z_i = f_i(x, y)$

Para graficar funciones de dos variables (En 3D) es necesario definir una grilla de puntos del dominio, en una matriz, para luego calcular el valor de la función en cada uno de dichos puntos. Esto se hace con los siguientes comandos:

- ✓ Comando meshgrid
  Genera los puntos del dominio donde se evaluará la función. Para ello se define un intervalo en x con su valor inicial de x (vix), el paso de discretización del intervalo x (pasox) , y el valor final de x (vfx). Análogamente se construye un intervalo para "y". El comando meshgrid genera los pares ordenados posibles entre los elementos del intervalo x con los elementos del intervalo y.

  ```
  >>[X Y]=meshgrid (vix:pasox:vfx , viy:pasoy:vfy)
  ```

- ✓ La función se define con el comando **@** y se debe declarar como variables tanto a x como a y.

  ```
  >> f1=@(x,y)  x.^2+y.^2-4
  >>f2=@(x,y)  exp(x)-y
  >>Z1=f1(X,Y);
  >>Z2=f2(X,Y);
  ```

- ✓ El comando para graficar en tres dimensiones es **mesh**, dando la superficie correspondiente a $z_i = f_i(x, y)$
  Vea el help de meshgrid y otros comandos parecidos
  Nosotros necesitamos la curva en el plano de ecuación $f_i(x, y) = 0$, la cual es la curva de nivel correspondiente a $z_i = 0$. para graficar las curvas de nivel correspondientes a una función $z_i = f_i(x, y)$ se utiliza el comando **contour**

  ```
  [X Y]=meshgrid(-2.5:.1:2.5,-2.5:.1:2.5);
  >> f=@(x,y) x.^2+y.^2-4;
  >> z=f(X,Y);
  mesh(X,Y,z)
  % grafica la superficie en 3D es un paraboloide (Figura 1).
  >>figure,  contour (X,Y,z,[0 0])
  ```

% curva de nivel correspondiente a z=0. Se obtiene una curva plana de ecuación $x^2 + y^2 - 4 = 0$ es una circunferencia de radio 2 (Figura 2)

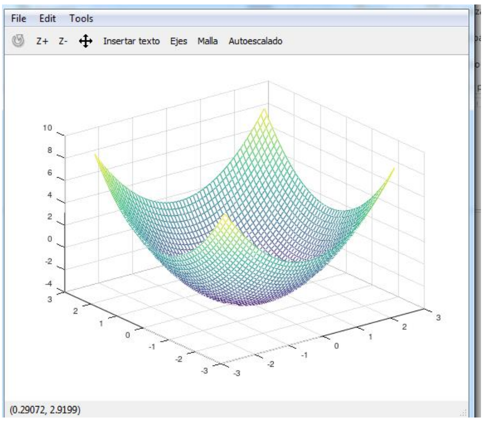

Figura 1

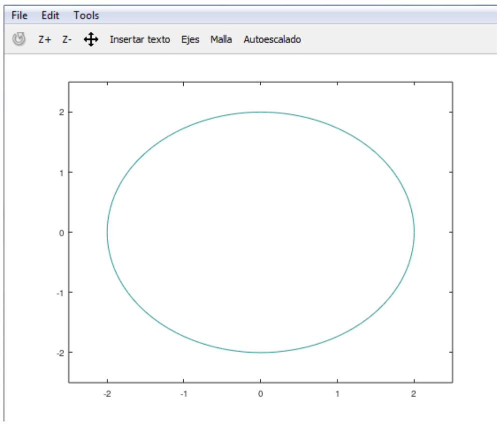

Figura2

- ✓ Para agregar en la gráfica otra función del sistema se procede en forma análoga, la figura en 3D no es necesaria, lo damos para información de Ustedes. Matlab tiene herramientas muy potentes para trabajar con gráficas

```
>>g=@(x,y) exp(x)-y
>>z2=g(X,Y)
>> hold on
>> contour(X,Y,z2)    % se puede observar en la figura 3
```

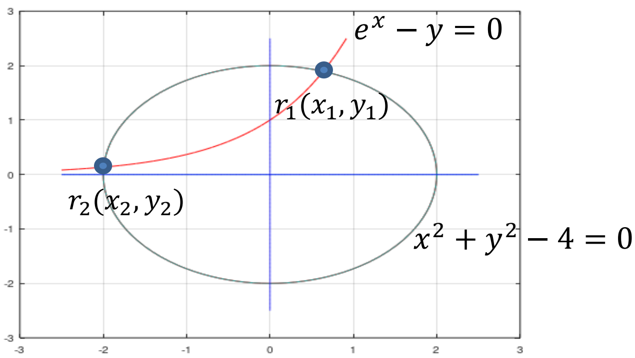

Figura3

Las raíces son $r_1 y\ r_2$, aplicando zoom podemos obtener en forma aproximada los valores de sus componentes.

Se puede observar que este método es muy limitado, solo se puede aplicar para un sistema de dos ecuaciones con dos incógnitas.

Estudiaremos el método más completo que se puede aplicar a un sistema de n- ecuaciones con n- incógnitas

MÉTODO DE NEWTON RAPHSON PARA SISTEMAS DE ECUACIONES NO LINEALES

Este método se puede aplicar a sistemas de n- ecuaciones con – incógnitas. Para explicar el método trabajaremos con un sistema de dos ecuaciones con dos incógnitas para luego generalizar

Sea el sistema de 2 ecuaciones no lineales con 2 incógnitas

$$\begin{cases} f(x,y) = 0 \\ g(x,y) = 0 \end{cases}$$

Sea (x_1, y_1) un valor inicial para comenzar a aplicar el método, y (x_2, y_2) una primera aproximación al valor de la raíz del sistema, aplicando el desarrollo de Taylor para funciones de dos variables en un entorno de (x_1, y_1) se tiene

$$f(x_2, y_2) = f(x_1, y_1) + df(x_1, y_1) + \frac{1}{2!}d^2 f(x_1, y_1) + \cdots + \frac{1}{n!}d^n f(x_1, y_1) + \cdots$$

Si consideramos una aproximación Lineal , esto es, realizar el desarrollo de Taylor hasta el df se tiene:

$$f(x_2, y_2) \cong f(x_1, y_1) + df(x_1, y_1) \quad (1)$$

Donde

$$df(x_1, y_1) = \frac{\partial f}{\partial x}(x_1, y_1)\left(\underbrace{x_2 - x_1}_{\Delta x}\right) + \frac{\partial f}{\partial y}(x_1, y_1)\left(\underbrace{y_2 - y_1}_{\Delta y}\right) \quad (2)$$

Llamando $\Delta x = x_2 - x_1 \quad \Delta y = y_2 - y_1$ y reemplazando (2) en (1)

$$f(x_2, y_2) \cong f(x_1, y_1) + \frac{\partial f}{\partial x}(x_1, y_1)\Delta x + \frac{\partial f}{\partial y}(x_1, y_1)\Delta y$$

Análogamente para la función $g(x, y)$

$$g(x_2, y_2) \cong g(x_1, y_1) + \frac{\partial g}{\partial x}(x_1, y_1)\Delta x + \frac{\partial g}{\partial y}(x_1, y_1)\Delta y$$

Si se considera que (x_2, y_2) es la primera aproximación a la raíz buscada, se impone que $f(x_2, y_2) = 0 \quad y \quad g(x_2, y_2) = 0$ luego

$$\begin{cases} 0 = f(x_1, y_1) + \frac{\partial f}{\partial x}(x_1, y_1)\Delta x + \frac{\partial f}{\partial y}(x_1, y_1)\Delta y \\ 0 = g(x_1, y_1) + \frac{\partial g}{\partial x}(x_1, y_1)\Delta x + \frac{\partial g}{\partial y}(x_1, y_1)\Delta y \end{cases}$$

En este sistema las incógnitas son Δx , Δy , que están elevadas al exponente uno, luego es un sistema lineal de dos ecuaciones con dos incógnitas que sabemos resolver.

$$\begin{cases} \frac{\partial f}{\partial x}(x_1, y_1)\Delta x + \frac{\partial f}{\partial y}(x_1, y_1)\Delta y = -f(x_1, y_1) \\ \frac{\partial g}{\partial x}(x_1, y_1)\Delta x + \frac{\partial g}{\partial y}(x_1, y_1)\Delta y = -g(x_1, y_1) \end{cases} \qquad (1)$$

Escrito en forma matricial y tomando la notación de las derivadas parciales

$$\frac{\partial f}{\partial x} = f_x \quad \frac{\partial f}{\partial y} = f_y$$

Se obtiene

$$\underbrace{\begin{bmatrix} f_x & f_y \\ g_x & g_y \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}}_{w} = \underbrace{\begin{bmatrix} -f(x_1, y_1) \\ -g(x_1, y_1) \end{bmatrix}}_{b} \quad (1)$$

$$A\ delta = b$$

Donde

$$delta = \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

Siendo A la matriz Jacobiana de f y g respecto de x, y

$$Jac = \begin{bmatrix} f_x & f_y \\ g_x & g_y \end{bmatrix}$$

cuyo determinante es el Jacobiano

$$J = \frac{\partial(f, g)}{\partial(x, y)}$$

Para que el sistema tenga solución única, la matriz Jacobiana asociada al sistema debe ser invertible, luego el determinante de la matriz asociada al sistema lineal debe ser distinto de cero, así sus filas son linealmente independientes, y el rango de la matriz es 2 .

Luego la **condición necesaria en el Método de Newton Raphson para sistemas de Ecuaciones No Lineales , es que el Jacobiano de las funciones sea distinto de cero**. Si esto no sucede el sistema no tiene solución y habrá que cambiar otro valor inicial donde el Jacobiano sea distinto de cero.

$$Condición\ Necesaria\ es\ que\ J = \left.\frac{\partial(f, g)}{\partial(x, y)}\right|_{(x_1, y_1)} \neq 0$$

Calculando la solución del sistema lineal (1) se obtiene Δx , Δy ,

$$delta = inv(A).b$$

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = inv\begin{pmatrix} f_x & f_y \\ g_x & g_y \end{pmatrix}.\begin{bmatrix} -f(x_1, y_1) \\ -g(x_1, y_1) \end{bmatrix}$$

A partir del vector de incrementos w se calcula el valor de (x_2, y_2) de la siguiente manera

$$\begin{cases} x_2 = x_1 + \Delta x \\ y_2 = y_1 + \Delta y \end{cases} \Leftrightarrow \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad \Leftrightarrow \quad X_2 = X_1 + delta$$

De esta manera se obtiene la primera aproximación de la raíz. Se debe analizar si los valores calculados son la solución del sistema no lineal, para ello se reemplaza el valor de (x_2, y_2) en el sistema , luego

$$si \quad \begin{matrix} |f(x_2, y_2)| < \varepsilon \\ |g(x_2, y_2)| < \varepsilon \end{matrix} \quad \Longrightarrow \quad (x_2, y_2) = \vec{r}$$

Si esto no se cumple entonces se considera el punto (x_2, y_2) como un nuevo valor inicial, o sea, $(x_1, y_1) = (x_2, y_2)$ y se repiten nuevamente los cálculos. Así se realiza un proceso iterativo hasta que se cumple que la diferencia en valor absoluto entre dos valores consecutivos es menor que un cierto error, que indicará la precisión con que se quiere trabajar

$$|x_2 - x_1| < \varepsilon \quad y \quad |y_2 - y_1| < \varepsilon \quad esto\ es\ equivalente\ a\ que\ \left\|\overrightarrow{delta}\right\| < \varepsilon$$

También se puede considerar la condición de corte en términos de las funciones del sistema o sea

$$\left\|\begin{matrix} f(x_2, y_2) \\ g(x_2, y_2) \end{matrix}\right\| = \left\|\vec{F}(x_2, y_2)\right\| < \varepsilon$$

Mientras más pequeño sea ε mejor será el valor de la aproximación a la raíz

Resumiendo

ALGORITMO

1. Datos $f(x,y), g(x,y)$ y $\overrightarrow{X_1} = (x_1, y_1)$ $como\ Valor\ Inicial$ y ε
2. Evaluar $f(x_1, y_1)\,, g(x_1, y_1)$ $f_x(x_1, y_1)\,, f_y(x_1, y_1), g_x(x_1, y_1)\,, g_y(x_1, y_1)$
3. Calcular

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = inv\begin{pmatrix} f_x & f_y \\ g_x & g_y \end{pmatrix}. \begin{bmatrix} -f(x_1, y_1) \\ -g(x_1, y_1) \end{bmatrix}$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

4. Analizar si

$$\|\vec{d}\| < \varepsilon \quad \begin{cases} \xrightarrow{\text{sí}} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \vec{r} \\ \xrightarrow{\text{no}} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} \text{ se repite desde 2.} \end{cases}$$

Ejemplo 2

Resolver el sistema dado en el Ejemplo 1 aplicando el método de Newton Raphson

$$\begin{matrix} x^2 + y^2 - 4 = 0 & \longrightarrow & f(x,y) = x^2 + y^2 - 4 \\ e^x - y = 0 & \longrightarrow & g(x,y) = e^x - y \end{matrix}$$

Consideraremos como valor inicial a $\overrightarrow{X_1} = (x_1, y_1) = (1,1)$

Cálculos de las funciones y las derivadas

$$\left.\begin{matrix} f \\ g \end{matrix}\right|_{\overrightarrow{X_1}} = \begin{matrix} -2 \\ 1.7183 \end{matrix} \qquad \left.\begin{matrix} f_x = 2x \\ g_x = e^x \end{matrix}\right|_{\overrightarrow{X_1}} = \begin{matrix} 2 \\ e \end{matrix} \qquad \left.\begin{matrix} f_y = 2y \\ g_y = -1 \end{matrix}\right|_{\overrightarrow{X_1}} = \begin{matrix} 2 \\ -1 \end{matrix}$$

Luego el sistema (1) queda:

$$\begin{cases} 2\Delta x + 2\Delta y = 2 \\ e\Delta x - \Delta y = -1.7183 \end{cases}$$

Resolviendo el sistema será

$$Jac.delta = b \qquad \begin{vmatrix} 2 & 2 \\ e & -1 \end{vmatrix} \begin{vmatrix} \Delta x \\ \Delta y \end{vmatrix} = \begin{vmatrix} 2 \\ -1.7183 \end{vmatrix}$$

Resuelto en OCTAVE

$$\begin{vmatrix} \Delta x \\ \Delta y \end{vmatrix} = inv\left(\begin{vmatrix} 2 & 2 \\ e & -1 \end{vmatrix}\right) \cdot \begin{vmatrix} 2 \\ -1.7183 \end{vmatrix}$$

$$\begin{vmatrix} \Delta x \\ \Delta y \end{vmatrix} = \begin{bmatrix} -0.19318 \\ 1.19318 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} -0.19318 \\ 1.19318 \end{bmatrix}$$

$$X_2 = \begin{bmatrix} 0.80682 \\ 2.19318 \end{bmatrix}$$

Ahora calculamos el valor de la función en el nuevo punto X_2 será

>>F=@(x,y) [x.^2+y.^2-4;exp(x)-y]

>> Fx2=F($X_2(1), X_2(2)$)

Fx2

1.460997
0.047591

Es evidente que la función en el punto X_2 está lejos del cero luego se toma a X_2 como X_1 y se repiten los cálculos

La matriz Jacobiana se construye calculando las derivadas en cada elemento Lo mejor es calcular las derivadas parciales por definición del cociente incremental, así se puede hacer un programa que sirva para cualquier función.

Programa

Para resolver sistemas de ecuaciones no lineales con Matlab u Octave

```
function [r,N]=NRsistema(F,X1,e)
X2=X1+0.5;
N=0;
while  norm(X2-X1)>e
 X1=X2;
 N=N+1;
 b=-F(X1(1),X1(2));
 Inc=0.00000001;
 %parcial respecto  a x de ambas funciones
Jac1=(F(X1(1)+Inc,X1(2))-F(X1(1),X1(2)))/Inc;
 %parcial respecto  a y de ambas funciones
Jac2=(F(X1(1),X1(2)+Inc)-F(X1(1),X1(2)))/Inc;
Jac=[Jac1 Jac2];
delta=Jac\b;
X2=X1+delta
end
r=X2;
```

El programa se ejecuta de la siguiente manera

```
>>X1=[1,1]
>>F=@(x,y) [x.^2+y.^2-4;exp(x)-y]
>>e=0.001
[r n]=NRsistema(F,X1,0.0001)
r =
   0.63926
   1.89508
N=5
```

.