

Práctica 6

Sistemas de ecuaciones no lineales

En esta práctica revisaremos algunos métodos básicos para la resolución numérica de sistemas de ecuaciones no lineales.

6.1. Método iterativo del punto fijo

Partimos de un sistema de ecuaciones no lineales de la forma

$$\begin{aligned}f_1(x_1, \dots, x_n) &= 0, \\f_2(x_1, \dots, x_n) &= 0, \\&\vdots \\f_n(x_1, \dots, x_n) &= 0,\end{aligned}$$

del que se pretende obtener una solución. Para utilizar el método del punto fijo, se reescribe el sistema de la forma

$$\begin{aligned}x_1 &= g_1(x_1, \dots, x_n), \\x_2 &= g_2(x_1, \dots, x_n), \\&\vdots \\x_n &= g_n(x_1, \dots, x_n).\end{aligned}$$

Se parte de una solución inicial (x_1^0, \dots, x_n^0) y se construye la sucesión

$$(x_1^{k+1}, \dots, x_n^{k+1}) = (g_1(x_1^k, \dots, x_n^k), \dots, g_n(x_1^k, \dots, x_n^k)) .$$

Si esta sucesión converge, lo hará a una solución del sistema de ecuaciones próxima a la solución inicial. La convergencia de la sucesión depende mucho de hacer una buena elección de las funciones g_1, \dots, g_n .

6.2. Método de Newton-Raphson

Este método se basa en utilizar el desarrollo de Taylor para aproximar una función derivable en las proximidades de un punto. Partimos de un sistema de la forma

$$\begin{aligned} f_1(x_1, \dots, x_n) &= 0, \\ f_2(x_1, \dots, x_n) &= 0, \\ &\vdots \\ f_n(x_1, \dots, x_n) &= 0, \end{aligned}$$

del que se pretende obtener una solución. Se supone que la solución buscada, (x_1, \dots, x_n) , se escribe de la forma

$$\begin{aligned} x_1 &= x_1^0 + \Delta x_1, \\ &\vdots \\ x_n &= x_n^0 + \Delta x_n, \end{aligned}$$

y, por tanto, se tiene que

$$\begin{aligned} f_1(x_1^0 + \Delta x_1, \dots, x_n^0 + \Delta x_n) &= 0, \\ &\vdots \\ f_n(x_1^0 + \Delta x_1, \dots, x_n^0 + \Delta x_n) &= 0. \end{aligned}$$

Utilizando el desarrollo de Taylor alrededor de (x_1^0, \dots, x_n^0) y quedándonos con el primer orden, tenemos

$$\begin{aligned} f_1(\vec{x}^0) + \sum_{i=1}^n \frac{\partial f_1(\vec{x}^0)}{\partial x_i} \Delta x_i &\approx 0, \\ &\vdots \\ f_n(\vec{x}^0) + \sum_{i=1}^n \frac{\partial f_n(\vec{x}^0)}{\partial x_i} \Delta x_i &\approx 0. \end{aligned}$$

donde $\vec{x}^0 = (x_1^0, \dots, x_n^0)$. Introduciendo la notación $\vec{F} = (f_1, \dots, f_n)$ y

$$DF = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix},$$

queda

$$\vec{F}(\vec{x}^0) + DF(\vec{x}^0)(\vec{x} - \vec{x}^0) \approx 0,$$

entonces

$$\vec{x} = \vec{x}^0 - DF^{-1}(\vec{x}^0) \vec{F}(\vec{x}^0)$$

y el método de Newton consiste en calcular la sucesión

$$\vec{x}^{k+1} = \vec{x}^k - DF^{-1}(\vec{x}^k) \vec{F}(\vec{x}^k).$$

Dado que calcular explícitamente la matriz inversa de la matriz Jacobiana no es, en general, un proceso muy eficiente desde el punto de vista numérico, a la hora de implementar el método se hace en dos pasos:

1. Se resuelve el sistema

$$DF(\vec{x}^k) \Delta \vec{x}^{k+1} = -\vec{F}(\vec{x}^k).$$

2. Se calcula

$$\vec{x}^{k+1} = \vec{x}^k + \Delta \vec{x}^{k+1}.$$

Las siguientes funciones muestran una posible implementación en Matlab del método de Newton, para resolver el sistema

$$\begin{aligned} x_1^2 - 10x_1 + x_2^2 + 8 &= 0, \\ x_1x_2^2 + x_1 - 10x_2 + 8 &= 0 \end{aligned}$$

```
function y=f(x)
% funcion para utilizar con newtonsi.m
y(1)=x(1)^2-10*x(1)+x(2)^2+8;
y(2)=x(1)*x(2)^2+x(1)-10*x(2)+8;

function df=jac(x)
% matriz jacobiana para usar con newtonsi.m
df(1,1)=2*x(1)-10;
df(1,2)=2*x(2);
df(2,1)=x(2)^2+1;
df(2,2)=2*x(1)*x(2)-10;
```

```

function [xr,k]=newtonsi(x,tol,imax)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Metodo de Newton para sistemas de ecuaciones
% Uso: [xr,k]=newtonsi(x,tol,imax)
%
% Input:
%x = vector x1,x2,...,xn inicial,
%tol=tolerancia
%
% Se ha de disponer de las funciones:
%   f.m funcion y=f(x) donde se define el sistema
%   jac.m funcion df=jac(x) donde se define la matriz
%   derivada del sistema.
%
% Output: xr= raiz, k= numero de iteraciones.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
k=1;
epi=1;
x1=x;
while norm(epi)>tol
    x=x1;
    fxn=f(x);
    axn=jac(x);
    epi=axn\fxn';
    x1=x-epi';
    k=k+1;
    if k>imax
        disp('no converge')
        break
    end
end
xr=x1;

```

6.3. Método de Broyden

El método de Broyden generaliza el método de la secante de resolución de ecuaciones no lineales para el caso de un sistema de ecuaciones.

Para el caso unidimensional se toma una aproximación de la derivada de

$f(x)$,

$$s^k = \frac{f(x^k) - f(x^{k-1})}{x^k - x^{k-1}},$$

y se hacen iteraciones de la forma

$$x^{k+1} = x^k - (s^k)^{-1} f(x_n).$$

Para el caso de un sistema de ecuaciones, se ha de obtener una aproximación de la matriz derivada, S_n , que cumpla

$$S^k (\vec{x}_n^k - \vec{x}^{k-1}) = \vec{F}(\vec{x}^k) - \vec{F}(\vec{x}^{k-1}).$$

Introducimos la siguiente notación

$$\begin{aligned}\vec{b}^k &= \vec{x}^{k+1} - \vec{x}^k, \\ \Delta \vec{F}^k &= \vec{F}(\vec{x}^{k+1}) - \vec{F}(\vec{x}^k), \\ S^{k+1} &= S^k + C^k.\end{aligned}$$

Con lo que tenemos que

$$S^{k+1} (\vec{x}^{k+1} - \vec{x}^k) = \vec{F}(\vec{x}^{k+1}) - \vec{F}(\vec{x}^k),$$

o sea,

$$(S^k + C^k) \vec{b}^k = \Delta \vec{F}^k,$$

y, por tanto,

$$C^k \vec{b}^k = \Delta \vec{F}^k - S^k \vec{b}^k.$$

De este modo, dado un vector \vec{w}^{kT} tal que $\vec{w}^{kT} \vec{b}^k \neq 0$, podemos elegir

$$C^k = \frac{1}{\vec{w}^{kT} \vec{b}^k} (\Delta \vec{F}^k - S^k \vec{b}^k) \vec{w}^{kT},$$

ya que

$$C^k \vec{b}^k = \Delta \vec{F}^k - S^k \vec{b}^k.$$

Si elegimos $\vec{w}^{kT} = \vec{b}^k$, se obtiene el primer método de Broyden, y si elegimos $\vec{w}^k = S^{kT} \vec{b}^k$ se obtiene el segundo método de Broyden.

Así pues, el método de Broyden para sistemas parte de una aproximación inicial para la raíz, \vec{x}_0 , y una aproximación para la matriz derivada del sistema, S^0 , y se basa en iteraciones de la forma

$$\begin{aligned}\vec{b}^k &= -S^{k-1}\vec{F}(\vec{x}^k) , \\ \vec{x}^{k+1} &= \vec{x}^k + \vec{b}^k , \\ \Delta\vec{F}^k &= \vec{F}(\vec{x}^{k+1}) - \vec{F}(\vec{x}^k) , \\ S^{k+1} &= S^k + \frac{1}{\vec{w}^{kT}\vec{b}^k} \left(\Delta\vec{F}^k - S^k\vec{b}^k \right) \vec{w}^{kT} .\end{aligned}$$

6.4. Ejercicios

E.1.- Un experimento biológico relaciona la concentración de una cierta toxina C con el tiempo mediante el modelo

$$C(t) = \text{sen}(t)e^{-\alpha t} + \cos(2t)e^{-\beta t} , \quad (6.1)$$

donde los parámetros α y β son desconocidos.

Del experimento se han recogido los siguientes datos

t	0	0.6	0.9	1.3	1.6
$C(t)$	1.0	0.3689	0.0371	-0.1620	-0.1608
t	2	2.4	2.7	3	
$C(t)$	-0.0718	0.0135	0.0446	0.0482	

- Plantea las ecuaciones normales para los parámetros α y β asociadas a minimizar el error cuadrático medio obtenido al suponer que los datos experimentales siguen el modelo (6.1).
- Resuelve el sistema de ecuaciones obtenido usando el método de Newton tomando como valor inicial $(\alpha^0, \beta^0) = (0,5, 0,5)$ y una tolerancia de 10^{-6} .
- Haz una representación gráfica del modelo obtenido junto con los datos experimentales.
- Compara la convergencia del método de Newton si la matriz Jacobiana se actualiza cada iteración con el método obtenido al actualizar la matriz jacobiana cada 4 iteraciones.

E.2.- Contruye una función de Matlab que implemente el método del punto fijo, otra que implemente el primer método de Broyden y otra que implemente el segundo método de Broyden.

Compara el funcionamiento del método del punto fijo, del método de Newton y de los métodos de Broyden al resolver el sistema de ecuaciones

$$\begin{aligned}3x_1 - \cos(x_2x_3) - \frac{1}{2} &= 0 \\x_1^2 - 81(x_2 + 0,1)^2 + \text{sen}(x_3) + 1,06 &= 0 \\e^{-x_1x_2} + 20x_3 + \frac{1}{3}(10\pi - 3) &= 0\end{aligned}$$

Para ello, haz una gráfica en donde se compare la evolución del error con el número de iteraciones.

Práctica 7

Más sobre sistemas de ecuaciones no lineales

Los métodos básicos expuestos en la práctica anterior dan solución a problemas sencillos. En general, la búsqueda de raíces de un sistema de ecuaciones no lineales es un problema numéricamente difícil y va a ser necesario desarrollar diferentes técnicas dependiendo del problema a tratar. A continuación, veremos algunas posibilidades útiles cuando no se conoce una buena aproximación inicial del problema a resolver, o cuando los métodos básicos son muy costosos desde el punto de vista computacional.

7.1. Métodos de continuación

Supongamos que se parte de un sistema de ecuaciones no lineales de la forma

$$\vec{F}(\vec{x}) = \vec{0}, \quad (7.1)$$

cuya solución, \vec{x}^* , es desconocida, y se parte de una aproximación inicial \vec{x}^0 .

Los métodos para la resolución de sistemas de ecuaciones no lineales tienen, en general, un radio de convergencia pequeño y es necesario partir de una solución inicial adecuada para que converjan. Conseguir esta solución inicial es, en ocasiones, complicado y hay que tratar de ampliar el radio de convergencia del método numérico de algún modo. Una posibilidad es utilizar un método de continuación.

Los métodos de continuación consisten en plantear una familia de problemas no lineales dependientes de un parámetro, $\lambda \in [0, 1]$, de forma que para

$\lambda = 0$, se parta de un problema con solución conocida y para $\lambda = 1$ se llega al problema (7.1).

Si partimos de una aproximación, \vec{x}^0 , al problema (7.1), podemos construir la función

$$\vec{G}(\lambda, \vec{x}) = \lambda \vec{F}(\vec{x}) + (1 - \lambda) \left(\vec{F}(\vec{x}) - \vec{F}(\vec{x}^0) \right) . \quad (7.2)$$

A la función \vec{G} , se le llama una homotopía entre la función $\vec{G}(0, \vec{x}) = \vec{F}(\vec{x}) - \vec{F}(\vec{x}^0)$ y la función $\vec{G}(1, \vec{x}) = \vec{F}(\vec{x})$.

Podemos fijar un entero N y tomar una sucesión de parámetros de la forma

$$\lambda_j = \frac{j}{N} , \quad j = 0, 1, \dots, N ,$$

con lo que para estos valores de λ_j obtenemos una familia de problemas no lineales que se van resolviendo de forma que, para resolver el problema

$$\vec{G}(\lambda_j, \vec{x}) = \lambda_j \vec{F}(\vec{x}) + (1 - \lambda_j) \left(\vec{F}(\vec{x}) - \vec{F}(\vec{x}^0) \right) ,$$

utilizamos como condición inicial, la solución obtenida en el problema

$$\vec{G}(\lambda_{j-1}, \vec{x}) = \lambda_{j-1} \vec{F}(\vec{x}) + (1 - \lambda_{j-1}) \left(\vec{F}(\vec{x}) - \vec{F}(\vec{x}^0) \right) .$$

Por otra parte, supongamos que $\vec{x}(\lambda)$ es una única solución de

$$\vec{G}(\lambda, \vec{x}) = \vec{0} , \quad \lambda \in [0, 1] .$$

El conjunto $\{\vec{x}(\lambda) / 0 \leq \lambda \leq 1\}$ puede verse como una curva parametrizada por λ que va desde $\vec{x}(0) = \vec{x}^0$ hasta $\vec{x}(1) = \vec{x}^*$. Si las funciones $\vec{G}(\lambda, \vec{x})$ y $\vec{x}(\lambda)$ son diferenciables, se tiene que

$$\frac{\partial \vec{G}(\lambda, \vec{x}(\lambda))}{\partial \lambda} + \sum_{i=1}^n \frac{\partial \vec{G}(\lambda, \vec{x}(\lambda))}{\partial x_i} x'_i(\lambda) = \vec{0} . \quad (7.3)$$

Como

$$\vec{G}(\lambda, \vec{x}(\lambda)) = \vec{F}(\vec{x}(\lambda)) + (\lambda - 1) \vec{F}(\vec{x}(0)) ,$$

tenemos que,

$$\frac{\partial \vec{G}}{\partial x_i} = \frac{\partial \vec{F}}{\partial x_i} = DF(\vec{x}(\lambda)) , \quad \frac{\partial \vec{G}}{\partial \lambda} = \vec{F}(\vec{x}(0)) ,$$

con lo que se tiene el sistema

$$DF(\vec{x}(\lambda)) \vec{x}'(\lambda) = -\vec{F}(\vec{x}(\lambda)) , \quad (7.4)$$

que es una ecuación diferencial ordinaria para $\vec{x}(\lambda)$, con la condición inicial $\vec{x}(0) = \vec{x}^0$. Distintos métodos de continuación se obtienen aplicando distintos métodos numéricos a la resolución del problema (7.4).

Por ejemplo, podemos despejar

$$\vec{x}' = (DF(\vec{x}(\lambda)))^{-1} \vec{F}(\vec{x}(\lambda)) \equiv \vec{H}(\vec{x}(\lambda)) , \quad (7.5)$$

y si utilizamos el método de Runge-Kutta de cuatro pasos tendremos que calcular para cada iteración, j ,

$$\begin{aligned} \vec{K}_1 &= \vec{H}(\vec{x}(\lambda_j)) , \\ \vec{K}_2 &= \vec{H}\left(\vec{x}(\lambda_j) + \frac{\Delta\lambda}{2}\vec{K}_1\right) , \\ \vec{K}_3 &= \vec{H}\left(\vec{x}(\lambda_j) + \frac{\Delta\lambda}{2}\vec{K}_2\right) , \\ \vec{K}_4 &= \vec{H}\left(\vec{x}(\lambda_j) + \Delta\lambda\vec{K}_2\right) , \end{aligned}$$

donde $\Delta\lambda = 1/N$. De este modo, se tiene el esquema

$$\vec{x}(\lambda_{j+1}) = \vec{x}(\lambda_j) + \frac{\Delta\lambda}{6} (\vec{K}_1 + 2\vec{K}_2 + 2\vec{K}_3 + \vec{K}_4) .$$

Otra posibilidad consiste en utilizar directamente la función `ode23()` del Matlab para resolver la ecuación (7.5).

7.2. Métodos con dirección de búsqueda

Otra posibilidad que puede resultar eficiente es modificar, por ejemplo, el método de Newton para ampliar el radio de convergencia del método a costa de reducir la velocidad de convergencia.

Supongamos que se quiere encontrar la raíz $x^* = 0$ de la función $f(x) = \arctan(x)$ y se parte de $x^0 = 10$. Es fácil ver que el método de Newton diverge. La razón de esto es $f' = (1+x^2)^{-1}$ es pequeña cuando x es grande, mientras que $f(x)$ se mantiene en valor cercano a $\pm\pi/2$. Esto hace que las iteraciones de Newton diverjan. Para solucionar este problema, se interpreta

$-f(x)/f'(x)$ com una direcció de búsqueda y se introduce un factor que va reduciendo la magnitud de las iteraciones hasta conseguir que el método converja, com se muestra en el siguiente algoritmo:

Algoritmo

1. seleccionar: `tol`, x (inicial).
2. mientras $|f(x)| > \text{tol}$, hacer:
 - a) si $f'(x) = 0$ parar com un error.
 - b) $s = -f(x)/f'(x)$ (direcció de búsqueda)
 - c) $x_p = x + s$
 - d) Si $|f(x_p)| < |f(x)|$ entonces $x = x_p$, (se acepta el punto).
Si no
 $s = s/2$ e ir al paso 2(c).

Esta estrategia se puede generalizar de distintos modos. En general, se calcula una direcció de búsqueda, com la de Newton

$$d = -\frac{f(x_c)}{f'(x_c)},$$

y las distintas iteraciones se hacen mediante pasos de la forma $s = \lambda d$ com $\lambda = 2^{-j}$ y $j \geq 0$, hasta que $f(x_c + s)$ satiface

$$|f(x_c + \lambda d)| < (1 - \alpha\lambda) |f(x_c)|,$$

donde α es un número pequeño que, generalmente, se toma com $\alpha = 10^{-4}$. Este tipo de estrategia de conoce com la *Regla de Armijo*.

7.3. Métodos de Newton inexactos

Supongamos que se pretende usar el método de Newton para resolver un problema no lineal de gran dimensió de la forma

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

Ya hemos visto que cundo se tratan problemas de gran dimensió pueden aparecer problemas de almacenamiento de las matrices y que el coste computacional crece mucho cuando se utilizan métodos directos para la resolució de sistemas de ecuaciones.

El método de Newton se basa en la solución de problemas lineales de la forma

$$DF(\vec{x}_i) \Delta \vec{x} = -\vec{F}(\vec{x}_i) . \quad (7.6)$$

Supongamos que para resolver el sistema (7.6) se utiliza un método iterativo, entonces el método resultante se conoce como un método de *Newton inexacto*. Entre los métodos iterativos posibles es interesante considerar los métodos iterativos basados en los subespacios de Krylov. Estos métodos se basan en productos matriz-vector. Para el problema (7.6), dado un vector \vec{v} , interesa ver cómo se podría calcular el producto $DF(\vec{x}_i) \vec{v}$. Si tenemos en cuenta que

$$\vec{F}(\vec{x}_i + \varepsilon \vec{v}) \approx \vec{F}(\vec{x}_i) + \varepsilon DF(\vec{x}_i) \vec{v} + o(\varepsilon^2) ,$$

tenemos que

$$DF(\vec{x}_i) \vec{v} \approx \frac{\vec{F}(\vec{x}_i + \varepsilon \vec{v}) - \vec{F}(\vec{x}_i)}{\varepsilon} . \quad (7.7)$$

Con esta aproximación podemos calcular el producto matriz-vector mediante una evaluación de la función no lineal \vec{F} y sin necesidad de almacenar su matriz jacobiana. Utilizando esta estrategia obtenemos un método de Newton-Krylov sin matriz jacobina.

Para que la aproximación (7.7) funcione bien, hace falta tomar un valor adecuado de ε . Algunos autores proponen elegir ε como un número algo mayor que la raíz cuadrada del épsilon de la máquina.

7.4. Ejercicios

E.1.- Dado el sistema de ecuaciones

$$\begin{aligned} x^2 + 3y^2 - z^3 + w^2 - 5 &= 0 \\ x^3 - 2y^2 - 10z + w &= 0 \\ x^2 + y^3 + z^2 - w + 20 &= 0 \\ x - y^3 + z + w^3 - 10 &= 0 \end{aligned}$$

obten una solución aproximada del mismo utilizando primero como punto inicial $(x, y, z, w)^0 = (4, -4, 4, -4)$, y posteriormente $(x, y, z, w)^0 = (1, -1, 1, -1)$. Utiliza para el primer caso un método de Newton. Para el segundo caso comprueba que el método de Newton no converge y utiliza un método de continuación.

E.2.- Se sabe que la solución del problema de contorno

$$y'' = \frac{1}{8} (32 + 2x^3 - yy') ; \quad x \in [1, 3] , \quad y(1) = 17 , \quad y(3) = \frac{43}{3} , \quad (7.8)$$

es

$$y = x^2 + \frac{16}{x} .$$

Discretiza el problema (7.8) utilizando una malla con 20 nodos, y las aproximaciones

$$y'(x_i) \approx \frac{y_{i+1} - y_{i-1}}{2\Delta x} , \quad y''(x_i) \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{\Delta x^2} ,$$

y obtén la solución numérica del problema utilizando el método de Newton directamente y una variación del mismo que haga uso de la *Regla de Armijo*. Compara la solución numérica obtenida con la solución exacta.

E.3.- Considera el problema de convección difusión no lineal siguiente

$$-\frac{d^2u}{dx^2} + 10u \frac{du}{dx} = \frac{1}{\sqrt{50-x}} + \left(500 + \frac{1}{4(50-x)^{\frac{3}{2}}} \right) x - 15x^2 ,$$

con $x \in [0, 50]$ y $u(0) = u(50) = 0$.

Discretiza el problema usando un mallado de 200 nodos y obtén su solución numérica usando un método de Newton-GMRES sin utilizar la matriz Jacobiana. Compara el resultado con la solución exacta

$$u = x\sqrt{50-x} .$$