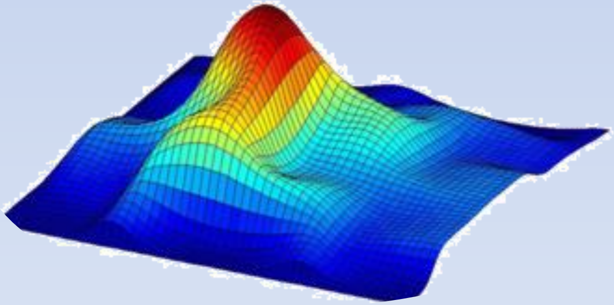
Método de SOR



PROF. JENNY PÉREZ

Métodos Iterativos

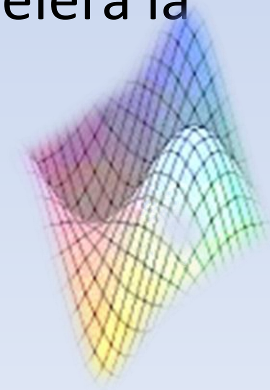
Los métodos iterativos transforman el sistema de ecuaciones lineales $AX=b$, por un sistema de la forma matricial $(D-L-U)x=b$, donde D es la matriz diagonal de A, L es la matriz estrictamente triangular inferior y U es la matriz estrictamente triangular superior. Posteriormente operan algebraicamente hasta obtener una representación equivalente que permite generar una sucesión de vectores de soluciones aproximadas para el vector x.

Método de Gauss-Seidel

Una modificación del método de Jacobi que permite acelerar la convergencia del método, que consiste en utilizar cada x_i^k para $j < i$, al reemplazarlos en los x_j^{k-1} debido a que los valores x_i^k previamente calculados permiten dar una mejor aproximación.

Método de Sobrerelajación sucesiva ó S.O.R

- Una modificación en el método de Gauss Seidel, consiste en agregar un parámetro w que permite mejorar la velocidad de convergencia, siempre que $1 < w < 2$.
- Si 0<w<1 se conoce como sub-relajación, se emplea para un sistema que no converge.
- Si w=1 (Método Gauss-Seidel)
- Si 1<w<2 se le llama sobre-relajación, el cual acelera la convergencia del sistema que ya converge.
- Si w>2 el método diverge.



Métodos Numéricos

SISTEMAS DE ECUACIONES LINEALES

En este capítulo se examinan métodos iterativos y métodos directos para resolver sistemas de ecuaciones lineales:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = n \end{cases}$$

Este sistema puede ser expresado en la forma matricial como $Ax = b$.

Donde $A = (aij)_{nxn} \in M(n \times n, \mathbb{R})$, $b = (b1, b2, \dots, bn)^T \in \mathbb{R}^n$ y

$x = (x1, x2, \dots, xn)^T \in \mathbb{R}^n$es la incógnita.

FORMULACIÓN

- Definiendo una variable G tal que:

$$G = \frac{1}{aii}(bi - \sum_{j=1}^{i-1} aijx_j^{(k)} - \sum_{j=i+1}^{n} aijx_j^{(k-1)})$$

- Si se aplica un parámetro w a la expresión iterativa de Gauss- Seidel de tal forma que se obtenga:

$$x^{(k)} = (1-w)x^{(k-1)} + wG$$

- Se define ahora la ecuación iterativa del método de S.O.R:

$$xi^{(k)} = (1-w)xi^{(k-1)} + \frac{w}{aii}(bi - \sum_{j=1}^{i-1} aijx_j^{(k)} - \sum_{j=i+1}^{n} aijx_j^{(k-1)})$$

Definiciónes: Definición 1.1.

Polinomio característico:

$$p(\lambda) = \det(A - \lambda I)$$

Definición 1.2 Los valores característicos o valores propios son los ceros del polinomio característico de la matriz A, y se definen como λ, con la propiedad $(A - \lambda I)\, x = 0,\ si\ x \neq 0$.

X se llama vector característicos o propio de A correspondiente a un valor característico λ.

Definición 1.3 El radio espectral $\rho(A) = \underset{i=1,\ldots,n}{máx} |\lambda i|$

Definición 1.4 Si A es una matriz definida positiva y tridiagonal entonces la elección optima de w para el método SOR está dada por

$$w = \frac{2}{1 + \sqrt{1 - [\rho(Tj)]^2}}$$

Donde $Tj = D^{-1}(L + U)$ entonces el método S.O.R converge. Luego de calcular el parámetro w de una matriz, se deben tener en cuenta que si la matriz es definida positiva y su parámetro está comprendido en el intervalo 0<w<2, entonces el método S.O.R converge para cualquier elección del vector inicial aproximado $x^{(0)}$.

ALGORITMO

Paso 1: Se inicializa el proceso en $k = 1$

Paso 2: Para cada $i = 1, \dots, n$;

$$xi^{(k)} = (1 - w)xi^{(k-1)} + \frac{w}{aii}(bi - \sum_{j=1}^{i-1} aijx_j^{(k)} - \sum_{j=i+1}^{n} aijx_j^{(k-1)}$$

Paso 3: Se calcula el error de tolerancia utilizando la norma infinita ι_∞.

$$\left\|x^{(k)} - x^{(k-1)}\right\|_\infty = \underset{i=1,\dots,n}{máx}\{\left|x^{(k)} - x^{(k-1)}\right|, \dots, \left|x^{(k)} - x^{(k-1)}\right|\}$$

$$Si\ \left\|x^{(k)} - x^{(K-1)}\right\|_\infty < \ \varepsilon\ vaya\ al\ paso\ 5$$

$$Si\ \left\|x^{(k)} - x^{(K-1)}\right\|_\infty > \ \varepsilon\ vaya\ al\ paso\ 4$$

Paso 4: Haga $\mathrm{k} = \mathrm{k} + 1$ y vaya al paso 2

Paso 5: Finaliza el proceso

EJERCICIO

Resolver el siguiente el siguiente sistema de ecuaciones utilizando w=0.9, un error de tolerancia del 0% y (0,0,0,0) como valores iniciales.

$$8x_1 + 5x_2 + 3x_3 + 2x_4 = 35$$

$$x_1 + 6x_2 - 2x_3 + 4x_4 = 23$$

$$-x_1 + 2x_2 + 5x_3 + 3x_4 = 30$$

$$3x_1 + x_2 + 2x_3 - 5x_4 = -9$$

- Solución:

$$8x_1 + 5x_2 + 3x_3 + 2x_4 = 35$$

$$x_1 + 6x_2 - 2x_3 + 4x_4 = 23$$

$$-x_1 + 2x_2 + 5x_3 + 3x_4 = 30$$

$$3x_1 + x_2 + 2x_3 - 5x_4 = -9$$

$$x_1 = \frac{w35 - w5x_2 - w3x_3 - w2x_4 + (1-w)\ x_1}{8}$$

$$x_2 = \frac{w23 - wx_1 + w2x_3 - w4x_4 + (1-w)\ x_2}{6}$$

$$x_3 = \frac{w30 + wx_1 - w2x_2 - w3x_4 + (1-w)\ x_3}{5}$$

$$x_4 = \frac{-w9 - w3x_1 - wx_2 - w2x_3 + (1-w)\ x_4}{-5}$$

$$x_1 = \frac{31.5 - 4.5(0) - 2.7(0) - 1.8(0) + 0.8(0)}{8} = \mathbf{3.9375}$$

$$x_2 = \frac{20.7 + 0.9(3.9375) - 1.8(0) - 3.6(0) + 0.6(0)}{6} = \mathbf{2.859375}$$

$$x_3 = \frac{27 + 0.9(3.9375) - 1.8(2.859375) - 2.7(0) + 0.5(0)}{5} = \mathbf{5.079375}$$

$$x_4 = \frac{-8.1 - 2.7(3.9375) - 0.9(2.859375) - 1.8(5.079375) - 0.5(0)}{-5} = \mathbf{6.0895125}$$

ALGORITMO EN MATLAB

```
function Sor4(A,b,x,tol,n)
%A = matriz
%b = vector  derecho de la matriz
%x = aproximacion lineal
%tol = tolerancia
%n = numero de iteraciones maximas
dim = size(A);
y = x;D = diag(diag(A));L = tril(A)-D;
U = triu(A)-D;Tj = (D)\(-L-U);ro  = max(eig(Tj));
w = 2/(1+sqrt(1-ro^2));band = 0;
k = 1;
while k<=n && band==0
    for i = 1 : dim(1)
        sum1 = 0;
        for j = 1: i-1
            sum1 = sum1+(A(i,j)*y(j));
        end
        sum2 = 0;
        for j = i+1 : dim(2)
            sum2 = sum2+(A(i,j)*x(j));
        end
        y(i) = (1-w)*x(i)+(w/A(i,i)*(b(i)-sum1-sum2))
    end
    if norm(y-x,inf)<tol
        band = 1;
    else
      x=y;
    end
    k = k+1;
end
if band == 1
        fprintf('vector solucion: \n');
        disp(y);
        fprintf('total iteraciones realizadas: %d \n',k);
        fprintf('w optimo: %d \n',w);
else
    fprintf('no se pudo hallar solucion en %d iteraciones...\n',n);
```

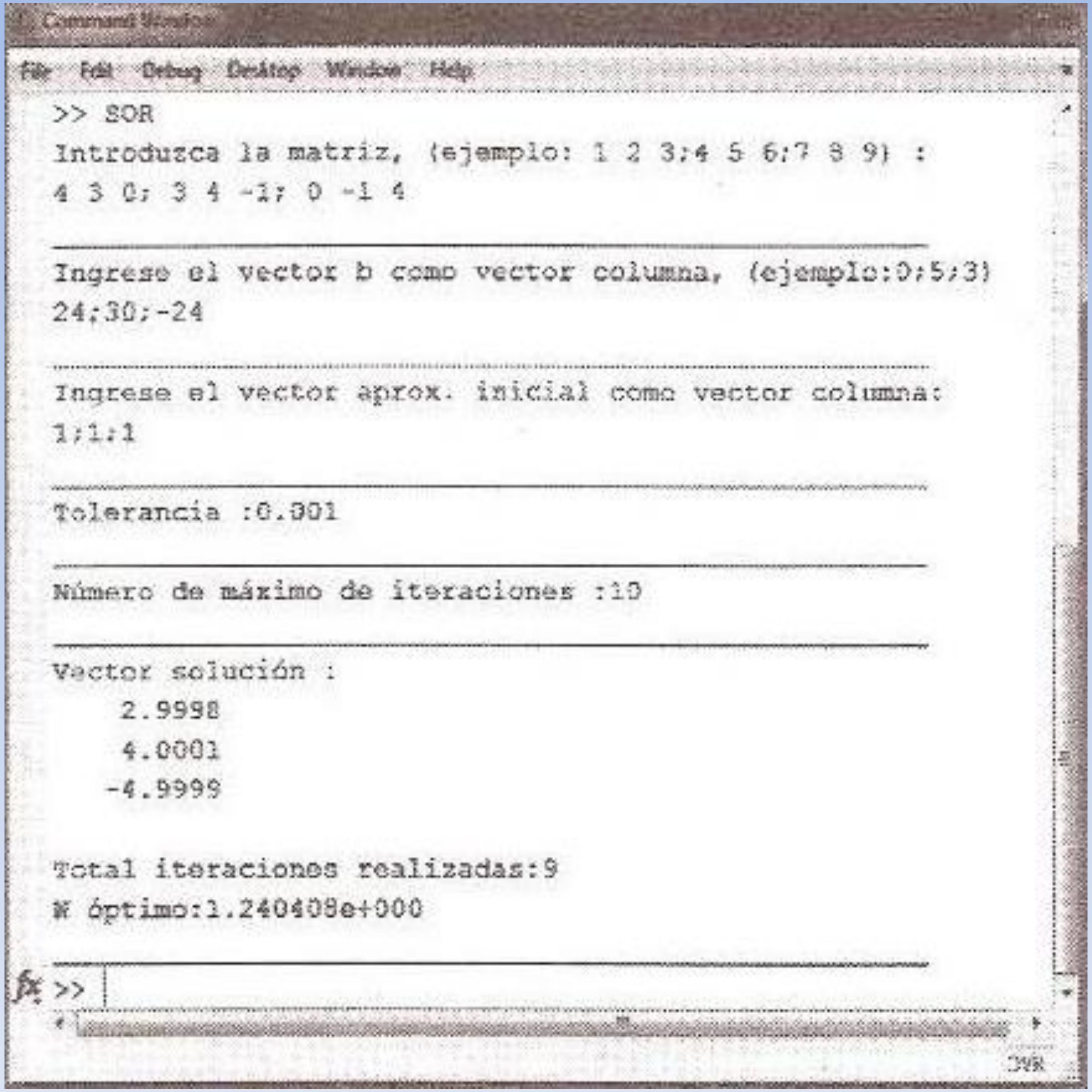
Ejemplo 1: *Dado el sistema de ecuaciones*

$4x1+3x2+0x3=243$

$x1+4x2-1x3=300$

$x1-1x2+4x3=-24$

Halle la solución utilizando el programa ***SOR.m****, la aproximación inicial es: x(0)=(1,1,1)t, una tolerancia de ε=0.001.*



```
>> SOR
Introduzca la matriz, (ejemplo: 1 2 3;4 5 6;7 8 9) :
4 3 0; 3 4 -1; 0 -1 4
_______________________________________________________
Ingrese el vector b como vector columna, (ejemplo:0;5;3)
24;30;-24
_______________________________________________________
Ingrese el vector aprox. inicial como vector columna:
1;1;1
_______________________________________________________
Tolerancia :0.001
_______________________________________________________
Número de máximo de iteraciones :10
_______________________________________________________
Vector solución :
    2.9998
    4.0001
   -4.9999

Total iteraciones realizadas:9
W óptimo:1.240408e+000
_______________________________________________________
>>
```

Repaso y Ejemplo de Jacobi:

Demostración JACOBI

$$Xi^{(K)} = \frac{1}{aii}(bi - \sum_{j\neq i;j=i}^{n} a_{ij}X_j^{k-1})$$

Dado el sistema $Ax = b$

Reemplazando $A = (D - L - U)$: $(D - L - U)x = b$

Desarrollando el lado izquierdo de la igualdad, como: $Dx - (L + U)x = b$

Si la matriz inversa D^{-1} existe, entonces se multiplica a ambos lados de la igualdad por D^{-1}: $D^{-1}Dx - D^{-1}(L + U)x = D^{-1}b$

Como $D^{-1}D = I$ se obtiene la expresión $x = D^{-1}(L + U)x + D^{-1}b$

Finalmente, si se considera una iteración k-ésima es posible obtener el origen a la forma matricial del método de Jacobi. $x^k = D^{-1}(L + U)x^{k-1} + D^{-1}b$

Cuyos límites serán la solución del sistema

Métodos Numéricos

Algoritmo de Jacobi

Paso 1: Haga k=1

Paso 2: Para cada $i = 1, \dots, n$; hacer:

$$x_i{}^k = \frac{1}{a_{ii}}\left(b_i - \sum_{j\neq i;j=1}^{n} a_{ii} x_j{}^{(k-1)}\right)$$

Paso 3: $\|x^{(k)} - x^{(k-1)}\|_\infty = \underset{i=1,\dots,n}{\text{máx}} \left\{ \left|x_0{}^{(k)} - x_0{}^{(k-1)}\right|, \dots, \left|x_n{}^{(k)} - x_n{}^{(k-1)}\right| \right\}$

Si $\|x^{(k)} - x^{(k-1)}\|\infty < \varepsilon$ vaya al paso 5

Si $\|x^{(k)} - x^{(k-1)}\|\infty > \varepsilon$ vaya al paso 4

Paso 4: Haga y vaya al paso 2.

Paso 5: Finaliza el proceso

Métodos Numéricos

Ejemplo

Dado el sistema de ecuaciones:

$$
\begin{gathered}
10_{x1} - 1_{x2} + 0_{x3} = 9 \\
-1_{x1} + 10_{x2} - 2_{x3} = 7 \\
0_{x1} - 2_{x2} + 10_{x3} = 6
\end{gathered}
$$

Halle la solución utilizando el método de Jacobi y compare con la solución exacta:

$$x_k = \left(00985789, 0.957894, 0.7915789\right)^t$$

Considere el vector de aproximación inicial dado por: $x^{(0)} = (0,0,0)^t$, una tolerancia de $\varepsilon = 0.001$. Utilice la norma infinita (l_∞).

Solucion:

Paso 1: Haga k=1

Paso 2: Para $i = 1,2,3$ y $x_i{}^{(k)} = \frac{1}{a_{ij}}\left(b_i - \sum_{j\neq i; j=1}^{n} a_{ij}x_j{}^{(k-1)}\right)$

$$x_1{}^{(k)} = \frac{1}{10}\left(9 + x_2{}^{(k-1)} - 0x_3{}^{(k-1)}\right)$$

$$x_2{}^{(k)} = \frac{1}{10}(7 + x_1{}^{(k-1)} - 2x_3{}^{(k-1)})$$

$$x_3{}^{(k)} = \frac{1}{10}(6 - 0x_1{}^{(k-1)} + 2x_2{}^{(k-1)})$$

Se obtiene la solución: $x^{(1)} = (0.90000{,}0.70000{,}0.60000)^t$.

Paso 3: Se evalúa el error: $\left\|x^{(k)} - x^{(k-1)}\right\|\infty$

$$\left\|x^{(1)} - x^{(0)}\right\|\infty = \frac{m\grave{a}x}{i=1,2,3}\{|(0.9000 - 0.00000)|, |(0.70000 - 0.00000)|, |(0.60000 - 0.00000)|\}$$

=0.90000

Como el error es mayor que la tolerancia se hace k=k+1 y se repite el Paso 2 usando el vector $x^{(1)} = (0.90000{,}0.70000, 0.60000)^t$.

Paso 2:

$$x_1{}^{(2)} = \frac{1}{10}\left(9 + x_2{}^{(k-1)} - 0x_3{}^{(k-1)}\right)$$

$$x_2{}^{(2)} = \frac{1}{10}(7 + x_1{}^{(k-1)} - 2x_3{}^{(k-1)})$$

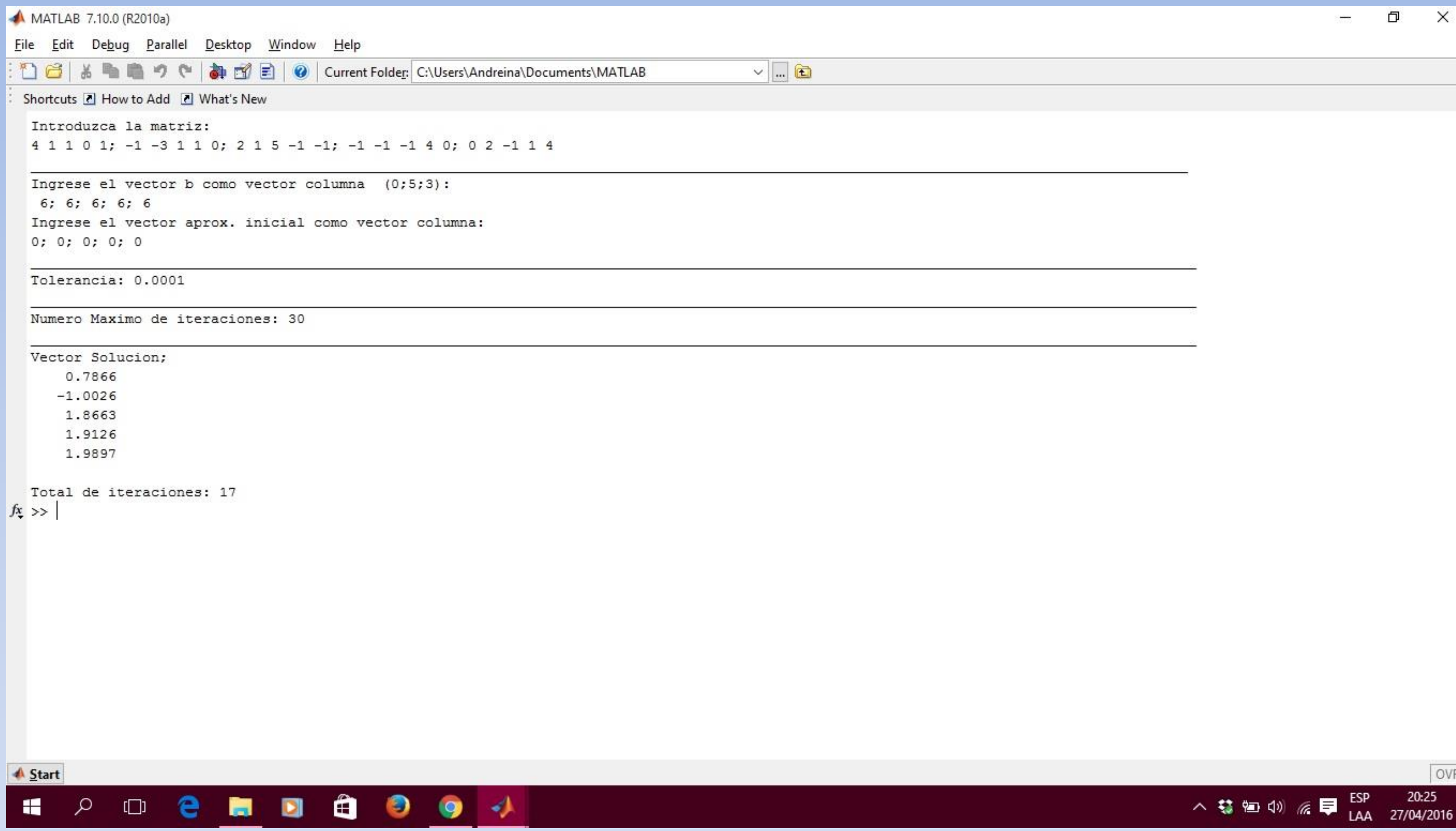
$$x_3{}^{(2)} = \frac{1}{10}(6 - 0x_1{}^{(k-1)} + 2x_2{}^{(k-1)})$$

Se obtiene el nuevo vector solución: $x^{(2)} = \left(0.970000, 0.910000, 0.740000\right)^t$.

Se evalúa el error entre los vectores $x^{(i)}\ y\ x^{(i+1)}$ para cada i para obtener la siguiente tabla de resultados

k	$x_1{}^{(k)}$	$x_2{}^{(k)}$	$x_3{}^{(k)}$	Error
0	0.000000	0.000000	0.000000	
1	0.900000	0.700000	0.600000	0.970000
2	0.970000	0.910000	0.740000	0.210000
3	00981000	0.945000	0.782000	0.042000
4	00984500	0.955500	0.789000	0.010500
5	00985550	0.957250	0.779110	0.002100
6	00985725	0.957775	0.791450	0.000525

El resultado es: $x^{(6)} = \left(00985725{,}0.957775, 0.791450\right)^t$



Métodos Numéricos