EJERCICIO N°4.

a) Es el algoritmo SOR, determinando ω.

Para ello debemos conocer:

Dado $\tilde{x} \in \mathbb{R}^n$ aproximación a la solución de $A\underline{x} = \underline{b}$

El vector residuo de $\tilde{x}$ se define:

$$\boxed{r = b - A\tilde{x}}$$ vector residual de $\tilde{x}$

El objetivo del método es generar una sucesión de aproximaciones $\{x^{(1)}, x^{(2)}, \dots x^{(n)}, \dots\}$ que hagan que los vectores residuo converjan rápidamente a cero.

$$\underline{r}_j^{(k)} = \{r_{1j}^{(k)}, r_{2j}^{(k)} \dots r_{nj}^{(k)}\}.$$ Este es el vector residuo en la iteración (k) para un sistema $A\underline{x} = \underline{b}$ de $n \times n$.

Su correspondiente vector solución será:

$$\underline{x}_j^{(k)} = [x_1^{(k)}, x_2^{(k)} \dots x_{j-1}^{(k)}, x_j^{k}, \dots x_n^{(k)}]^T.$$

La m-ésima componente de $\underline{r}_j^{(k)}$

$$r_{mi}^{(k)} = b_m - \underbrace{\sum_{j=1}^{i-1} a_{mj} x_j^{(k)}}_{\text{estos } x_j \text{ ya fueron calculados}} - \underbrace{\sum_{j=i}^{n} a_{mj} x_j^{(k-1)}}_{\text{estos faltan calcular}}$$ Gauss-Seidel.

la i-ésima componente del residuo será:

$$r_{ii}^{(k)} = b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} - \underbrace{a_{ii} x_i^{(k-1)}}_{\text{separamos el término } j=i}$$

$$a_{ii} x_i^{(k-1)} + r_{ii}^{(k)} = \underbrace{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)}}_{a_{ii} x_i^{(k)} \to \text{del método de Gauss-Seidel.}}$$

$$\Rightarrow \boxed{a_{ii} x_i^{(k-1)} + r_{ii}^{(k)} = a_{ii} x_i^{(k)}}$$

PDF-XChange Editor Click to BUY NOW! www.pdf-xchange.com

[illegible] lo tanto para el método de Gauss-Seidel.

$$\boxed{x_i^{k} = x_i^{(k-1)} + \frac{r_{ii}^{(k)}}{a_{ii}}}$$

Conexión entre los vectores residuales y el método de Gauss-Seidel.

$r_{i+1}^{(k)} \rightarrow$ asociado a $x_{i+1} = (x_1^{(k)}, x_2^{(k)}, \dots, x_i^{(k)}, x_{i+1}^{k-1}, x_{i+2}^{(k-1)}, \dots, x_n^{(k)})^T$

$\Rightarrow$ la i-ésimo componente será

$$r_{i,i+1}^{(k)} = b_i - \sum_{j=1}^{i} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)}$$

$$r_{i,i+1}^{(k)} = b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} - a_{ii} x_i^{(k)}$$

$\rightarrow r_{i,i+1}^{(k)} = 0$

$$\boxed{x_i^{(k)} = x_i^{(k-1)} + \omega \frac{r_{ii}^{(k)}}{a_{ii}}}$$

De acuerdo como elijamos ω positivo es posible reducir la norma del vector residuo y lograr una convergencia mucho más rápida.

↑ Si está involucrada esta ecuación el método se denomina MÉTODO DE RELAJACIÓN.

Si $0 < \omega < 1 \rightarrow$ SUBRELAJACIÓN

$1 < \omega < 2 \rightarrow$ SOBRERELAJACIÓN

aceleran la convergencia en métodos como Gauss-Seidel.

SOR (Successive Over Relaxation)
(sobrerelajación sucesiva)

Reformulando la ecuación anterior

$$x_i^{(k)} = (1-\omega) x_i^{(k-1)} + \frac{\omega}{a_{ii}} \left[b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} \right]$$

Al igual que los otros dos métodos determinamos la matriz de iteración $\Rightarrow$

$$a_{ii}x_i^{k} + \omega \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} = (1-\omega)\, x_i^{(k-1)} a_{ii} - \omega \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} + \omega b_i$$

$\Rightarrow$ en forma matricial o vectorial será: $\boxed{A = D - L - U}$

$$(D-\omega L)\,\underline{x}^{k} = \left[(1-\omega)D + \omega U\right] x^{(k-1)} + \omega \underline{b}$$

$$x^{(k)} = \Big\{\underbrace{(D-\omega L)^{-1}\left[(1-\omega)D+\omega U\right]}_{T_{SOR}}\Big\}\,\underline{x}^{(k-1)} + \underbrace{\omega\,(D-\omega L)^{-1}\underline{b}}_{c_{SOR}}$$

Teoremas:

1) Si $a_{ii} \neq 0$ para cada $i = 1, 2, \dots n \Rightarrow \boxed{\rho(T_{SOR}) \geq |\omega - 1|}$.
o sea el método sólo converge si $0 < \omega < 2$.

2) Si A es definida positiva ($x^T \underline{A}\, \underline{x} > 0$) y si $0 < \omega < 2 \Rightarrow$
SOR converge para cualquier vector inicial $x^{(0)} \in \mathbb{R}^n$

3) Si A es $\boxed{\text{definida positiva y tridiagonal}} \Rightarrow$ IMPORTANTE
$\rho(T_{GS}) = [\rho(T_J)]^2 < 1$ y la elección óptima de

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

con esta elección de $\omega \rightarrow$ tenemos $\rho(T_{SOR}) = \omega - 1$.

Ejemplo:

$$A = \begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix}$$

simétrica, def positiva y tridiagonal.

$$\Rightarrow T_J = D^{-1}(L+U) = \underbrace{\begin{bmatrix} 1/4 & 0 & 0 \\ 0 & 1/4 & 0 \\ 0 & 0 & 1/4 \end{bmatrix}}_{D^{-1}} \underbrace{\begin{bmatrix} 0 & -3 & 0 \\ -3 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}}_{L+U} = \underbrace{\begin{bmatrix} 0 & -0.75 & 0 \\ -0.75 & 0 & 0.25 \\ 0 & 0.25 & 0 \end{bmatrix}}_{T_J}$$

Hallemos sus autovalores =>.

$$(T_J - \lambda I) = \begin{bmatrix} -\lambda & -0.75 & 0 \\ -0.75 & -\lambda & 0.25 \\ 0 & 0.25 & -\lambda \end{bmatrix}$$

$$\det(T_J - \lambda I) = -\lambda(\lambda^2 - 0.625)$$

Autovalores

$\Rightarrow$ $\lambda = 0$

$\lambda_1 = \pm\sqrt{0.625}$

$\Rightarrow$ como $\rho(T_J) = \max_i |\lambda_i|$

$\Rightarrow \rho(T_J) = \sqrt{0.625}$ por lo que $[\rho(T_J)]^2 = 0.625$

$$\Rightarrow \omega_{op} = \frac{2}{1+\sqrt{1-(0.625)}} \cong 1.24$$

$\Rightarrow 1 < \omega < 2$ sobrerelajación

y su convergencia será mucho más rápida

4b) Dada la matriz: $\underline{X}^T A \underline{X} > 0$ verificar

$$A = \begin{bmatrix} 10 & 5 & 0 & 0 \\ 5 & 10 & -4 & 0 \\ 0 & -4 & 8 & -1 \\ 0 & 0 & -1 & 5 \end{bmatrix}$$

tridiagonal.
simetrica def - positiva
aplicamos SOR

10
5
0
0

[1 0 0 0]

"para sacar el radio espectral en Octave (Matlab)"

p = max(abs(eig(A)));

$$\underline{b} = [6 \; 25 \; -11 \; -11]^T$$

$$\Rightarrow \rho(T_{GS}) = [\rho(T_J)]^2 < 1.$$

$$y \; \omega = \frac{2}{1+\sqrt{1-[\rho(T_J)]^2}} \qquad \rho(T_{SOR}) = \omega - 1$$

$\Rightarrow$ Podemos usar eliminación Gaussiana y vemos que no aplicamos intercambio de renglones.
A es tridiagonal con $p=3$ $\Rightarrow$ por ser tridiagonal, simetrico, definido positivo $\Rightarrow$ CONVERGE

Calculemos $T_{Jacobi} \Rightarrow$

$$T_J = D^{-1}(L+U) = \underbrace{\begin{bmatrix} 1/10 & & & 0 \\ & 1/10 & & \\ & & 1/8 & \\ 0 & & & 1/5 \end{bmatrix}}_{D^{-1}} \cdot \underbrace{\begin{bmatrix} 0 & -5 & 0 & 0 \\ -5 & 0 & 4 & 0 \\ 0 & 4 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{L+U}$$

$$T_J = \begin{bmatrix} 0 & -1/2 & 0 & 0 \\ -1/2 & 0 & 2/5 & 0 \\ 0 & 1/2 & 0 & 1/8 \\ 0 & 0 & 1/5 & 0 \end{bmatrix} \Rightarrow |T_J - \lambda I| = ?$$

$$\det(T_J - \lambda I) = \det \begin{bmatrix} -\lambda & -1/2 & 0 & 0 \\ -1/2 & -\lambda & 2/5 & 0 \\ 0 & 1/2 & -\lambda & 1/8 \\ 0 & 0 & 1/5 & -\lambda \end{bmatrix}$$

$$= -\lambda \cdot \begin{bmatrix} -\lambda & 2/5 & 0 \\ 1/2 & -\lambda & 1/8 \\ 0 & 1/5 & -\lambda \end{bmatrix} - \left(-\frac{1}{2}\right) \begin{bmatrix} -1/2 & 2/5 & 0 \\ 0 & -\lambda & 1/8 \\ 0 & 0 & 1/5 & -\lambda \end{bmatrix} + 0 + 0$$

$$= -\lambda \cdot \left[-\lambda^3 + \frac{\lambda}{40} + \frac{\lambda}{5}\right] + \frac{1}{2}\left[-\frac{\lambda^2}{2} + \frac{1}{90}\right]$$

$$= \lambda^4 - \frac{9}{40}\lambda^2 - \frac{\lambda^2}{4} + \frac{1}{160} = \lambda^4 - \frac{19}{40}\lambda^2 + \frac{1}{160}$$

$$\boxed{\det[T_J - \lambda I] = 40\lambda^4 - 19\lambda^2 + \frac{1}{4} = 0}$$

$$t_1 = \frac{19 + \sqrt{321}}{80} \qquad t_2 = \frac{19 - \sqrt{321}}{80}$$

si $t = \lambda^2 \Rightarrow$

$$40t^2 - 19t + 1/4 = 0$$

$$\Rightarrow \pm\sqrt{\frac{19+\sqrt{321}}{80}} = \lambda_1, \lambda_2 \qquad \pm\sqrt{\frac{19-\sqrt{321}}{80}} \quad \lambda_3, \lambda_4$$

Sacamos el maximo $\lambda_i \Rightarrow$

$$\max_{1 \le k \le n=4} |\lambda_k| = |\lambda_1| = |\lambda_2| \cong 0.6793 = \rho(T_J)$$

...emos calcular el optimo $\Rightarrow$

$$\omega = \frac{2}{1+\sqrt{1-[\rho(T_J)]^2}} = \frac{2}{1+\sqrt{1-\frac{19+\sqrt{321}}{80}}}$$

$$\omega \cong \frac{8\sqrt{5}}{4\sqrt{5}+\sqrt{61+\sqrt{321}}} \cong 1.0034$$

$$\rho(T_{GS}) = [\rho(T_J)]^2 \cong \frac{19+\sqrt{321}}{80}$$

$$\rho(T_{SOR}) = \omega - 1 = \frac{8\sqrt{5}}{1+\sqrt{1-\frac{19+\sqrt{321}}{80}}} \cong 0.0034$$

Si aplicamos los algoritmos de SOR y G.S podemos ver q' ambos metodos llegan a la solucion en 13 iteraciones con una tolerancia de 1e-4 dando $x_1 \cong -0.79761$ $x_3 = -0.25884$

$x_2 \cong 2.79526$ $x_4 = -2.25177$

Ambos convergen con con el mismo numero se debe a que $\omega \to 1$ y para este valor particular de ω. el método SOR se convierte en el método de Gauss-Seidel.

Si SOR tiene el parametro $\omega = 1 \Rightarrow$ es Gauss-Seidel.

NOTA

— x —

SOR: Se puede interpretar como un método PREDICTOR CORRECTOR: Es decir el predictor puede ser un metodo como Jacobi o Gauss-Seidel. y el CORRECTOR realiza un promedio entre la iteración anterior y la obtenida por el PREDICTOR (Jacobi o Gauss-Seidel)

— o —

1) Obtención de SOR a través de JACOBI

Jacobi $\quad x_{i_J}^{(k)} = \frac{1}{a_{ii}}\left[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k-1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k-1)}\right] \underline{- a_{ii}\, x_i^{(k-1)}}$

$\underline{+ a_{ii}\, x_i^{(k-1)}}$

$$\Rightarrow \boxed{x_{i_J}^{k} = x_i^{(k-1)} + \frac{1}{a_{ii}}\left[b_i - \sum_{j=1}^{n} a_{ij}\, x_j^{(k-1)}\right]} \quad (1)$$

JACOBI.

SOR le aplico un coeficiente "ω" de SOBRE RELAJACION.

$$x_{i_{SOR}}^{(k)} = \omega\, x_i^{(k)} + (1-\omega)\, x_i^{(k-1)}$$

$\rightarrow$ obtenido de Jacobi (1)

$$x_{i_{SOR}}^{(k)} = (1-\omega)\, x_i^{(k-1)} + \frac{\omega}{a_{ii}}\Big[\underbrace{b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k-1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k-1)}}_{\text{JACOBI}}\Big]$$

en forma matricial

$$A = L + D + U \Rightarrow$$

$$D\, \underline{x}_{SOR}^{(k)} = (1-\omega)\, D\, \underline{x}^{(k-1)} + \omega\left[\underline{b} - (L+U)\, \underline{x}^{(k-1)}\right]$$

Es decir $\quad \boxed{\underline{x}_{SOR}^{(k)} = T_{SOR}\, \underline{x}^{(k-1)} + c_{SOR}}$

$$\Rightarrow \underline{x}^{(k)} = (1-\omega)\, \underline{x}^{(k-1)} + \omega D^{-1}\underline{b} - \omega D^{-1}(L+U)\, \underline{x}^{(k-1)}$$

$$\underline{x}^{(k)} = \underbrace{\left[(1-\omega) - \omega D^{-1}(L+U)\right]}_{T_{SOR}} \underline{x}^{(k-1)} + \underbrace{\omega D^{-1}\underline{b}}_{c_{SOR}}.$$

2) Obtenido a traves de GAUSS-SEIDEL

$$\Rightarrow x_{i\,GS}^{(k)} = \frac{1}{a_{ii}}\left[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k-1)}\right].$$

$$+a_{ii}x_i^{(k-1)} - a_{ii}x_i^{(k-1)}$$

Le sumamos y restamos.

$$\Rightarrow x_{i\,GS}^{(k)} = x_i^{(k-1)} + \frac{1}{a_{ii}}\underbrace{\left[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k-1)}\right]}_{G.S} \quad (2)$$

aplicando el factor de sobrerelajación "ω"

$$\boxed{x_{i\,SOR}^{(k)} = \omega\, x_{i\,GS}^{(k)} + (1-\omega)\, x_i^{(k-1)}}$$ anterior

~~actual (iteración (k))~~ reemplazo por (2)

$$x_{i\,SOR}^{(k)} = (1-\omega)\,x_i^{(k-1)} + \frac{\omega}{a_{ii}}\left[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k-1)}\right]$$

factorizamos o descomponemos la matriz A en -

$$\boxed{A = L + D + U.}$$

$$D\,x_{SOR}^{(k)} = (1-\omega)\,D\,x^{(k-1)} + \omega.\left[b - L\,x^{(k)} - U\,x^{(k-1)}\right]$$

agrupando:

$$\Rightarrow (D+\omega L)\,x_{SOR}^{(k)} = (1-\omega)\,D\,x^{(k-1)} + \omega\left[b - U\,x^{(k-1)}\right].$$

$$(D+\omega L)\,x_{SOR}^{(k)} = \left[(1-\omega)D - \omega U\right]x^{(k-1)} + \omega b.$$

$$x_{SOR}^{(k)} = \underbrace{(D+\omega L)^{-1}\left[(1-\omega)D - \omega U\right]}_{T_{SOR}}\, x^{(k-1)} + \underbrace{(D+\omega L)^{-1}\omega b}_{C_{SOR}}.$$

Si hubiéramos factorizado:

$\boxed{A = D - L - U.} \Rightarrow$

$$\boxed{\underline{x}^{(k)}_{SOR} = \left\{(D-\omega L)^{-1}\left[(1-\omega)D+\omega U\right]\right\}\underline{x}^{(k-1)} + \omega(D-\omega L)^{-1}\underline{b}}$$

— o —

¿Cómo formamos la ecuación del error? $\boxed{e: \text{error}}$

$$(D+\omega L)\underline{e}^{(k)} = \left[(1-\omega)D - \omega U\right]\underline{e}^{(k-1)}$$

promedio entre los errores de la iteración actual y la anterior... multiplicamos ambos miembros por D^{-1}

$$(I+\omega D^{-1}L)\underline{e}^{k} = \left[(1-\omega)I - \omega D^{-1}U\right]\underline{e}^{(k-1)}$$

$$\underline{e}^{(k)} = (I+\omega D^{-1}L)^{-1}\left[(1-\omega)I - \omega D^{-1}U\right]\underline{e}^{(k-1)}$$

$$\boxed{\|e^{(k)}\| \leq \|(I+\omega D^{-1}L)^{-1}\|\,\|[(1-\omega)I-\omega D^{-1}U]\|\,\|e^{(k-1)}\|}$$

— o —

$(I+\omega D^{-1}L)^{-1}\left[(1-\omega)I - \omega D^{-1}U\right]$. Para minimizar el error hay que minimizar la norma de esta matriz.

En el caso de (SOR) la convergencia del método es lineal. y luego de "m" iteraciones el error está acotado por:

$$\|x - x^{(k)}\| \leq C^{m}\|x - x^{(0)}\|.$$

y si se requiere una tolerancia de error relativo ε.

$\Rightarrow$ el nro de iteraciones debe ser:

$$\underbrace{\frac{\|x-x^{(k)}\|}{\|x-x^{(0)}\|}}_{\varepsilon} \leq C^{m}.$$

$$\boxed{m \geq \frac{\ln \varepsilon}{c.}} \qquad \Rightarrow \boxed{\ln \varepsilon \leq m \ln C.}$$