

Métodos Numéricos
2do Cuatrimestre 2024
Práctica 8
Métodos Iterativos



**DEPARTAMENTO
DE COMPUTACION**
Facultad de Ciencias Exactas y Naturales - UBA

1. En los siguientes casos, calcular las primeras dos iteraciones de los métodos de Jacobi y Gauss-Seidel correspondientes al sistema de ecuaciones lineales $Ax = b$, comenzando con $x^{(0)} = (0, 0, 0)^t$.

$$(a) \quad A = \begin{pmatrix} 2 & -1 & 1 \\ 3 & 3 & 9 \\ 3 & 3 & 5 \end{pmatrix}, \quad b = \begin{pmatrix} -1 \\ 0 \\ 4 \end{pmatrix} \quad (b) \quad A = \begin{pmatrix} 0 & 2 & 4 \\ 1 & -1 & -1 \\ 1 & -1 & 2 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 0,375 \\ 0 \end{pmatrix}$$

$$(c) \quad A = \begin{pmatrix} 10 & -1 & 0 \\ -1 & -1 & -2 \\ 0 & -2 & 10 \end{pmatrix}, \quad b = \begin{pmatrix} 9 \\ 7 \\ 6 \end{pmatrix}$$

2. Se define el *radio espectral* $\rho(A)$ de A por $\rho(A) := \max\{|\lambda|; \lambda \text{ es autovalor de } A\}$. Demostrar que $\rho(A) \leq \|A\|$ para cualquier norma consistente.
3. Analizar la convergencia de los métodos iterativos de Jacobi y Gauss-Seidel para la matriz

$$A = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

donde $|\rho| < 1$, comenzando con $x^{(0)} \neq (0, 0)^t$.

4. Probar que el método de Jacobi converge para sistemas de ecuaciones lineales de 2×2 cuya matriz es simétrica definida positiva.
5. Sea A una matriz de $n \times n$ tal que A se expresa en la forma $A := M - N$, donde M, N son matrices de $n \times n$ y M es no singular. Sea $R := M^{-1}N$. A fin de resolver el sistema de ecuaciones lineales $Ax = b$, dado un vector $x^{(0)} \in \mathbb{R}^n$ arbitrario consideramos la sucesión $\{x^{(k)}\}_{k \geq 0} \subset \mathbb{R}^n$ definida por $x^{(k+1)} := Rx^{(k)} + c$, donde $c = M^{-1}b$.
- a) Demostrar que si $\|R\| < 1$ para alguna norma subordinada, entonces $x^{(k)}$ converge a una solución del sistema $Ax = b$.
- b) Demostrar que si A es singular entonces $\rho(R) \geq 1$.
6. Sea $A \in \mathbb{R}^{n \times n}$. Escribimos $A = D - L - U$, donde $D, L, U \in \mathbb{R}^{n \times n}$, D es diagonal, L es triangular inferior con ceros en la diagonal y U es triangular superior con ceros en la diagonal. Demostrar que si A es estrictamente diagonal dominante por filas, entonces $\|D^{-1}(L + U)\|_\infty < 1$.
7. Sean las matrices abajo indicadas A_1 y A_2 , y sean J_1 y J_2 las matrices de iteración del método de Jacobi asociadas a A_1 y A_2 respectivamente.

$$A_1 = \begin{pmatrix} 1 & -1/4 \\ -9/10 & 1 \end{pmatrix} \quad A_2 = \begin{pmatrix} 1 & -1/3 \\ -1/2 & 1 \end{pmatrix}$$

- a) Probar que $\rho(J_1) > \rho(J_2)$
- b) Ejecutar el método de Jacobi para resolver el sistema $A_1x = b$ y $A_2x = b$ (para algún b) y comparar la cantidad de iteraciones realizadas.
- c) Concluir que una mayor dominancia diagonal¹ no necesariamente implica una convergencia más rápida del método de Jacobi.

8. El sistema de ecuaciones lineales $\begin{pmatrix} 1 & -a \\ -a & 1 \end{pmatrix} x = b$ donde $a \in \mathbb{R}$, puede resolverse bajo ciertas condiciones mediante el siguiente método iterativo:

$$\begin{pmatrix} 1 & 0 \\ -\omega a & 1 \end{pmatrix} x^{(k+1)} = \begin{pmatrix} 1-\omega & \omega a \\ 0 & 1-\omega \end{pmatrix} x^{(k)} + \omega b \quad \text{para } k = 0, \dots$$

- a) ¿Para qué valores de a converge el método cuando $\omega = 1$?
- b) Para $a = 0,5$, encontrar el valor de $\omega \in \{0,8; 0,9; 1,0; 1,1; 1,2; 1,3\}$ que minimiza el radio espectral de la matriz

$$\begin{pmatrix} 1 & 0 \\ -\omega a & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1-\omega & \omega a \\ 0 & 1-\omega \end{pmatrix}.$$

9. Sea $A \in \mathbb{R}^{n \times n}$ una matriz simétrica, con λ_i autovalores de A tal que $1 < \lambda_1 \leq \dots \leq \lambda_n$ y sea ω una constante positiva. Se define el siguiente algoritmo iterativo, para $i = 1, \dots, n$:

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

- a) Hallar el esquema de iteración de forma matricial y verificar que si el sistema iterativo converge, entonces lo hace a una solución del sistema $Ax = b$.
- b) Demostrar que el esquema iterativo planteado converge para cualquier $x^{(0)}$ inicial si y solo si $\omega < 2/\lambda_n$.
(Sugerencia: usar que si λ es autovalor de A y $\alpha, \beta \in \mathbb{R}$, entonces los autovalores de $\alpha I + \beta A$ son $\alpha + \beta \lambda$).

10. Sean $A \in \mathbb{R}^{m \times n}$ con columnas linealmente independientes, $b \in \mathbb{R}^m$ y $\omega \in \mathbb{R}$ una constante no nula. Se desea resolver el sistema $A^t A x = A^t b$ mediante un esquema iterativo. Dado un $x^{(0)}$ inicial, se propone el siguiente algoritmo:

```

 $x := x^{(0)}$ 
 $r := b - Ax^{(0)}$ 
while  $x$  no converja a la solución do
   $d := \omega A^t r$ 
   $x := x + d$ 
   $r := r - Ad$ 
end

```

¹Definimos *dominancia diagonal por filas* de una matriz $A \in \mathbb{R}^{n \times n}$ como $dd_f(A) = \max_{1 \leq i \leq n} \frac{|a_{ii}|}{\sum_{j=1}^n |a_{ij}|}$. De manera similar se define la dominancia diagonal por columnas.

- a) Probar que si el esquema iterativo converge, lo hace a una solución del sistema planteado. ¿Cuál es la matriz que gobierna la iteración del esquema? (*Sugerencia*: Probar que en cada iteración $r = b - Ax$).
- b) Demostrar que el esquema converge si y sólo si $0 < \omega < 2/\lambda_{\max}$ con λ_{\max} el mayor autovalor de la matriz $A^t A$.

11. Sea $A = QR$ la factorización QR de la matriz $A \in \mathbb{R}^{n \times n}$, con $1 < r_{11} \leq r_{22} \leq \dots \leq r_{nn}$ y $b \in \mathbb{R}^n$. Se desea hallar la solución del sistema

$$(I - Q^t A)x = b \quad (1)$$

- a) Para resolver el sistema (1), se propone el siguiente sistema iterativo, con ω una constante no nula:

$$x^{(k+1)} = ((1 - \omega)I + \omega R)x^{(k)} + \omega b, \quad k = 0, 1, \dots$$

- i) Demostrar que, si el sistema iterativo converge, entonces lo hace a una solución de (1).
- ii) Hallar los valores de ω para los cuales se puede asegurar la convergencia del método.
- b) Demostrar que Jacobi y Gauss-Seidel convergen en a lo sumo n pasos a la solución del sistema (1).

12. Se desea resolver el siguiente sistema:

$$\begin{pmatrix} A & B \\ \alpha I & C \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

donde A y C son matrices cuadradas y triangulares superiores. Se propone el siguiente algoritmo iterativo dado un $\mathbf{x}^{(0)} = (x_1^{(0)}, x_2^{(0)})^t$ inicial:

- Paso 1: Realizar una iteración de Jacobi para mejorar el valor de x_1 para el sistema $Ax_1 = b_1 - Bx_2$
 - Paso 2: Realizar una iteración de Gauss-Seidel para mejorar el valor de x_2 para el sistema $Cx_2 = b_2 - \alpha x_1$, utilizando el valor de x_1 obtenido en el Paso 1.
 - Volver al Paso 1 utilizando el x_2 hallado en el Paso 2.
- a) Escribir, para cada Paso por separado, la iteración matricial que actualiza x_1 en el Paso 1 y x_2 en el Paso 2.
- b) Escribir la iteración en forma matricial para obtener $\mathbf{x}^{(k)}$ en función de $\mathbf{x}^{(k-1)}$, siendo $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)})^t$, y mostrar cuál es la matriz que gobierna la iteración.

Resolver en computadora

- i) Resolver usando Jacobi y Gauss–Seidel, y comparar los resultados obtenidos (comenzando con $x^{(0)} = (0, 0, 0)^t$)

$$A = \begin{pmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 2 \\ 0 \\ 1 \end{pmatrix}$$

Funciones útiles

- En Matlab no existen funciones predefinidas para Jacobi ni Gauss–Seidel, pero las siguientes funciones son útiles para construirlos:

```
d = diag(A); %retorna la diagonal de la matriz A en un vector d
B = diag(d); %retorna la matriz B conteniendo en su diagonal el vector d
B = tril(A); %parte triangular inferior de la matriz A
B = triu(A); %parte triangular superior de la matriz A
```

En ambos casos, **tril** y **triu** permiten un segundo parámetro para incluir más (o menos) diagonales además de la principal, dependiendo si este parámetro es positivo (o negativo). Usando estas funciones, se pueden generar las matrices D , L y U de la separación $A = D - L - U$ con las siguientes sentencias:

```
D = diag(diag(A));
L = tril(-A, -1);
U = triu(-A, 1);
```

- En Python, existen rutinas similares de indexación con Numpy².

Referencias

- [1] R.L. Burden and J.D. Faires. *Numerical Analysis*. Brooks/Cole, Cengage Learning, 2005.
- [2] D.R. Kincaid and E.W. Cheney. *Numerical Analysis: Mathematics of Scientific Computing*. Pure and applied undergraduate texts. American Mathematical Society, 2002.
- [3] D.S. Watkins. *Fundamentals of Matrix Computations*. Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts. Wiley, 2010.

²<http://docs.scipy.org/doc/numpy/reference/routines.indexing.html>