Práctica 9

Métodos Numéricos y Computación

En esta práctica trataremos de resolver un sistema de ecuaciones lineales:

$$\left.\begin{array}{rcl} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n & = & b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n & = & b_2 \\ & \vdots & \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n & = & b_n \end{array}\right\} \tag{1}$$

que puede ser representado como $Ax = b$ donde:

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \text{ y } b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

Los métodos para la resolución de los sistemas de ecuaciones se clasifican en métodos directos o iterativos. Los métodos directos tratan de transformar el sistema de ecuaciones en otro equivalente a partir de transformaciones elementales sobre las matrices involucradas: intercambio de filas, suma de una fila más otra multiplicada por un escalar y multiplicación de una fila por un escalar. Estas operaciones también se pueden realizar por columnas si fuera necesario.

Ejercicio 1 *Construye tres funciones correspondientes a las seis operaciones elementales en una matriz. Cada una de las funciones tendrá como entrada la matriz sobre la que actuamos, la posición de las filas que vamos a manipular y el escalar (en caso de que se utilice).*

Si transformamos el sistema de ecuaciones (1) en un sistema triangular superior (método de Gauss), su solución es fácil de obtener:

$$\left.\begin{array}{rcl} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n & = & b_1 \\ a_{22}x_2 + \cdots + a_{2n}x_n & = & b_2 \\ & \vdots & \\ a_{nn}x_n & = & b_n \end{array}\right\} \Rightarrow \begin{array}{l} x_n = \frac{b_n}{a_{nn}}, \\ x_i = \frac{b_i - \sum_{j=i+1}^{n} a_{ij}x_i}{a_{ii}}, \\ i = 1, 2, \ldots, n-1. \end{array}$$

Ejercicio 2 *Implementa una función* `solucionU` *que devuelva la solución de un sistema de ecuaciones lineales con matriz de coeficientes triangular superior.*

Ejercicio 3 *Implementa una función* `gauss_parcial` *que devuelva la solución de un sistema de ecuaciones usando el método de Gauss (transformando el sistema en un sistema triangular superior) y resuelve el siguiente sistema de ecuaciones:*

$$\left.\begin{array}{rcl} x_1 + 2x_2 - 3x_3 & = & -16 \\ 3x_1 + x_2 - 2x_3 & = & -10 \\ 2x_1 - 3x_2 + x_3 & = & -4 \end{array}\right\} \tag{2}$$

Comprueba tu resultado usando la función `solve` *incluida en la librería* `scipy.linalg`*.*

Como bien sabemos, una de las aplicaciones de las transformaciones elementales es el cálculo de la inversa de una matriz. Observemos que si a una matriz invertible A se le aplican las operaciones elementales $E_1, E_2, \ldots, E_k$ (en este orden) a fin de "transformar" A en la matriz identidad, entonces:

$$E_k E_{k-1} \ldots E_1 A = I \Leftrightarrow A^{-1} = E_k E_{k-1} \ldots E_1 I,$$

o, en otras palabras, A^{-1} se obtiene aplicando las mismas transformaciones a la matriz identidad. Esto también es útil para resolver un sistema de ecuaciones pues si $Ax = b$, entonces $x = A^{-1}b$.

Ejercicio 4 *Obtén la inversa de la siguiente matriz aplicando, paso a paso, las transformaciones elementales anteriores:*

$$A = \begin{pmatrix} 0 & 1 & 2 \\ 1 & 1 & -1 \\ 2 & 1 & 0 \end{pmatrix}$$

Comprueba tu resultado usando la función `inv` *incluida en la librería* `scipy.linalg`.

Otro de los métodos directos para la resolución de un sistema de ecuaciones lineales pasa por la factorización LU de la matriz de coeficientes. Una matriz A tiene descomposición LU si se puede escribir como $A = LU$ donde L es una matriz triangular inferior y U es una matriz triangular superior. La ventaja es que si queremos resolver un sistema del tipo $Ax = b$ y $A = LU$, entonces la solución del sistema se obtiene resolviendo $Ly = b$ y, posteriormente, $Ux = y$ que son, respectivamente, triangular inferior y triangular superior.

Ejercicio 5 *Obtén la descomposición LU de la siguiente matriz:*

$$A = \begin{pmatrix} 7 & 3 & -1 & 2 \\ 3 & 8 & 1 & -4 \\ -1 & 1 & 4 & -1 \\ 2 & -4 & -1 & 6 \end{pmatrix}$$

Los métodos iterativos tratan de obtener la solución de un sistema de ecuaciones construyendo una sucesión de aproximaciones. En este caso, denotaremos como $x^{(m)} = (x_1^{(m)}, x_2^{(m)}, \ldots, x_n^{(m)})$, para $m = 0, 1, \ldots$ cada aproximación de la solución. Si el sistema se escribe como $Ax = b$ aplicaremos una transformación

$$x^{(m+1)} = Bx^{(m)} + c, \quad \text{para } m = 0, 1, \ldots, \tag{3}$$

donde $x^{(0)} = (x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)})$, donde $B \in \mathcal{M}_{n\times n}(\mathbb{R})$ y $c \in \mathbb{R}^n$ es un vector de constantes que no dependen de la iteración a realizar. Además, el punto de partida, $x^{(0)}$, será cualquier vector en $\mathbb{R}^n$.

La construcción de esta sucesión no tendría sentido de no estar seguros que converge a la solución. En particular, un método del tipo (3) será convergente cuando el método sea consistente con el sistema $Ax = b$ y el radio espectral de B sea menor que 1. Algunos métodos muy utilizados son los de Jacobi y Gauss-Seidel. La construcción de esta sucesión finalizará cuando o bien se alcance un número máximo de iteraciones, o bien tengamos que $||x^{(m+1)} - x^{(m)}|| < \varepsilon$ para un $\varepsilon > 0$ fijado que es conocido como la tolerancia.

Ejercicio 6 *Implementa el método de Jacobi como una función* `jacobi` *en la que tomaremos como parámetros de entrada la matriz de coeficientes, el vector de términos independientes, el punto semilla, el tipo de norma a utilizar, la tolerancia y el número máximo de iteraciones que puede realizar el método. Además, la función deberá comprobar si el método es convergente o no y, en caso de que sea convergente, devolverá la aproximación de la solución, el número de pasos necesarios y el valor del radio espectral.*

Ejercicio 7 *Aplica el método de Jacobi para aproximar las soluciones de:*

$$a)\quad \left.\begin{array}{rcl} x_1+2x_2-3x_3 & = & -16 \\ 3x_1+x_2-2x_3 & = & -10 \\ 2x_1-3x_2+x_3 & = & -4 \end{array}\right\}$$

$$b)\quad \left.\begin{array}{rcl} 5x_1+x_2+2x_3 & = & 1 \\ x_1+4x_2+x_3 & = & 2 \\ 2x_1+2x_2+5x_3 & = & 2 \end{array}\right\}$$

Observación: Toma como $x^{(0)}$ *un vector de tantos ceros como sea necesario, considera la norma infinito, cien pasos como máximo y una tolerancia de* 10^{-5}.

Ejercicio 8 *Obtén la solución del siguiente sistema de ecuaciones:*

$$\begin{array}{ccccccccccccc}
12m_1 & - & 2m_2 & + & m_3 & & & & & & & = & 5 \\
-2m_1 & + & 12m_2 & - & 2m_3 & + & m_4 & & & & & = & 5 \\
m_1 & - & 2m_2 & + & 12m_3 & - & 2m_4 & + & m_5 & & & = & 5 \\
 & & m_2 & - & 2m_3 & + & 12m_4 & - & 2m_5 & + & m_6 & = & 5 \\
 & & & & & & & & & & & \vdots & \\
 & & m_{46} & - & 2m_{47} & + & 12m_{48} & - & 2m_{49} & + & m_{50} & = & 5 \\
 & & & & m_{47} & - & 2m_{48} & + & 12m_{49} & - & 2m_{50} & = & 5 \\
 & & & & & & m_{48} & - & 2m_{49} & + & 12m_{50} & = & 5
\end{array}$$

a partir de:

a) La función `solve`.

b) El método de Gauss.

c) El método de Jacobi tomando como $x^{(0)}$ *un vector de ceros, la norma infinito, cien pasos como máximo y una tolerancia de* 10^{-5}.

Ejercicios para entregar

Crea un script con nombre `PrimerApellido_SegundoApellido_Nombre.py` donde resuelvas los siguientes ejercicios teniendo en cuenta las soluciones de los anteriores y los contenidos de las presentaciones de la teoría. Incluye todos aquellos comentarios que consideres oportunos. La entrega deberá realizarse de 16:00 a 20:00 horas del viernes 3 de abril desde la tarea habilitada en Moodle dentro del Tema 5 con nombre `Práctica 9 - Ejercicios para entregar`.

Ejercicio 1 *Implementa una función* `solucionL` *que devuelva la solución de un sistema de ecuaciones lineales con matriz de coeficientes triangular inferior y, úsala junto con* `solucionU` *y la descomposición LU, para resolver el siguiente sistema de ecuaciones:*

$$\left.\begin{array}{rcl} 15x+50y+67z & = & 1 \\ 10x+20y+23z & = & 0 \\ 5x+6y+7z & = & -1 \end{array}\right\}$$

Ejercicio 2 *Implementa una función* `gaussjordan_total` *que devuelva la solución de un sistema de ecuaciones usando el método de* ***Gauss-Jordan*** *(transformando el sistema en un sistema diagonal) con pivoteo* ***total*** *y resuelve el siguiente sistema de ecuaciones:*

$$\left.\begin{array}{rcl} x+2y-z+3t & = & -8 \\ 2x+2z-t & = & 13 \\ -x+y+z-t & = & 8 \\ 3x+3y-z+2t & = & -1 \end{array}\right\}$$

Comprueba tu resultado usando la función `gauss_parcial` *y la función* `solve` *incluida en la librería* `scipy.linalg`.

Ejercicio 3 *Implementa una función* `inv_parcial` *que devuelva la inversa de una matriz usando el pivoteo que prefieras (especifícalo en el script) y comprueba tu función con la matriz:*

$$A = \begin{pmatrix} 2 & -1 & -1 & 0 \\ -1 & 1 & 0 & -1 \\ -1 & 0 & 2 & -1 \\ 0 & -1 & -1 & 1 \end{pmatrix}$$

Ejercicio 4 *Implementa el método de Gauss-Seidel como una función* `gauss_seidel` *en la que tomaremos como parámetros de entrada la matriz de coeficientes, el vector de términos independientes, el punto semilla, el tipo de norma a utilizar, la tolerancia, y el número máximo de iteraciones que puede realizar el método. Además, la función deberá comprobar si el método es convergente o no y, en caso de que sea convergente, devolverá la aproximación de la solución, el número de pasos necesarios y el valor del radio espectral.*

Ejercicio 5 *Aplica los métodos de Jacobi y Gauss-Seidel para aproximar las soluciones de:*

a)
$$\left.\begin{array}{rcl} 5x_1+x_2+2x_3 & = & 1 \\ x_1+4x_2+x_3 & = & 2 \\ 2x_1+2x_2+5x_3 & = & 2 \end{array}\right\}$$

b)
$$\left.\begin{array}{rcl} x+2y-z+3t & = & -8 \\ 2x+2z-t & = & 13 \\ -x+y+z-t & = & 8 \\ 3x+3y-z+2t & = & -1 \end{array}\right\}$$

¿Qué ocurre con el sistema de ecuaciones del apartado (b)? Observación: Toma como $x^{(0)}$ un vector ceros, considera la norma infinito, cien pasos como máximo y una tolerancia de 10^{-5}.

Ejercicio 6 *Obtén la solución del sistema de ecuaciones $Ax = b$ donde A es una matriz de orden n, $A = (a_{ij})$, $i, j = 0, 1, 2, \ldots, n-1$, cuyos coeficientes vienen dados por:*

$$a_{ij} = \begin{cases} (i+1)(n-j), & \text{si } i \leq j, \\ a_{ji}, & \text{si } i > j, \end{cases}$$

y b es el vector de términos independientes $b = (1, 2, \ldots, n)$, para $n = 9$. Utiliza para ello:

a) *La función* `solve`.

b) *El cálculo de la inversa de la matriz de los coeficientes.*

c) *El método de Gauss.*

d) *Los métodos de Jacobi y Gauss-Seidel tomando como $x^{(0)}$ un vector de ceros, considerando la norma infinito, cien pasos como máximo y una tolerancia de 10^{-5}.*