Práctica II: Resolución de sistemas lineales

Métodos Numéricos I - Doble Grado en Ingeniería Informática y Matemáticas - Universidad de Granada - Curso 2016_2017

Métodos directos

Aunque no es objeto de la asignatura, puede ser útil reparar en que el comando **LinearSolve** resuelve un sistema de ecuaciones lineales compatible determinado. Solamente hay que introducir, como argumentos, la matriz de coeficientes y el vector de términos independientes:

```
LinearSolve[{{1, 1}, {1, 3}}, {1, 0}]
```

La sentencia **Solve** permite resolver un sistema de ecuaciones lineales compatible, ya sea determinado o indeterminado, auque las ecuaciones se introducen directamente:

```
Solve[{x + y == 1, x + 3 y == 0}, {x, y}]
```

```
Solve[{x + 2 y + 3 z == 0, 4 x + 5 y + 6 z == 0, 7 x + 8 y + 9 z == 0}, {x, y, x}]
```

Con las herramientas de que dispones, pudes programar el método de Gauss, y de hecho, aparece propuesto como el segundo ejercicio.

Con relación a los métodos de factorización LU como Doolittle y Crout, *Mathematica* cuenta con el comando **LUDecomposition**, que aplicado a una matriz regular a da como salida su factorización LU tipo Doolittle. El primer argumento de la salida da una matriz S que, de forma compacta, no es más que la factorización LU de a: L es la matriz triangular inferior con 1's en la diagonal principal y con la parte inferior de S, y U es la matriz triangular superior con la diagonal principal de S y la parte superior de S. El segundo argumento indica la eventual permutación de filas.

```
a = {{2, 2, 1, 2}, {4, 6, 1, 3}, {6, 12, 1, 3}, {2, 4, -1, -1}}
```

```
LUDecomposition[a]
```

Es decir, la matriz *a* admite la factorización tipo Doolitle $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 3 & 1 & 0 \\ 1 & 1 & -1 & 1 \end{pmatrix}\begin{pmatrix} 2 & 2 & 1 & 2 \\ 0 & 2 & -1 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix}$.

Por su parte, la sentencia **LUBackSubstitution** devuelve la solución del sistema ax=b a partir de la factorización LU de a que acabamos de obtener:

```
b = {1, -2, 3, -4}
```

```
LUBackSubstitution[LUDecomposition[a], b]
```

Métodos iterativos

Los métodos iterativos de resolución de sistemas de ecuaciones lineales pueden implementarse mediante el uso de sentencias trabajadas en la práctica anterior, y así se propone en los ejercicios.

Ejercicios

- Programa la resolución de un sistema triangular superior compatible determinado. Aplícalo al sistema de matriz de coeficientes $\begin{pmatrix} 2.45 & 1.23 & 2.59 & 0 \\ 0 & 1.1 & -1.7 & 3 \\ 0 & 0 & 2.41 & 8/9 \\ 0 & 0 & 0 & 0.82 \end{pmatrix}$ y vector de términos independientes $\begin{pmatrix} 1 \\ 1/2 \\ 1/3 \\ 1/4 \end{pmatrix}$.
- Programa el método de Gauss y úsalo para resolver el sistema con matriz de coeficientes $\begin{pmatrix} 2 & 2.1 & 5.09 & 0.1 \\ 0 & 1.1 & -1.7 & 3.2 \\ 1 & 1 & -3.1 & 8.9 \\ -5.23 & 1 & 4.27 & 0.82 \end{pmatrix}$ y vector de términos independientes $\begin{pmatrix} 0.1 \\ -0.2 \\ 0.33 \\ 0.44 \end{pmatrix}$.
- Programa el método de Crout y aplícalo para encontrar la solución del sistema con matriz de coeficientes y vector de términos independientes, respectivamente, $\begin{pmatrix} 1 & 2 & -4 \\ 1.1 & 3.2 & -5.4 \\ 3.1 & 9.4 & -14.6 \end{pmatrix}$ y $\begin{pmatrix} 2.1 \\ 1.2 \\ 2.2 \end{pmatrix}$.
- Programa los métodos de Jacobi y Gauss-Seidel y aplícalos, partiendo de la iteración inicial $\begin{pmatrix} 1.01 \\ -0.76 \\ 2.2 \end{pmatrix}$ y realizando 16 iteraciones, para obtener una aproximación de la solución del sistema con matriz de coeficientes $\begin{pmatrix} 2 & -0.75 & 0.65 \\ -0.97 & 8 & -3.1 \\ 2 & -3 & -5.4 \end{pmatrix}$ y vector de términos independientes $\begin{pmatrix} 1/6 \\ -1/16 \\ 3/\pi \end{pmatrix}$.
- Considera una matriz de orden 2×2 con coeficientes reales no nulos. Demuestra que el correspondiente método de Gauss-Seidel es convergente si, y solo si, el radio espectral asociado al método de Gauss-Seidel es estrictamente menor que el de Jacobi.