Càlcul numèric
Eduardo Pérez Motato. (30/05/2024)

Pràctica 3
NIU: 1709992

Exercici 1:

a) Desembolupar un codi que proporcioni la descomposició LU d'una matrui arbitraria de dimensió máxima 5×5.

Solució. Fet a `Pr3Ex1a.c`. La funció `LU` demana la dimensió, una matriu de la dimensió i un vector d'aquesta dimensió i fa la descomposició LU amb pivotatge màximal amb reemplaçament (es dir, la matriu L_A i U_A es guarden a l'espai de memoria de A, destruint A) guardant els pivots al vector. □

b) Obtenir la descomposició LU de les matrius A i B, calcular els seus determinants

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 4 & 9 & 16 \\ 1 & 8 & 27 & 64 \\ 1 & 16 & 81 & 256 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 \\ 0 & 1 & 3 & 1 & 0 \\ 0 & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 1 & 5 \end{pmatrix}$$

Solució. Mitjançants la funció `LU` de l'apartat anterior, es posa la matriu A i la matriu B. S'imprimeix per pantalla aquestes matrius i després s'imprimeix la matriu A_L, U_A i el vector de permutacions de A (després el programa fa el mateix amb B). Finalment dona els determinants, calculats multiplicant la diagonal de U_A (com és amb reemplaçament, es calcula la diagonal a l'espai A). Al ser amb permutacions, es calculen el nombre de permutacions (més bé, es calcula el nombre de posicions no correctes n). Si $n \neq 0$, s'ha de multiplicar per -1 i si $n \mod 2 \equiv 1$, llaors també s'ha de multiplicar per -1. Ambdues condicions convinades es té el veritable determinant.

$$L_A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0.428571 & 1 & 0 \\ 1 & 0.142857 & 0.545455 & 0 \end{pmatrix} \quad L_B = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0.5 & 1 & 0 \\ 0 & 0 & 0 & 0.285714 & 1 \end{pmatrix}$$

$$U_A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & 14 & 78 & 252 \\ 0 & 0 & -9.428571 & -48 \\ 0 & 0 & 0 & 2.181818 \end{pmatrix} \quad U_B = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 2 & 1 & 0 \\ 0 & 0 & 0 & 3.5 & 1 \\ 0 & 0 & 0 & 0 & 4.714286 \end{pmatrix}$$

$$Perm(A) = \begin{pmatrix} 0 \\ 3 \\ 2 \\ 1 \end{pmatrix} \quad Perm(B) = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$$

$$\det A = 288 \;\; \det B = 33$$

□

Exercici 2:

a) Desembolupar un codi per resoldre un sistema lineal pel metode de Jacobi.
Solució. Fet a `Pr3Ex2a.c`, on soluciona $\begin{pmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{pmatrix} x = \begin{pmatrix} 6 \\ 20 \\ 4 \end{pmatrix}$. La funció `Jacobi` demana el nombre de dimensió, una matriu quadrada d'aquestes dimensions, un vector de solucions (on es guardaran, pot no estar inicialitzat) i un vector també de la dimensió abans demanada al qual equival aquesta equació. Demana també una tolerancia i un nombre màxim d'iteracions. Retorna 1 o 0, segons si ha assolit el nombre máxim d'iteracions (sent 1 la resposta afirmativa). També imprimeix per pantalla el nombre d'iteracions. Ha estat altament inspirat en el pseudocodi d'aquest diaporama a la diapositiva 104. □

b) Desembolupar un codi per resoldre un sistema lineal pel metode de Gauss-Seidel.
Solució. Fet a `Pr3Ex2a.c`, on soluciona $\begin{pmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{pmatrix} x = \begin{pmatrix} 6 \\ 20 \\ 4 \end{pmatrix}$. La funció `GaussSeidel` demana el nombre de dimensió, una matriu quadrada d'aquestes dimensions, un vector de solucions (on es guardaran, pot no estar inicialitzat) i un vector també de la dimensió abans demanada al qual equival aquesta equació. Demana també una tolerancia i un nombre màxim d'iteracions. Retorna 1 o 0, segons si ha assolit el nombre máxim d'iteracions (sent 1 la resposta afirmativa). També imprimeix per pantalla el nombre d'iteracions. Ha estat altament inspirat en el pseudocodi d'aquest diaporama a la diapositiva 114. □

c) Calcular amb un error menor que 10^{-5} la solució del sistema mitjançants els metodes de Jacobi i Gauss-Seidel. Comparar el nombre d'iteracions.

$$\begin{cases} 10x_1 - x_2 + 2x_3 = 6 \\ -x_1 + 11x_2 - x_3 + 3x_4 = 25 \\ 2x_1 - x_2 + 10x_3 - x_4 = -11 \\ 3x_2 - x_3 + 8x_4 = 15 \end{cases}$$

Solució. Ambdues funcions retornen un resoltat admisible per l'error demanat. Jacobi ho retorna després de 5 iteracions mentre Gauss-Seidel ho retorna després de 7. Això denota que Jacobi és més rápid en aquest cas, malgrat de normal Gauss-Seidel convergeix més ràpid. □