Matrices triangulares y descomposición LU

Problemas para examen

Si en algún problema se pide calcular el número de flops (operaciones aritméticas con punto flotante), entonces en el examen será suficiente calcular el número de las multiplicaciones y divisiones (juntas).

Matrices de permutación

Definición (matriz de permutación). Sea $\varphi \in S_n$. Entonces la matriz $P_\varphi \in \mathcal{M}_n(\mathbb{R})$ se define de la siguiente manera:

$$P_\varphi = \left[\delta_{\varphi(i),j}\right]_{i,j=1}^{n}.$$

1. Producto de una matriz de permutación por un vector. Sean $\varphi \in S_n$, $v \in \mathbb{R}^n$ y $j \in \{1, \ldots, n\}$. Deduzca una fórmula para $(P_\varphi v)_j$.

2. Programación: producto de una matriz de permutación por un vector. Escribir una función que calcule el producto $P_s v$, donde $s \in S_n$ y $v \in \mathbb{R}^n$. Los argumentos de la función son los vectores $s = (s(1), \ldots, s(n))$ y v.

3. Producto de matrices de permutación. Sean $\varphi, \psi \in S_n$. Demuestre que

$$P_\varphi P_\psi = P_{\psi\varphi}.$$

Indicaciones:

- Pruebe la fórmula para $n = 5$, con dos permutaciones que no conmutan.
- En el caso general deduzca una fórmula para $(P_\varphi P_\psi)_{p,q}$, usando la definición del producto de matrices, la definición de la matriz de permutación y la regla para simplificar sumas con la delta de Kronecker.

4. Producto de una matriz de permutación por una matriz general (permutación de los renglones de una matriz). Sea $A \in \mathcal{M}_{m\times n}(\mathbb{R})$ y sea $\varphi \in S_m$. Deduzca una fórmulas para las entradas de la matriz $P_\varphi A$. En algún lenguaje de programación escriba una función que lo haga.

5. Programación: permutación de los renglones de una matriz. Escriba una función que calcule la matriz $B = \left[A_{\varphi(j),k}\right]_{j,k=1}^{m,n}$, donde $A \in \mathcal{M}_{m\times n}(\mathbb{R})$ es una matriz dada y $\varphi \in S_m$ es una permutación dada como un arreglo $\varphi(1), \ldots, \varphi(m)$.

6. Producto de una matriz general por una matriz de permutación. Sea $A \in \mathcal{M}_{m\times n}(\mathbb{R})$ y sea $\varphi \in S_n$. Explique cómo construir la matriz AP_φ a partir de la matriz A. En algún lenguaje de programación escriba una función que lo haga.

Matrices diagonales y su multiplicación por vectores

7. Denotemos por $\mathcal{D}_n(\mathbb{R})$ al conjunto de las matrices reales diagonales $n \times n$. Escriba la definición formal del conjunto $\mathcal{D}_n(\mathbb{R})$.

8. Matriz diagonal generada por un vector. Sea $a \in \mathbb{R}^n$. Se denota por $\operatorname{diag}(a)$ la matriz diagonal con entradas diagonales $a_1, \ldots, a_n$:

$$\operatorname{diag}(a) = \left[a_j \delta_{j,k}\right]_{j,k=1}^{n}.$$

9. Producto de una matriz diagonal por un vector. Sean $d, v \in \mathbb{R}^n$. Demuestre que

$$\operatorname{diag}(d) v = d \odot v.$$

Recordamos que $\odot$ denota le producto de vectores por componentes.

10. Operaciones con matrices diagonales. Sean $a, b \in \mathbb{R}^n$, $\lambda \in \mathbb{R}$. Haga las siguientes operaciones con matrices diagonales (enuncie y demuestre las fórmulas):

$$\operatorname{diag}(a) + \operatorname{diag}(b), \qquad \lambda \operatorname{diag}(a), \qquad \operatorname{diag}(a) \operatorname{diag}(b).$$

11. Producto de una matriz diagonal por una matriz general. Sean $d \in \mathbb{R}^m$, $A \in \mathcal{M}_{m \times n}(\mathbb{R})$. Deduzca una fórmula para las entradas de la matriz $\operatorname{diag}(d) A$.

12. Producto de una matriz general por una matriz diagonal. Sean $d \in \mathbb{R}^n$, $A \in \mathcal{M}_{m \times n}(\mathbb{R})$. Deduzca una fórmula para las entradas de la matriz $A \operatorname{diag}(d)$.

Matrices triangulares

13. Denotemos por $\mathfrak{ut}_n(\mathbb{R})$ al conjunto de las matrices triangulares superiores y por $\mathfrak{lt}_n(\mathbb{R})$ al conjunto de las matrices triangulares inferiores:

$$\mathfrak{ut}_n(\mathbb{R}) := \Big\{ A \in \mathcal{M}_n(\mathbb{R}) \colon \quad \forall i, j \in \{1, \ldots, n\} \quad i > j \quad \Rightarrow \quad A_{i,j} = 0 \Big\};$$

$$\mathfrak{lt}_n(\mathbb{R}) := \Big\{ A \in \mathcal{M}_n(\mathbb{R}) \colon \quad \forall i, j \in \{1, \ldots, n\} \quad i < j \quad \Rightarrow \quad A_{i,j} = 0 \Big\}.$$

Encuentre la intersección $\mathfrak{ut}_n(\mathbb{R}) \cap \mathfrak{lt}_n(\mathbb{R})$.

14. Programación: producto de una matriz triangular inferior por un vector. Escriba una función que calcule el producto de una matriz triangular inferior por un vector aprovechando la estructura de la matriz triangular. En otras palabras, hay que omitir las operaciones con las entradas que son nulas por la definición de matriz triangular inferior. Calcule el número de flops.

15. Programación: producto una matriz triangular superior por un vector. Escriba una función que calcule el producto de una matriz triangular superior por un vector aprovechando la estructura de la matriz triangular. En otras palabras, hay que omitir las operaciones con las entradas que son nulas por la definición de matriz triangular superior. Calcule el número de flops.

16. Producto de matrices triangulares superiores.
Sean $A, B \in \mathfrak{ut}_n(\mathbb{R})$. Demuestre que $AB \in \mathfrak{ut}_n(\mathbb{R})$. Enuncie y demuestre la fórmula para las entradas del producto AB. En particular, enuncie una fórmula para las entradas diagonales del producto AB.

17. Producto de matrices triangulares inferiores.
Sean $A, B \in \mathfrak{lt}_n(\mathbb{R})$. Demuestre que $AB \in \mathfrak{lt}_n(\mathbb{R})$. Enuncie y demuestre la fórmula para las entradas del producto AB. En particular, enuncie una fórmula para las entradas diagonales del producto AB.

18. Programación: producto de matrices triangulares superiores. Escriba una función que calcule el producto de matrices triangulares superiores aprovechando su estructura. En otras palabras, hay que omitir las operaciones con las entradas que son nulas por la definición de matriz triangular superior. Calcule el número de flops.

19. Programación: producto de matrices triangulares inferiores. Escriba una función que calcule el producto de matrices triangulares inferiores aprovechando su estructura. En otras palabras, hay que omitir las operaciones con las entradas que son nulas por la definición de matriz triangular inferior. Calcule el número de flops.

20. Criterio de invertibilidad de matrices triangulares superiores. Sea $A \in \mathfrak{ut}_n(\mathbb{R})$. Muestre que A es invertible si y sólo si todas sus entradas diagonales son no nulas.

21. Criterio de invertibilidad de matrices triangulares inferiores. Sea $A \in \mathfrak{lt}_n(\mathbb{R})$. Muestre que A es invertible si y sólo si todas sus entradas diagonales son no nulas.

22. Teorema sobre la inversa de una matriz unitriangular inferior. Sea A una matriz unitriangular inferior de orden n. Muestre que A^{-1} también es unitriangular inferior. Sugerencia: encontrar una cadena de operaciones elementales que transforman A en I_n, luego expresar A^{-1} como un producto de ciertas matrices elementales y analizar su forma. Otra demostración: suponer que $AB = I_n$, analizar las entradas de AB en un orden adecuado y hacer conclusiones sobre las entradas de B.

23. Teorema sobre la inversa de una matriz triangular superior. Sea A una matriz triangular superior de orden n con entradas diagonales no nulas. Muestre que A^{-1} también es triangular superior. Sugerencia: encontrar una cadena de operaciones elementales que transforman A en I_n, luego expresar A^{-1} como un producto de ciertas matrices elementales y analizar su forma. Otra demostración: suponer que $AB = I_n$, analizar las entradas de AB en un orden adecuado y hacer conclusiones sobre las entradas de B.

24. Programación: inversión de una matriz unitriangular inferior (problema optativo). Escriba una función que calcule la inversa de la matriz dada L, donde L es una matriz cuadrada unitriangular inferior. Calcule el número de flops.

25. Programación: inversión de una matriz triangular superior (problema optativo). Escriba una función que calcule la inversa de la matriz dada L, donde L es una matriz cuadrada triangular superior. Calcule el número de flops.

Solución de sistemas de ecuaciones lineales, casos simples

26. Programación: solución de sistemas de ecuaciones lineales con matrices unitriangulares inferiores, método por renglones. Escriba una función que resuelva sistemas de ecuaciones lineales de la forma $Lx = b$, donde $L \in \mathrm{LT}_n(\mathbb{R})$ es una matriz unitriangular inferior y $b \in \mathbb{R}^n$. Utilice la función "producto punto" programada anteriormente o su análogo en el lenguaje de programación elejido. Aproveche la estructura de la matriz y trabaje solamente con los "trozos interesantes" de renglones o columnas. Calcule el número de flops.

27. Programación: solución de sistemas de ecuaciones lineales con matrices unitriangulares inferiores, método por columnas. Escriba una función que resuelva sistemas de ecuaciones lineales de la forma $Lx = b$, donde $L \in \mathrm{LT}_n(\mathbb{R})$ es una matriz unitriangular inferior y $b \in \mathbb{R}^n$. Utilice operaciones lineales con los "trozos interesantes" de columnas. Se recomienda usar la función "axpy" programada anteriormente o su análogo en el lenguaje de programación elejido. Calcule el número de flops.

28. Programación: solución de sistemas de ecuaciones lineales con matrices triangulares inferiores. Escriba una función que resuelva sistemas de ecuaciones lineales de la forma $Lx = b$, donde $L \in \mathrm{LT}_n(\mathbb{R})$ es una matriz triangular inferior y $b \in \mathbb{R}^n$. Resuelva el problema de dos maneras: por renglones (con el producto punto) y por columnas (con operaciones lineales). Calcule el número de flops.

29. Programación: solución de sistemas de ecuaciones lineales con matrices triangulares superiores. Escriba una función que resuelva sistemas de ecuaciones lineales de la forma $Lx = b$, donde $L \in \mathrm{UT}_n(\mathbb{R})$ es una matriz triangular superior y $b \in \mathbb{R}^n$. Resuelva el problema de dos maneras: por renglones (con el producto punto) y por columnas (con operaciones lineales). Calcule el número de flops.

30. Programación: sistemas tridiagonales de ecuaciones lineales.
Escriba una función que resuelva sistemas tridiagonales de ecuaciones lineales, usando

pivotes diagonales. Los datos iniciales son los vectores a, b, c, r de longitudes n, $n-1$, $n-1$, n, respectivamente. Por ejemplo, para $n=5$, se trata del sistema

$$\left[\begin{array}{ccccc|c} a_1 & b_1 & 0 & 0 & 0 & r_1 \\ c_1 & a_2 & b_2 & 0 & 0 & r_2 \\ 0 & c_2 & a_3 & b_3 & 0 & r_3 \\ 0 & 0 & c_3 & a_4 & b_4 & r_4 \\ 0 & 0 & 0 & c_4 & a_5 & r_5 \end{array}\right].$$

Calcule el número de flops.

Factorizaciones LU, Cholesky y PLU

31. Fórmulas del p-ésimo paso de la factorización LU. Consideramos el algoritmo de factorización LU. Sea n el orden de la matriz y sea $p \in \{1,\ldots,n-1\}$. Denotemos por L y U a las matrices obtenidas después de $p-1$ pasos del algoritmo. Por ejemplo, si $n=6$, $p=3$, entonces L y U son de la forma

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ L_{2,1} & 1 & 0 & 0 & 0 & 0 \\ L_{3,1} & L_{3,2} & 1 & 0 & 0 & 0 \\ L_{4,1} & L_{4,2} & 0 & 1 & 0 & 0 \\ L_{5,1} & L_{5,2} & 0 & 0 & 1 & 0 \\ L_{6,1} & L_{6,2} & 0 & 0 & 0 & 1 \end{bmatrix}, \qquad U = \begin{bmatrix} U_{1,1} & U_{1,2} & U_{1,3} & U_{1,4} & U_{1,5} & U_{1,6} \\ 0 & U_{2,2} & U_{2,3} & U_{2,4} & U_{2,5} & U_{2,6} \\ 0 & 0 & U_{3,3} & U_{3,4} & U_{3,5} & U_{3,6} \\ 0 & 0 & U_{4,3} & U_{4,4} & U_{4,5} & U_{4,6} \\ 0 & 0 & U_{5,3} & U_{5,4} & U_{5,5} & U_{5,6} \\ 0 & 0 & U_{6,3} & U_{6,4} & U_{6,5} & U_{6,6} \end{bmatrix}.$$

Escriba las operaciones que se aplican a las matrices L y U en el p-ésimo paso del algoritmo. Se recomienda no hacer operacions con las entradas que ya definitivamente son cero. Además se recomienda poner ceros en las posiciones $U_{4,3}$, $U_{5,3}$, $U_{6,3}$, en vez de calcular sus valores nuevos con las operaciones elementales.

32. Fórmulas del p-ésimo paso de la factorización LU en forma compacta. Consideremos la versión compacta del algoritmo LU, cuando las entradas no triviales de la matriz L se guardan en la parte inferior de la matriz U. Modifique la solución del problema anterior de manera adecuada.

33. Programación: factorización LU. Escriba una función que calcule la factorización LU de una matriz dada. Puede suponer que dicha factorización existe. La función puede regresar el par de matrices (L,U) o una matriz que guarde ambos factores L y U de manera compacta. Utilice las fórmulas e ideas de los Problemas 31 y 32. Calcule el número de flops.

34. Teorema: unicidad de la factorización LU. Sea A una matriz cuadrada invertible. Demuestre que si A tiene una factorización LU, entonces esta factorización es única.

35. Definición: matriz estrictamente invertible. Una matriz cuadrada A de orden n se llama *estrictamente invertible* si para cada $k \in \{1, \ldots, n\}$ la submatriz $A_{(1,\ldots,k),(1,\ldots,k)}$ de la matriz A, ubicada en la intersección de sus primeros k renglones y k columnas, es invertible. En otras palabras, A es estrictamente invertible si todos sus menores principales líderes son distintos de cero.

36. Lema sobre las submatrices principales líderes del producto de una matriz triangular inferior por una matriz triangular superior. Sea $A = LU$, donde L es una matriz triangular inferior y U es una matriz triangular superior, todas de tamaño $n \times n$. Demuestre que para cada $p \in \{1, \ldots, n\}$

$$A(1:p, 1:p) = L(1:p, 1:p)U(1:p, 1:p).$$

37. Lema sobre los menores principales líderes en la eliminación de Gauss con pivotes diagonales. Sea A una matriz de tamaño $n \times n$ y sea B una matriz obtenida de la matriz A al aplicar operaciones elementales de la forma $R_q += \mu R_p$, con $q > p$. Demostrar que los menores principales líderes de la matriz B coinciden con los menores correspondientes de la matriz A, esto es, para cada $p \in \{1, \ldots, n\}$,

$$\det B(1:p, 1:p) = \det A(1:p, 1:p).$$

38. Teorema: criterio de existencia de una factorización LU. Sea A una matriz cuadrada invertible. Demuestre que A tiene una factorización LU si y sólo si A es estrictamente invertible.

39. Ejemplo de una matriz invertible que no tiene factorización LU. Construya una matriz de orden 3 que sea invertible, pero no tenga ninguna factorización LU.

40. Programación: factorización de Cholesky. Escriba una función que calcule el factor triangular inferior de Cholesky de una matriz dada (suponer que dicha factorización existe). Calcule el número de flops.

41. Teorema: criterio de existencia de una factorización de Cholesky. Sea A una matriz real cuadrada de orden n. Demuestre que las siguientes condiciones son equivalentes:

- A tiene una factorización de Cholesky $A = LL^\top$, donde L es una matriz triangular inferior invertible.
- A es simétrica y positiva definida, esto es, $A^\top = A$ y $x^\top Ax > 0$ para cada $x \in \mathbb{R}^n \setminus \{0\}$.

42. Pivoteo parcial (por columnas). En el algoritmo de factorización PLU, denotemos por U al factor U obtenido después de aplicar $p-1$ pasos del algoritmo. Explique los cálculos y operaciones que utiliza el pivoteo parcial por columnas en el inicio del p-ésimo paso del algoritmo, antes de aplicar operaciones elementales. Calcule el número de flops.

43. Pivoteo parcial escalado (por columnas). En el algoritmo de factorización PLU, denotemos por U al factor U obtenido después de aplicar $p-1$ pasos del algoritmo. Explique los cálculos y operaciones que utiliza el pivoteo parcial escalado por columnas en el inicio del p-ésimo paso del algoritmo, antes de aplicar operaciones elementales. Calcule el número de flops.

44. Programación: factorización PLU. Escriba una función que calcule una factorización PLU de la matriz dada $A \in \mathcal{M}_n(\mathbb{R})$, usando el pivoteo por columna. Regresar la permutación que corresponde a la matriz P y las matrices L y U. Calcule el número de flops.

45. Programación: solución de sistemas de ecuaciones lineales usando la factorización PLU. Escriba una función que resuelva sistemas de ecuaciones lineales de la forma $Ax = b$ usando una factorización PLU de la matriz dada A. Use funciones de los problemas anteriores. Calcule el número de flops.

46. Programación: cálculo del determinante de una matriz. Escriba una función que calcule el determinante de la matriz dada A. Use el pivoteo parcial por columna o el pivoteo por columna escalado. Sugerencia: simplifique la función que realiza la factorización PLU.