Prerrequisitos de la asignatura
Álgebra Lineal Numérica

El propósito de *Álgebra Lineal Numérica* es analizar algoritmos típicos de álgebra lineal, optimizando la rapidez y la precisión. Álgebra Lineal Numérica requiere ciertos conocimientos, habilidades y destrezas de programación, de álgebra lineal y de análisis. Este texto contiene varios ejercicios típicos que se recomienda resolver antes de inscribirse a esta asignatura. Por lo común, estos ejercicios se resuelven en varios cursos de nivel superior: programación, métodos numéricos, álgebra lineal y análisis (o cálculo avanzado).

Prerrequisitos de programación y de métodos numéricos

Haber escrito, depurado y ejecutado al menos 3 000 lineas de código en algunos lenguajes de programación, orientados a la programación de funciones (esto corresponde a uno o dos semestres verdaderos de programación). Inscribirse al curso de Álgebra Lineal Numérica sin esta experiencia mínima sería tan absurdo como entrar a un curso avanzado de Ecuaciones Diferenciales en posgrado sin haber estudiado Cálculo I y Cálculo II en licenciatura.

En particular, se requieren las siguientes experiencias elementales:

- programar funciones, cuyos argumentos y resultados son arreglos;
- programar funciones y llamarlas en otras funciones;
- programar funciones con varios ciclos `for`, encajados (anidados) uno dentro del otro;
- calcular el número de operaciones aritméticas en funciones del inciso anterior;
- programar funciones con ciclos de tipo `while`;
- inventar datos para hacer pruebas (en particular, generar arreglos de números pseudoaleatorios); detectar y corregir errores en programas.

Ejercicio 1 (multiplicación de una matriz por un vector)**.** Programar una función (darle el título `mulmatvec` o algún otro) que multiplique una matriz dada por un vector dado.

> Entrada: una matriz A, un vector b.
>
> Condición que satisface la entrada: el número de las columnas de A coincide con la longitud de b (suponer que esta condición se cumple y no verificarla).
>
> Salida: el vector Ab.

Sugerencia: usar dos ciclos `for` anidados. Por supuesto, si el lenguaje de programación ya tiene una función para multiplicar matrices por vectores, no se trata de utilizar la función dada; hay que programar su propia. Por supuesto, la función `mulmatvec` no debe contener operaciones de entrada y salida (como `printf` y `scanf` en C). La función `mulmatvec` se debe llamar en otra función que genere los datos A y b y muestre los resultados.

Ejercicio 2 (multiplicación de una matriz por un vector, el número de operaciones)**.** Calcular el número de operaciones de multiplicación que se realizan en el algoritmo anterior, si A es una matriz cuadrada de orden n.

Ejercicio 3 (generar una matriz con una estructura especial)**.** Escribir una función de un argumento entero positivo n que genere la matriz cuadrada $A = [A_{j,k}]_{j,k=1}^n$, donde

$$A_{j,k} = \begin{cases} 1, & \text{si } j = k; \\ \dfrac{1}{|j-k|}, & \text{si } j \neq k. \end{cases}$$

Hay dos maneras de resolver este ejercicio: 1) usar el operador condicional; 2) no usarlo.

Ejercicio 4 (generar la matriz de permutación asociada a la permutación dada)**.** Programar una función que construya y regrese la matriz de permutación P_φ asociada a la permutación dada φ. Se recomienda evitar el uso del operador condicional.

> Entrada: un arreglo φ.
>
> Condición que satisface la entrada: los elementos $\varphi(1), \ldots, \varphi(n)$ son los números $1, \ldots, n$ escritos sin repeticiones, pero posiblemente en otro orden.
>
> Salida: la matriz de permutación P_φ.

Por ejemplo, si φ es el arreglo $3, 2, 4, 1$, entonces la función debe regresar la matriz

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Ejercicio 5 (solución de sistemas de ecuaciones lineales con matrices triangulares superiores)**.** Programar el algoritmo de la sustitución hacia atrás para resolver sistemas de ecuaciones lineales de la forma $Ux = b$, donde U es una matriz triangular superior de orden n con elementos diagonales no nulos y b es un vector de longitud n.

> Entrada: una matriz U y un vector b.
>
> Condiciones que satisface la entrada: la matriz U es cuadrada, triangular superior y tiene elementos diagonales no nulos, y la longitud del vector b coincide con el orden de U.
>
> Salida: un vector x tal que $Ux = b$.

Calcular el número de operaciones de multiplicación y división que se realizan en el algoritmo, si el orden de la matriz U es n.

Ejercicio 6 (solución de sistemas tridiagonales de ecuaciones lineales, usando pivotes diagonales, sin construir la matriz completa)**.** 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

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

Ejercicio 7 (el método del punto fijo)**.** Escribir una función que realice el *método del punto fijo* llamado también el *método de iteración simple*. El algoritmo debe terminarse cuando la distancia entre dos aproximaciones sucesivas es menor que el número t dado o cuando el número de las iteraciones hechas es mayor que el número m dado.

> Entrada: una función f, un punto inicial a, el número máximo de pasos m y la "tolerancia" $t > 0$.
>
> Salida: un punto x que aproxime el punto fijo.
>
> Ejemplo para hacer pruebas: $f(x) = \frac{1}{2}\left(x + \frac{3}{x}\right)$, $a = 1$.

Ejercicio 8 (el método de bisección)**.** Escribir una función que realice el método de bisección. El algoritmo debe terminarse cuando la distancia entre los extremos del intervalo encontrado es menor que el número t dado o cuando el número de las iteraciones hechas es mayor que el número m dado.

> Entrada: una función f, dos números reales a y b, el número máximo de pasos m y la "tolerancia" t.
>
> Condiciones que satisface la entrada: f tiene un argumento real y regresa un valor real, $a < b$, $f(a)f(b) < 0$, $m > 0$, $t > 0$.
>
> Salida: una aproximación de un cero de f en $[a, b]$.

Ejercicio 9 (el método de la regla falsa)**.** Escribir una función que realice el método de la regla falsa. Los argumentos y el resultado de la función son los mismos que en el ejercicio anterior.

Ejercicio 10 (violación de la ley asociativa de adición en la aritmética con redondeo)**.** Denotemos por $\oplus$ y $\otimes$ a las operaciones con números reales en el formato `float double` (formato con punto flotante de precisión doble), con redondeo al más cercano. Es la aritmética que se usa comunmente en los lenguajes C, Octave, Python+numpy, y otros. Encontrar tres números a, b, c tales que

$$a \oplus (b \oplus c) \neq (a \oplus b) \oplus c.$$

Hacer la comprobación en algún lenguaje de programación. Sugerencia: recordar el concepto del *épsilon de la máquina*.

Prerrequisitos de álgebra lineal

- Demostrar propiedades de operaciones con matrices, por ejemplo,

$$A(B+C) = AB + AC, \qquad (AB)^\top = B^\top A^\top, \qquad AI = A.$$

- Entender la estructura del producto de matrices. Por ejemplo, saber escribir cada columna del producto AB como una combinación lineal de las columnas de A.
- Aplicar criterios de invertibilidad de una matriz cuadrada.
- Aplicar el teorema sobre el rango y la nulidad (dimensión de la imagen y del núcleo).
- Practicar los algoritmos principales para espacios euclidianos o unitarios: proyección de un vector sobre el subespacio generado por una lista de vectores ortogonales, ortogonalización de Gram–Schmidt, cálculo de la matriz de Gram.
- Demostrar los teoremas principales sobre el concepto de ortogonalidad: la desigualdad de Schwarz, el teorema sobre la proyección de un vector sobre el subespacio generado por una lista de vectores ortogonales, corolario sobre la independencia lineal de una lista ortogonal de vectores no nulos, el teorema sobre la conservación de subespacios generados en la ortogonalización de Gram–Schmidt.
- Trabajar con varias descripciones equivalentes de matrices ortogonales (en el caso real) o unitarias (en el caso complejo).
- Demostrar propiedades espectrales de matrices autoadjuntas (hermitianas).

Ejercicio 11. Sea A una matriz real cuadrada de orden n. Escribir al menos 15 condiciones equivalentes a la condición que A es invertible. En total se pueden encontrar más de 20 condiciones equivalentes a la invertibilidad de A.

Ejercicio 12. Sean A y B dos matrices reales cuadradas triangulares superiores del mismo tamaño. Demostrar que su producto AB también es una matriz triangular superior y deducir fórmulas para los elementos diagonales de la matriz AB.

Ejercicio 13. Sea A una matriz cuadrada de orden 4 con elementos reales. Supongamos que A es invertible por la izquierda, es decir, existe una matriz B tal que $BA = I_4$. Demostrar que A es invertible por la derecha.

Ejercicio 14. En el espacio $\mathbb{R}^2$ con el producto interno canónico (el producto punto) consideremos los vectores

$$v = \begin{bmatrix} 3 \\ 2 \end{bmatrix}, \qquad a = \begin{bmatrix} 2 \\ -1 \end{bmatrix}.$$

Denotemos por S al subespacio generado por el vector a. Construir vectores $u, w \in \mathbb{R}^2$ tales que $v = u + w$, $u \in S$ y $w \perp S$.

Ejercicio 15. Sea A una matriz real cuadrada de orden n cuyan columnas forman una base ortonormal de $\mathbb{R}^n$. Demostrar que los renglones de A también forman una base ortonormal de $\mathbb{R}^n$.

Prerrequisitos de análisis

- Verificar que una función es contractiva.
- Demostrar el teorema sobre el punto fijo de una función contractiva.
- Saber la definición de las normas $\|\cdot\|_p$ y $\|\cdot\|_\infty$ en el espacio $\mathbb{R}^n$.
- Demostrar la equivalencia de las normas $\|\cdot\|_1$, $\|\cdot\|_2$ y $\|\cdot\|_\infty$.
- Trabajar con la norma de un operador lineal.

Ejercicio 16. Sea $f\colon [a,b] \to [a,b]$ una función continua y derivable, con

$$L \coloneqq \sup_{x\in(a,b)} |f'(x)| < 1.$$

¿Cuántos pasos del método del punto fijo serán suficientes para aproximar el punto fijo de f con la precisión $\le \varepsilon$?

Ejercicio 17. Escribir cuatro definiciones equivalentes de la norma de un operador lineal.

Ejercicio 18. Sea A una matriz real cuadrada de orden n. Denotemos por T al operador lineal asociado a esta matriz:

$$T(x) \coloneqq Ax \qquad (x \in \mathbb{R}^n).$$

Demostrar que la norma del operador lineal T es finita.

Ejercicio 19. Sea A una matriz real cuadrada de orden n. Calcular el gradiente de la función

$$q(x) \coloneqq x^\top A\, x.$$

Sugerencia: si es difícil resolver el problema en general, entonces se recomienda empezar con el caso particular $n = 2$ o $n = 3$.

Ejercicio 20. Sean a y b dos vectores del espacio $\mathbb{R}^n$. Se sabe que

$$\begin{aligned} &\|a\|_1 = 8, && \|a\|_2 = 5, && \|a\|_\infty = 3,\\ &\|b\|_1 = 11, && \|b\|_2 = 10, && \|b\|_\infty = 9. \end{aligned}$$

Escribir varias cotas superiores para $|a^\top b|$ y elegir la cota superior más precisa. Notamos que $a^\top b$ es lo mismo que el producto punto de los vectores a y b.

Ejercicio 21. Sea $A \in \mathcal{M}_n(\mathbb{R})$. Calcular $f'(0)$, donde la función $f\colon \mathbb{R} \to \mathbb{R}$ está definida mediante la regla

$$f(x) \coloneqq \det(I_n + xA).$$