Splines (funciones polinomiales por trozos)

Problemas para examen

Interpolación lineal y cúbica

1. Fórmulas para la interpolación lineal. Dados $t_1, \ldots, t_n, x_1, \ldots, x_n \in \mathbb{R}$ tales que

$$t_1 < \ldots < t_n < t_{n+1} \coloneqq +\infty,$$

encontrar una función $S\colon [t_1, t_n] \to \mathbb{R}$ con las siguientes propiedades:

- Para cada $j \in \{1, \ldots, n\}$, la restricción de S al intervalo $[t_j, t_{j+1})$ es un polinomio de grado ≤ 1:
$$S(u) = a_j + b_j(u - t_j). \tag{1}$$
- S es continua en $[t_1, t_n]$.
- $S(t_j) = x_j$ para cada j en $\{1, \ldots, n\}$.

En otras palabras, hay que encontrar los coeficientes $a_1, \ldots, a_n$ y $b_1, \ldots, b_n$ tales que S sea continua y tome los valores dados en los nodos $t_1, \ldots, t_n$.

2. Fórmulas para la interpolación cúbica. Dados $t_1, \ldots, t_n, x_1, \ldots, x_n \in \mathbb{R}$ tales que

$$t_1 < \ldots < t_n < t_{n+1} \coloneqq +\infty,$$

encontrar una función $S\colon [t_1, t_n] \to \mathbb{R}$ con las siguientes propiedades:

- Para cada $j \in \{1, \ldots, n\}$, la restricción de S al intervalo $[t_j, t_{j+1})$ es un polinomio de grado ≤ 3:
$$S(u) = a_j + b_j(u - t_j) + c_j(u - t_j)^2 + d_j(u - t_j)^3.$$
- S es de clase C^2 en el segmento $[t_1, t_n]$.
- $S(t_j) = x_j$ para cada j en $\{1, \ldots, n\}$.
- $S''(t_1) = S''(t_n) = 0$.

En otras palabras, hay que encontrar los coeficientes a_j, b_j, c_j y d_j tales que S tengas las propiedades requeridas. Se recomienda expresar todos los coeficientes a_j, b_j y d_j en términos de $h_j = x_{j+1} - x_j$, y_j y c_j, y formar un sistema de ecuaciones lineales para las incógnitas c_j.

3. Programación: el número de elementos de un arreglo ordenado que son menores o iguales que un número dado. Escribir una función `countleq` que determine y devuelva el número de las componentes de un arreglo real ordenado que son menores o iguales que un número real.

Entrada: Un vector real a y un número $v \in \mathbb{R}$. Denotamos la longitud de a por n. Se supone que

$$a_1 \le \cdots \le a_n.$$

Salida: $\#\{j \in \{1, \dots, n\}\colon\ a_j \le v\}$.

Ejemplo: `countleq([-2; -2; 3; 4; 4; 7; 8], 4)` debe devolver `5`.

Ejemplo: `countleq([-3; -1; 3; 4; 5; 5; 11], 8)` debe devolver `6`.

Ejemplo: `countleq([-3; -1; 3; 4; 5; 5; 11], -4)` debe devolver `0`.

Ejemplo: `countleq([-3; -1; 3; 4; 5; 5; 11], 15)` debe devolver `7`.

4. Programación: valores de un spline lineal en puntos dados. Escribir una función que calcule y devuelva los valores de un spline lineal en puntos dados. El spline está dado por los nodos y por los coeficientes a_j y b_j de la fórmula (1).

Entrada: una matriz $S \in \mathbb{R}^{n\times 3}$, un vector $u \in \mathbb{R}^m$. Las columnas de la matriz S hacen papel de los vectores t, a, b de la fórmula (1).

Salida: el arreglo de los valores $S(u_1), \dots, S(u_m)$.

Sugerencia: para cada j encontrar k tal que $u_j \in [v_k, v_{k+1})$. Aplicar la función `countleq`.

5. Programación: coeficientes de un spline lineal que toma los valores dados en los puntos dados. Escribir una función que calcule y devuelva la solución del Problema 1.

Entrada: $t, x \in \mathbb{R}^n$, tales que $t_1 < \ldots < t_n$.

Salida: Matriz $S \in \mathbb{R}^{n\times 3}$ cuyas columnas son los vectores t, a, b.

6. Programación: solución de un sistema tridiagonal de ecuaciones lineales. Escribir una función que resuelva sistemas de ecuaciones lineales con matrices tridiagonales. Por ejemplo, para $n = 4$ el sistema tiene la siguiente forma:

$$\begin{bmatrix} a_1 & b_1 & 0 & 0 & 0 \\ c_1 & a_2 & b_2 & 0 & 0 \\ 0 & c_2 & a_3 & b_3 & 0 \\ 0 & 0 & c_3 & a_4 & b_4 \\ 0 & 0 & 0 & c_4 & a_5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \end{bmatrix}. \tag{2}$$

Se supone que la matriz es estrictamente invertible, esto es, todos los menores principales líderes son diferentes de cero. Esto implica que el sistema se puede resolver con la eliminación de Gauss sin pivoteo.

ENTRADA: Los vectores a, b, c, r.

SALIDA: El vector x.

En la solución de este problema no se debe construir la matriz del sistema en la forma explícita. Hay que trabajar con los vectores a, b, d que guardan la parte interesante de la matriz.

7. Programación: valores de un spline cúbico en puntos dados. Dados los nodos (abscisas) y los coeficientes de un spline cúbico, calcular sus valores en los puntos dados.

ENTRADA: Una matriz $S \in \mathbb{R}^{n \times 5}$ que contiene las columnas t, a, b, c, d del problema 2, un vector $u \in \mathbb{R}^m$.

SALIDA: El vector de los valores del spline cúbico en los puntos $u_1, \ldots, u_m$.

8. Programación: coeficientes de un spline cúbico que toma los valores dados en los puntos dados. Escribir una función que resuelva el Problema 2.

ENTRADA: Vectores $t, x \in \mathbb{R}^n$ tales que $t_1 < \ldots < t_n$.

SALIDA: La matriz S que consiste de las columnas t, a, b, c, d.

Splines básicos

9. Definición recursiva de los splines básicos. Escribir la definición de las funciones $B_{0,k}$ y la definición recursiva de las funciones $B_{p,k}$, $p \geq 1$.

10. Fórmulas explícitas para los splines básicos lineales y cuadráticos. Usando las fórmulas recursivas deducir fórmulas directas para los splines básicos $B_{1,k}$ y $B_{2,k}$.

11. Valores de los splines básicos cúbicos en los nodos. Calcular los valores de $B_{p,k}$ en $t_{k+1}, t_{k+2}, t_{k+3}$.

12. Sistema de ecuaciones lineales para hacer la interpolación en términos de los splines básicos cúbicos. Dados los nodos (abscisas) $T_1, \ldots, T_n$ y los valores (ordenadas) $x_1, \ldots, x_n$, explicar cómo construir los nodos $t_1, \ldots, t_{n+4}$ y los coeficientes $\alpha_1, \ldots, \alpha_n$ tales que

$$\sum_{k=1}^{n} \alpha_k B_{p,k}(T_j) = x_j.$$

13. El soporte de un spline básico. Demostrar por inducción que $B_{p,k}$ se anula en el complemento de cierto intervalo. Demostrar por inducción que $B_{p,k}$ toma valores estrictamente positivos en cierto intervalo.

14. Fórmula recursiva para una combinación lineal de splines básicos. Expresar

$$\sum_{k=1}^{n-p-1} \alpha_k B_{p,k}(u)$$

en términos de las funciones $B_{p-1,k}$.

15. Partición de la unidad. Demostrar que la función

$$\sum_{k=1}^{n-1} B_{p,k}$$

es igual a 1 en cierto intervalo.

16. Identidad de Marsden. Expresar la función potencial

$$f(u) \coloneqq (u - \tau)^p$$

como una combinación lineal de los splines básicos $B_{p,1}, \ldots, B_{p,n-p-1}$.

17. Expresión de polinomios en términos de splines básicos. Demostrar que cualquier polinomio de grado $\leq p$ se expresa como una combinación lineal de splines básicos, en cierto intervalo.

18. Independencia lineal de splines básicos considerados como funciones en todo el dominio. Demostrar que las funciones $B_{p,1}, \ldots, B_{p,n-p-1}$ son linealmente independientes.

19. Funcionales duales de los splines básicos. Explicar la definición formal de los funcionales duales de los splines básicos y demostrar la fórmula explícita para calcular estos funcionales.

20. Derivada de un spline básico. Enunciar y demostrar la fórmula para la derivada de $B_{p,k}$.

21. Derivada de una combinación lineal de splines básicos. Deducir una fórmula para la derivada de la función

$$\sum_{k=1}^{n-p-1} \alpha_k B_{p,k}.$$

22. Programación: cálculo de los splines básicos de grado 0. Escribir una función que calcule los valores de las funciones $B_{0,1}, \ldots, B_{0,n-1}$ en puntos dados $u_1, \ldots, u_m$.

Entrada: $t \in \mathbb{R}^n$ con $t_1 < \ldots < t_n$, $u \in \mathbb{R}^m$.

Salida: La matriz $\left[B_{0,k}(u_j)\right]_{j,k=1}^{m,n-1}$.

23. Programación: cálculo de los cocientes. Escribir una función que calcule los valores de las funciones $q_{p,1}, \ldots, q_{p,n-p}$ en puntos dados $u_1, \ldots, u_m$. Recordamos que

$$q_{p,k}(u) = \frac{u - t_k}{t_{k+p} - t_k}.$$

24. Programación: cálculo de los splines básicos. Escribir una función que calcule los valores de las funciones $B_{p,1}, \ldots, B_{p,n-p-1}$ en puntos dados $u_1, \ldots, u_m$.

25. Programación: interpolación cúbica usando splines básicos. Escribir una función que resuelva el Problema 12.

Entrada: $T, x \in \mathbb{R}^n$ tales que $T_1 < \ldots < T_n$.

Salida: $\alpha \in \mathbb{R}^n$.