

Interpolación con splines cuadráticos

Métodos Numéricos

Rubén
Morales

Pablo
Baeyens

Francisco
Morales

Pablo
Medina

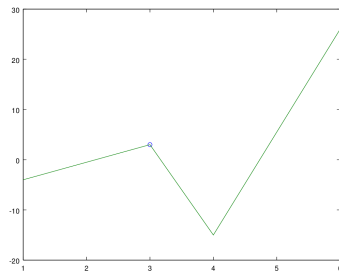
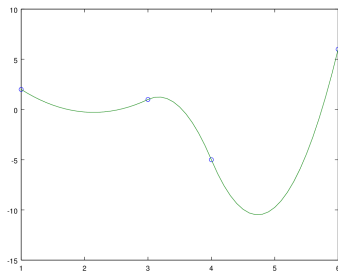
Miguel
Anguita

10 de Junio de 2015

Splines cuadráticos

Definición

Un **spline cuadrático** es una función $s \in S_2^1(P)$, para cierta partición P sobre un intervalo.



Dimensión del espacio

Proposición

Sea $[a, b]$ intervalo, $P = \{x_i\}_{i=0 \dots n} \in \mathcal{P}([a, b])$, entonces $\dim(S_2(P)) = n + 2$.

Dimensión del espacio

Proposición

Sea $[a, b]$ intervalo, $P = \{x_i\}_{i=0\dots n} \in \mathcal{P}([a, b])$, entonces $\dim(S_2(P)) = n + 2$.

Dimensión del espacio

Proposición

Sea $[a, b]$ intervalo, $P = \{x_i\}_{i=0\dots n} \in \mathcal{P}([a, b])$, entonces $\dim(S_2(P)) = n + 2$.

Demostración.

Podemos deducirlo a partir de la fórmula general:

$$\dim(S_k^r(P)) = (k - r)n + r + 1$$

Así:

$$\dim(S_2^1(P)) = (2 - 1)n + 1 + 1 = n + 2$$



Dimensión del espacio

Proposición

Sea $[a, b]$ intervalo, $P = \{x_i\}_{i=0\dots n} \in \mathcal{P}([a, b])$, entonces $\dim(S_2(P)) = n + 2$.

Demostración.

Podemos deducirlo a partir de la fórmula general:

$$\dim(S_k^r(P)) = (k - r)n + r + 1$$

Así:

$$\dim(S_2^1(P)) = (2 - 1)n + 1 + 1 = n + 2$$



Dimensión del espacio

Proposición

Sea $[a, b]$ intervalo, $P = \{x_i\}_{i=0\dots n} \in \mathcal{P}([a, b])$, entonces $\dim(S_2(P)) = n + 2$.

Demostración.

Podemos deducirlo a partir de la fórmula general:

$$\dim(S_k^r(P)) = (k - r)n + r + 1$$

Así:

$$\dim(S_2^1(P)) = (2 - 1)n + 1 + 1 = n + 2$$



Base del espacio

Base

Conocida la dimensión podemos establecer una base:

$$\{1, x, x^2, (x - x_1)_+^2, \dots, (x - x_{n-1})_+^2\}$$

Problema de interpolación

Problema

Hallar $s \in S_2(P)$ tal que para todo $0 \leq i \leq n$, $s(x_i) = y_i$ y para un cierto k , $s'(x_k) = d_k$.

Los datos del problema son:

$$\begin{array}{c|ccccc} x_i & x_0 & \cdots & x_k & \cdots & x_n \\ y_i & y_0 & \cdots & y_k & \cdots & y_n \\ d_i & & & d_k & & \end{array}$$

Método local

Dada d_k , calculamos mediante diferencias divididas s_k y s_{k+1} .

s_k : d_k queda a la **derecha**

$$\begin{array}{ccc}
 x_{k-1} & y_{k-1} & \\
 x_k & y_k & p_k \\
 x_k & y_k & d_k \quad \frac{d_k - p_k}{h_k}
 \end{array}$$

$$s_k(x) = y_{k-1} + p_k(x - x_{k-1}) + \frac{d_k - p_k}{h_k}(x - x_{k-1})(x - x_k)$$

Método local

Dada d_k , calculamos mediante diferencias divididas s_k y s_{k+1} .

s_{k+1} : d_k queda a la **izquierda**

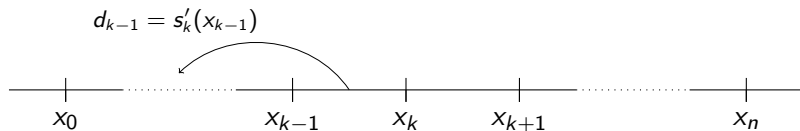
$$\begin{array}{cccc}
 x_k & y_k & & \\
 x_k & y_k & d_k & \\
 x_{k+1} & y_{k+1} & p_{k+1} & \frac{p_{k+1}-d_k}{h_k}
 \end{array}$$

$$s_{k+1}(x) = y_k + d_k(x - x_k) + \frac{p_{k+1} - d_k}{h_{k+1}}(x - x_k)(x - x_k)$$

Método local

Repetimos este proceso actualizando la derivada:

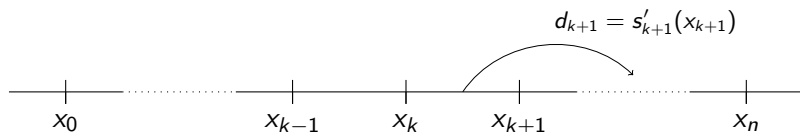
- Hacia la **izquierda**, aplicando la primera fórmula.
- Hacia la derecha, aplicando la segunda fórmula.



Método local

Repetimos este proceso actualizando la derivada:

- Hacia la izquierda, aplicando la primera fórmula.
- Hacia la **derecha**, aplicando la segunda fórmula.



Ejemplo: Método local

Método local

Hallar $s \in S_2(-1, 1, 3, 6, 7)$ que interpole los datos de la tabla.

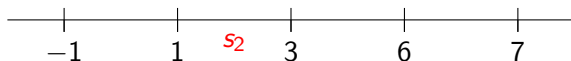
x_i	-1	1	3	6	7
y_i	1	4	8	2	9
d_i			5		

Ejemplo: Método local

Primero avanzamos hacia la izquierda:

$$\begin{array}{rcl} 1 & 4 & \\ 3 & 8 & 2 \\ 3 & 8 & \mathbf{5} \end{array} \quad \frac{3}{2}$$

$$s_2(x) = 4 + 2(x - 1) + \frac{3}{2}(x - 1)(x - 3)$$

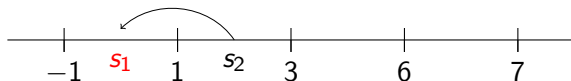


Ejemplo: Método local

Primero avanzamos hacia la izquierda:

$$\begin{array}{rcl}
 -1 & 1 & \\
 1 & 4 & \frac{3}{2} \\
 1 & 4 & -1 \quad \frac{-5}{4}
 \end{array}
 \qquad
 s_1(x) = 1 + \frac{3}{2}(x+1) - \frac{5}{4}(x+1)(x-1)$$

$$d_1 = s_2'(x_1) = -1$$



Ejemplo: Método local

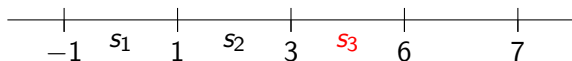
Y después hacia la derecha:

3 8

3 8 **5**

6 2 -2 $-\frac{7}{3}$

$$s_3(x) = 8 + 5(x - 3) - \frac{7}{3}(x - 3)(x - 3)$$



Ejemplo: Método local

Y después hacia la derecha:

6 2

6 2 **-9**

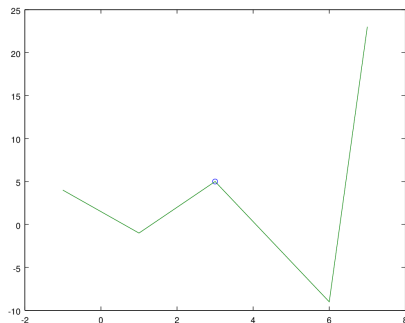
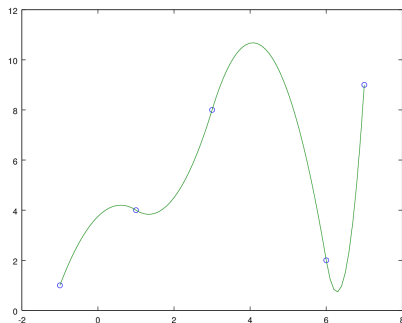
7 9 7 16

$$s_4(x) = 2 - 9(x - 6) + 16(x - 6)(x - 6)$$



Ejemplo: Método local

$$s(x) = \begin{cases} 1 + \frac{3}{2}(x+1) - \frac{5}{4}(x+1)(x-1) & x \in [-1, 1] \\ 4 + 2(x-1) + \frac{3}{2}(x-1)(x-3) & x \in (1, 3] \\ 8 + 5(x-3) - \frac{7}{3}(x-3)(x-3) & x \in (3, 6] \\ 2 - 9(x-6) + 16(x-6)(x-6) & x \in (6, 7] \end{cases}$$



Método global

Resolvemos el sistema, añadiendo la condición para la derivada:

$$\begin{pmatrix}
 1 & x_0 & x_0^2 & 0 & \cdots & 0 \\
 1 & x_1 & x_1^2 & (x_1 - x_1)_+^2 & \cdots & 0 \\
 \vdots & & \vdots & \vdots & \cdots & \vdots \\
 \vdots & & \vdots & \vdots & \cdots & \vdots \\
 1 & x_n & x_n^2 & (x_n - x_1)_+^2 & \cdots & (x_n - x_{n-1})_+^2 \\
 0 & 1 & 2x_k & 2(x_k - x_1)_+ & \cdots & 2(x_k - x_{n-1})_+
 \end{pmatrix}
 \begin{pmatrix}
 a \\
 b \\
 c \\
 \alpha_1 \\
 \vdots \\
 \alpha_{n-1}
 \end{pmatrix}
 =
 \begin{pmatrix}
 y_0 \\
 y_1 \\
 \vdots \\
 \vdots \\
 y_n \\
 d_k
 \end{pmatrix}$$

Método global

Resolvemos el sistema, añadiendo la condición para la derivada:

$$\begin{pmatrix}
 1 & x_0 & x_0^2 & 0 & \cdots & 0 \\
 1 & x_1 & x_1^2 & (x_1 - x_1)_+^2 & \cdots & 0 \\
 \vdots & & \vdots & \vdots & \cdots & \vdots \\
 \vdots & & \vdots & \vdots & \cdots & \vdots \\
 1 & x_n & x_n^2 & (x_n - x_1)_+^2 & \cdots & (x_n - x_{n-1})_+^2 \\
 0 & 1 & 2x_k & 2(x_k - x_1)_+ & \cdots & 2(x_k - x_{n-1})_+
 \end{pmatrix}
 \begin{pmatrix}
 a \\
 b \\
 c \\
 \alpha_1 \\
 \vdots \\
 \alpha_{n-1}
 \end{pmatrix}
 =
 \begin{pmatrix}
 y_0 \\
 y_1 \\
 \vdots \\
 \vdots \\
 y_n \\
 d_k
 \end{pmatrix}$$

Método global

Resolvemos el sistema, añadiendo la condición para la derivada:

$$\begin{pmatrix} 1 & x_0 & x_0^2 & 0 & \cdots & 0 \\ 1 & x_1 & x_1^2 & (x_1 - x_1)_+^2 & \cdots & 0 \\ \vdots & & \vdots & \vdots & \cdots & \vdots \\ \vdots & & \vdots & \vdots & \cdots & \vdots \\ 1 & x_n & x_n^2 & (x_n - x_1)_+^2 & \cdots & (x_n - x_{n-1})_+^2 \\ 0 & 1 & 2x_k & 2(x_k - x_1)_+ & \cdots & 2(x_k - x_{n-1})_+ \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ \alpha_1 \\ \vdots \\ \alpha_{n-1} \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ \vdots \\ y_n \\ d_k \end{pmatrix}$$

$$s(x) = a + bx + cx^2 + \alpha_1(x - x_1)_+^2 + \cdots + \alpha_{n-1}(x - x_{n-1})_+^2$$

Ejemplo: Método global

Método global

Hallar $s \in S_2(2, 4, 5, 8)$ que interpole los datos de la tabla.

x_i	2	4	5	8
y_i	7	3	5	5
d_i		4		

Ejemplo: Método global

Debemos hallar $a, b, c, \alpha_1, \alpha_2 \in \mathbb{R}$ tales que, para $x \in [2, 8]$:

$$s(x) = a + bx + cx^2 + \alpha_1(x - 4)_+^2 + \alpha_2(x - 5)_+^2$$

Ejemplo: Método global

Debemos hallar $a, b, c, \alpha_1, \alpha_2 \in \mathbb{R}$ tales que, para $x \in [2, 8]$:

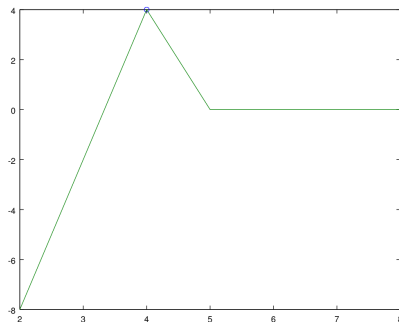
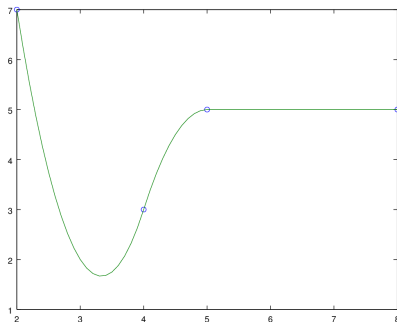
$$s(x) = a + bx + cx^2 + \alpha_1(x-4)_+^2 + \alpha_2(x-5)_+^2$$

$$\begin{pmatrix} 1 & 2 & 4 & 0 & 0 \\ 1 & 4 & 16 & 0 & 0 \\ 1 & 5 & 25 & 1 & 0 \\ 1 & 8 & 64 & 16 & 9 \\ 0 & 1 & 8 & 0 & 0 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ \alpha_1 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} 7 \\ 3 \\ 5 \\ 5 \\ 4 \end{pmatrix}$$

$$a = 35 \quad b = -20 \quad c = 3 \quad \alpha_1 = -5 \quad \alpha_2 = 2$$

Ejemplo: Método global

$$s(x) = \begin{cases} 3x^2 - 20x + 35 & x \in [2, 4] \\ -2x^2 + 20x - 45 & x \in (4, 5] \\ 5 & x \in (5, 8] \end{cases}$$



SplineCuad

Implementamos la función con el **método global**.

- 1 Calculamos la matriz de coeficientes:

```
n = length(x) - 1;  
  
A(:,1) = [ones(n+1,1); 0];  
A(:,2) = [x'           ; 1];  
A(:,3) = [x'.^2         ; 2.*x(k+1)];  
  
for j = 4 : n + 2  
    t = @(s) (s > x(j-2)) .* (s - x(j-2));  
    A(:, j) = [t(x') .^2; 2.*t(x(k+1))];  
end
```

SplineCuad

Implementamos la función con el **método global**.

② Y resolvemos el sistema:

```
sol = A \ [y' ; d_k];

for k = 1:n
    p = sol(3:-1:1)';

    for l = 2:k
        p += sol(l+2).*[1, -2.*x(l), x(l).^2];
    end

    B(k, :) = polyaffine(p,[-x(k) 1]);
end

s = mkpp(x,B);
```

SplineCuadLocal

- 1 Recorremos todos los nodos de $k+1$ en adelante:

```
for i = (k+1):(length(x)-1)
    p = (y(i+1)-y(i))/(x(i+1)-x(i));
    q = (p-d)/(x(i+1)-x(i));
    v = [x(i) x(i)];
    s(i,:) = [0, 0, y(i)]+[0, d, -d*x(i)]+q*poly(v);
    d = polyval(polyder(s(i,:)),x(i+1));
end
d = d_k;
```

SplineCuadLocal

- ② Recorremos todos los nodos desde k hasta el 1:

```
for j = k:-1:1
    p = (y(j+1)-y(j))/(x(j+1)-x(j))
    q = (d-p)/(x(j+1)-x(j))
    v = [x(j) x(j+1)]
    s(j,:) = [0 0 y(j)]+[0 p -p*x(j)]+q*poly(v);
    d = polyval(polyder(s(j,:)), x(j))
end
```

SplineCuadLocal

- ③ Y cambiamos de base usando polyaffine:

```
for i = 1:length(s)
    s(i,:) = polyaffine(s(i,:), [-x(i), 1]);
end
```