

Ampliación de interpolación con Splines

Miguel Anguita Ruiz Pablo Baeyens Fernández Pablo David Medina Sánchez
Ruben Morales Pérez Francisco Javier Morales Piqueras

Índice

1. Splines cuadráticos	2
1.1. Introducción a los splines	2
1.2. Descripción del espacio de splines cuadráticos	2
1.3. Interpolación con splines cuadráticos	2
1.4. Error en los splines cuadráticos	2
1.5. Ejemplos	3
2. Splines cúbicos	4
2.1. Construcción a partir de los valores de s'' en los nodos $\{x_i\}$	4
2.2. Propiedades de minimización	4
2.2.1. Cota de error en los splines cúbicos	5
2.3. Ejemplos	6
3. Implementación en ordenador: Octave	7
3.1. Spline Lineal	7
3.2. Splines cuadráticos	7
A. Definiciones y notación	8

1. Splines cuadráticos

1.1. Introducción a los splines

Definición. Sea $[a, b]$ un intervalo, $P = \{x_i\}_{i=0 \dots n} \in \mathcal{P}([a, b])$, $k, r \in \mathbb{N}$, $r < k$. Se dice que $s : [a, b] \rightarrow \mathbb{R}$ es un spline si $s \in C^r([a, b])$ y para todo $1 \leq i \leq n$, $s|_{[x_{i-1}, x_i]} \in \mathbb{P}_k$. $S_k^r(P)$ es el espacio de dichas funciones.

1.2. Descripción del espacio de splines cuadráticos

Partimos de $[a, b]$ un intervalo y $P \in \mathcal{P}([a, b])$. En esta primera sección nos centramos en los splines cuadráticos: los pertenecientes a $S_2^1(P)$.

Sus trozos son polinomios de grado menor o igual que 2 de la forma $ax^2 + bx + c$. Además son funciones de clase 1 (derivables en $[a, b]$ con derivada continua), lo que proporciona condiciones interesantes para resolver problemas de interpolantes.

Veamos algunas propiedades:

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. Sea $s \in S_2(P)$.

- Para cada intervalo $[x_{i-1}, x_i]$ $s|_{[x_{i-1}, x_i]}(x) = ax^2 + bx + c$ para ciertos $a, b, c \in \mathbb{R}$. Por lo tanto cada trozo está determinado por 3 parámetros. Con n trozos tenemos $3n$ parámetros en total.
- Si imponemos la continuidad y derivabilidad en los extremos tenemos que

$$s_i(x_i) = s_{i+1}(x_i) \quad s'_i(x_i) = s'_{i+1}(x_i)$$

para todo $i = 1 \dots n - 1$. De cada condición se obtienen $n - 1$ ecuaciones, por lo tanto obtendremos: $n - 1 + n - 1 = 2n - 2$ ecuaciones linealmente independientes.

Por lo tanto, $\dim(S_2(P)) = 3n - (2n - 2) = n + 2$. □

Con el conocimiento de la dimensión del espacio podemos describir una base del espacio de splines cuadráticos con el uso de potencias truncadas. Una **base del espacio** es: $\{1, x, x^2, (x - x_1)_+^2, \dots, (x - x_{n-1})_+^2\}$.

1.3. Interpolación con splines cuadráticos

1.4. Error en los splines cuadráticos

Teorema. Sean $f \in C^2([a, b])$, $\{x_i\}_{i=0 \dots n} \in \mathcal{P}([a, b])$, $s \in S_2^1(\{x_i\}_{i=0 \dots n})$ spline para f , $h = \max\{x_i - x_{i-1}\}_{i=1 \dots n}$, $E = f - s$. Además, sea $M > 0$ tal que:

$$M \geq \sup\{|f''(x) - f''(y)| : |x - y| \leq h, x, y \in [a, b]\}$$

Entonces, se verifica, para todo $x \in [a, b]$:

$$E(x) \leq \frac{h^2 M}{2} \quad (1)$$

La demostración, así como cotas para las derivadas y cotas más precisas en función de la localización de x puede encontrarse en *Quadratic Interpolatory Splines*, W. Kammerer, G. Reddien y R.S. Varga, (1973).

1.5. Ejemplos

Problema. Dados los datos de la tabla, halla mediante el método global el spline cuadrático que interpole los nodos y cuya derivada en x_1 sea 4.

$$\begin{array}{c|cccc} x_i & 2 & 4 & 5 & 8 \\ y_i & 7 & 3 & 5 & 5 \\ d_i & & 4 & & \end{array}$$

Solución. Debemos hallar $s \in S_2(P)$ con $a, b, c, \alpha, \beta \in \mathbb{R}$ tales que, para $x \in [2, 8]$:

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

Planteamos el sistema de ecuaciones $GX = b$:

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

Resolviendo el sistema, obtenemos la solución $a = 35, b = -20, c = 3, \alpha = -5, \beta = 2$. Por tanto, para $x \in [2, 8]$:

$$s(x) = 35 - 20x + 3x^2 - 5(x-4)_+^2 + 2(x-5)_+^2$$

Es decir:

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

2. Splines cúbicos

Uno de los problemas de la interpolación polinomial es que, al ir aumentando el número de nodos el grado del polinomio necesario para interpolarlos aumenta. Esto conlleva fluctuaciones en los extremos de la interpolación.

Si dividimos el intervalo en una partición podemos interpolar utilizando un polinomio en cada intervalo, es decir, utilizando **splines cúbicos**. Como veremos después este método minimiza la cota de error.

Dentro de los cúbicos encontramos los de clase 1 y 2, denotados por:

1. Los splines cúbicos de clase 1 son continuos y derivables con derivada continua. Forman un espacio vectorial de dimensión $2(n+1)$, cuya base es: . Estos splines no aseguran derivabilidad en los extremos. En un contexto geométrico esto significa que la función no es *suave* en los puntos de unión. Generalmente las condiciones físicas necesitan esa suavidad, y es aquí donde intervienen los splines cúbicos de clase 2.
2. Los splines cúbicos de clase 2 son continuos y 2 veces derivables. Como sabemos que la dimensión de un spline la dimensión de este espacio es $(3-2)n+2+1=n+3$.

Como tenemos $n+1$ variables, tenemos 2 libertades en la resolución.

2.1. Construcción a partir de los valores de s'' en los nodos $\{x_i\}$

2.2. Propiedades de minimización

Comenzamos planteando un problema de minimización sobre el espacio euclídeo $(C^2([a, b]), < \cdot, \cdot >)$, con la métrica y norma definida de la forma usual:

$$< f, g > = \int_a^b fg, \quad ||f|| = \sqrt{\int_a^b f^2}$$

Planteamos el problema:

Problema. Sea $f \in C^2([a, b])$, $P \in \mathcal{P}([a, b])$. Sea $H \subset C^2([a, b])$ definido por:

$$H = \{g \in C^2([a, b]) : \forall p \in P \ g(p) = f(p) \text{ y } g'(a) = f'(a), \ g'(b) = f'(b)\}$$

Hallar $u \in H$ tal que $||u''||$ sea mínima.

Para resolver el problema, demostramos el siguiente teorema:

Teorema (Minimización). Sea $f \in C^2([a, b])$, $P \in \mathcal{P}([a, b])$, s spline sujeto para f . Se verifica:

$$\forall u \in H : ||s''|| \leq ||u''||$$

Demostración. Sea $u \in H$, $e = u - s$. Tenemos:

$$\|u''\|^2 = \|e'' + s''\|^2 = \|e''\|^2 + \|s''\|^2 + 2 \langle e'', s'' \rangle$$

Dividimos $\langle e'', s'' \rangle$ en intervalos:

$$\langle e'', s'' \rangle = \int_a^b e'' s'' = \sum_1^{n-1} \int_{x_i}^{x_{i+1}} e'' s''$$

En cada intervalo, integramos por partes:

$$\sum_1^{n-1} \int_{x_i}^{x_{i+1}} e'' s'' = \sum_1^{n-1} e'(x) s''(x) \Big|_{x_i}^{x_{i+1}} - \sum_1^{n-1} \int_{x_i}^{x_{i+1}} e' s'''$$

La primera sumatoria es una suma telescópica, por lo que conservamos el primer y último término:

$$\sum_1^{n-1} e'(x) s''(x) \Big|_{x_i}^{x_{i+1}} = e'(b) s''(b) - e'(a) s''(a) = (u'(b) - s'(b)) s''(b) - (u'(a) - s'(b)) s''(a) = 0$$

ya que $u, s \in H$.

En cuanto a la segunda, $s'''|_{[x_i, x_{i+1}]}$ es constante, por lo que podemos sacarlo de la integral:

$$\sum_1^{n-1} s_i \int_a^b e'(x) = \sum_1^{n-1} s_i (e(b) - e(a)) = 0$$

Es decir, $\langle e'', s'' \rangle = 0$. Por tanto:

$$\|u''\|^2 = \|e''\|^2 + \|s''\|^2 + 2 \langle e'', s'' \rangle = \|e''\|^2 + \|s''\|^2 \geq \|s''\|^2$$

donde utilizamos que la norma siempre es positiva. □

Así, podemos observar que el **spline cúbico sujeto** asociado a una función f tiene la menor norma de su segunda derivada de entre las que interpolan a f en una partición dada, por lo que resuelve nuestro problema.

2.2.1. Cota de error en los splines cúbicos

Teorema. Sea $f \in C^4([a, b])$, $n \in \mathbb{N}$, $P = \{x_i\}_{i=0 \dots n} \in \mathcal{P}([a, b])$ y $s \in S_3^1(P)$ spline para f . Además, sean $h = \max\{x_i - x_{i-1}\}_{i=1 \dots n}$, $M > 0$ cota superior de $|f^{iv}|$ en $[a, b]$ y $E = f - s$, $x \in [a, b]$.

Se verifica:

$$|E(x)| \leq \frac{5M}{384} h^4 \tag{2}$$

La demostración, así como cotas para las derivadas, puede consultarse en *Optimal Error Bounds for Cubic Spline Interpolation*, Charles Hall y Weston Meyer, (1976).

2.3. Ejemplos

3. Implementación en ordenador: Octave

Hemos implementado las siguientes funciones en Octave:

1. **SplineLineal**: Calcula spline **lineal**. (*Usado en los splines cúbicos*)
2. **Spline31**: Calcula spline de **clase 1**.
3. **SplineNat**: Calcula spline **natural**.
4. **SplinePer**: Calcula spline **periódico**.
5. **SplineSuj**: Calcula spline **sujeto**.
6. **SplineCuad**: Calcula spline **cuadrático** de clase 1.

3.1. Spline Lineal

La función que nos permite calcular un spline lineal es muy

```
function s = SplineLineal(x,y)
    p = diff(y)./diff(x);
    A = [p' y(1:end-1)'];
    s = mkpp(x,A);
end
```

3.2. Splines cuadráticos

Utilizando el sistema que vimos anteriormente, podemos definir fácilmente una función que calcule los coeficientes de un spline cuadrático de clase 1:

```
function s = coefsSplineCuad(x, y, d_k, k)
    # Número de intervalos
    n = length(x) - 1;

    # 1, x, x^2
    A(:,1) = [ones(n+1,1); 0];
    A(:,2) = [x' ; 1];
    A(:,3) = [x'.^2 ; 2.*x(k+1)];

    # Potencias truncadas
    for j = 4 : n + 2
        pot = @(t) (t > x(j-2)) .* (t - x(j-2));
        A(:,j) = [pot(x').^2; 2.*pot(x(k+1))];
    end

    # Resolución del sistema
    s = A \ [y' ; d_k];

end
```

A. Definiciones y notación

Definición. Sea $I \subset \mathbb{R}$ un intervalo cerrado y acotado con extremos a, b :

- Una **partición** P de I es un subconjunto finito de I con $a, b \in P$.
- $\mathcal{P}(I)$ es el conjunto de todas las particiones de I .

Definición. Sea $a \in \mathbb{R}$, $n \in \mathbb{N}$. La **potencia truncada** en a de grado n , $(x - a)_+^n$ viene dada por:

$$(x - a)_+^n = \begin{cases} 0 & \text{si } x \leq a \\ (x - a)^n & \text{si } x > a \end{cases}$$

Cualquier potencia truncada de grado n es de clase $n - 1$, y su derivada de orden n presenta una discontinuidad en a . La derivada de $(x - a)_+^n$ en x es $n(x - a)_+^{n-1}$.

Su implementación en Octave es bastante sencilla: dados **a** y **n**, podemos definir la potencia truncada como función anónima de la siguiente forma:

```
pot = @(x) (x > a) * (x - a)^n
```

Como Octave tiene tipos dinámicos convertirá $(x > a)$ a 1 si $x > a$ y a 0 en otro caso.