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

1.1. Introducción a los splines

Definición. Sea $[a,b]$ un intervalo, $P = \{x_i\}_{i=0...n} \in \mathscr{P}([a,b])$, $k, r \in \mathbb{N}$, $r < k$. Se dice que $s : [a,b] \to \mathbb{R}$ 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 $P = \{x_i\}_{i=0...n} \in \mathscr{P}([a,b])$. 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 nos proporcionará condiciones interesantes para resolver problemas de interpolantes con este tipo de splines.

Describamos a continuación este espacio.

Proposición. *Sea $[a,b]$ intervalo, $P = \{x_i\}_{i=0...n} \in \mathscr{P}([a,b])$, 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:

$$s_i(x_i) = s_{i+1}(x_i) \qquad s_i'(x_i) = s_{i+1}'(x_i)$$

para todo $i = 1...n-1$. 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 representativa del espacio de splines cuadráticos con el uso de potencias truncadas. Una **base del espacio** es : $\{1, x, x^2, (x - x_1)_+^2, ..., (x - x_{n-1})_+^2\}$.

1.3. Interpolación con splines cuadráticos

1.4. Error en los splines cuadráticos

Teorema. *Sean $f \in C^2([a,b])$, $\{x_i\}_{i=0...n} \in \mathscr{P}([a,b])$, $s \in S_2^1(\{x_i\}_{i=0...n})$ spline para f,*

$h = max\{x_i - x_{i-1}\}_{i=1...n}$, $E = f - s$. 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 $x \in [a, b]$*:*

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

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

2. Splines cúbicos

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

2.1.1. Splines cúbicos a partir de las segundas derivadas:

Uno de los problemas de la interpolación polinomial es que, al ir aumentando los nodos (diferentes), el grado del polinomio aumenta $(gr(p)) = n - 1)$. Esto conlleva unas fluctuaciones en los extremos de la interpolación. (1)

Sin embargo, si dividimos el intervalo en una partición $P = \{x_i\}_{i=0\ldots n} \in \mathscr{P}([t_0, t_n])$, con un serie de subintervalos, podemos aproximar un polinomio en cada intervalo minimizando la cota de error.

Para no volver a tener el problema de las fluctuaciones, indeseables en la mayoría de las aplicaciones, se suelen utilizar polinomios interpolantes de grado ≤ 3. Esta técnica se conoce como aproximación polinomial fragmentaria, donde:

Spline cúbicos: La aproximación más utilizada es la interpolación con splines cúbicos debido a que proporciona un excelente ajuste a los puntos tabulados y su cálculo no es excesivamente complejo.

Propiedades:

Dada una función definida en $[a, b]$, una partición del intervalo $P = \{x_i\}_{i=0\ldots n} \in \mathscr{P}([a, b])$:

- S es un polinomio cúbico denotado por S_j en el subintervalo de extremos x_j y x_{j+1}, para $j = 0, 1, ..n - 1$.
- $S_j(x_j) = f(x_j)$ y $S_j(x_{j+1}) = f(x_{j+1})$
- $S'_{j+1}(x_{j+1}) = S'_j(x_{j+1})$
- $S''_{j+1}(x_{j+1}) = S''_j(x_{j+1})$

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

1. Los splines cúbicos de clase 1 son continuos y derivables en su dominio. Son un espacio vectorial de dimensión 2*(n+1), cuya base es: .

Una desventaja de estos splines es que no se asegura que haya derivabilidad en los extremos, en un contexto geométrico eso 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 S_k^r es $(k - r)n + r + 1$ la dimensión de este espacio vectorial es $(3 - 2)n + 2 + 1 = n + 3$. Cuando $k = r + 1$ el superíndice se omite.

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

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:

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

Planteamos el problema:

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

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

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

$$||u''||^2 = ||e'' + s''||^2 = ||e''||^2 + ||s''||^2 + 2 < e'', s'' >$$

Dividimos $< e'', s'' >$ en intervalos:

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

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, $< e'', s'' >= 0$. Por tanto:

$$||u''||^2 = ||e''||^2 + ||s''||^2 + 2 < e'', s'' >= ||e''||^2 + ||s''||^2 \geq ||s''||^2$$

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* $f \in C^4([a,b])$, $n \in \mathbb{N}$, $P = \{x_i\}_{i=0\ldots n} \in \mathscr{P}([a,b])$ *y* $s \in S_3^1(P)$ *spline para* f. *Además, sean* $h = max\{x_i - x_{i-1}\}_{i=1\ldots n}$, $M > 0$ *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, x²
  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 $a, b \in P$.
- $\mathscr{P}(I)$ 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.