

Eduardo González Díaz

Funciones Spline

Spline Functions

Trabajo Fin de Grado
Grado en Matemáticas
La Laguna, Septiembre de 2017

DIRIGIDO POR
Ruymán Cruz Barroso

Ruymán Cruz Barroso

*Departamento de Análisis Ma-
temático*

Universidad de La Laguna

38271 La Laguna, Tenerife

Agradecimientos

Me gustaría agradecer a mi familia sus incondicionales muestras de apoyo, que han contribuido a superar esta larga y en ocasiones difícil etapa de formación, puedo asegurar que sin su ayuda no podría haber llegado hasta aquí, no podría haber alcanzado el tan deseado objetivo de concluir mis estudios en Matemáticas.

Asimismo me gustaría agradecer a mi tutora académica y actualmente Vicedecana en Matemáticas D^a. María Candelaria González Dávila, al margen de lo estrictamente académico, su predisposición a ayudar al alumnado durante todos estos años. En especial he de reconocer que sin su ayuda y colaboración me hubiera sido prácticamente imposible afrontar los problemas burocráticos surgidos en la adaptación del plan de estudio antiguo Licenciatura en Matemáticas al actual Grado en Matemáticas.

Por supuesto, trasladar mi más sincero reconocimiento al profesor D. Ruymán Cruz Barroso, tutor en este Trabajo Fin de Grado por su aportación e inestimable colaboración para la realización del mismo.

Resumen · Abstract

Resumen

La interpolación polinómica consiste en aproximar una función deseada por una función polinómica, generada a partir de un número finito de datos conocidos. Esta herramienta es de uso frecuente en multitud de procesos numéricos de la Matemática Aplicada y en diversas áreas de la Ingeniería. Dada la importancia de este tema por sus numerosas aplicaciones, en esta memoria se pretende reforzar conocimientos adquiridos sobre la interpolación polinómica clásica, e introducir y estudiar el concepto de Funciones Spline, una herramienta dentro de la interpolación que se ajusta convenientemente a ciertas actividades de la vida cotidiana.

Esta memoria se ha estructurado en tres capítulos. El primero está dedicado a la interpolación polinómica clásica. En el segundo se presenta la interpolación mediante funciones spline, y se estudian sus propiedades más relevantes. En el último capítulo se estudian las funciones B-spline y su importancia teórica. Además, numerosos ejemplos y aplicaciones son presentados a lo largo de todo el trabajo.

El objetivo principal de este trabajo es introducir el concepto de Funciones Spline y estudiar sus propiedades más relevantes. Estas funciones nos proporcionarán soluciones a los problemas que se puedan generar en la interpolación polinómica al aplicar los métodos de Newton o de Lagrange para tablas grandes de datos, suavizando así el comportamiento de nuestra función interpolante. Nos aportarán una base que nos permitirá construir y trabajar con polinomios sencillos a trozos de manera eficiente.

Palabras clave: *Interpolación polinómica – Funciones Spline – Funciones B-Spline – Interpolación por funciones spline.*

Abstract

The polynomial interpolation consists in approximating a desired function by a polynomial function, generated from a finite number of known data. This tool is often used in many numerical processes of Applied Mathematics and in diverse areas of Engineering. Due to the importance of the subject because its many applications, this report aims to strengthen the knowledge acquired about the classic polynomial interpolation, and introduce and study the concept of Spline Functions, a tool within the polynomial interpolation that fits conveniently to certain activities of the daily life.

This report has been structured in three chapters. The first one is dedicated to the classical polynomial interpolation. In the second chapter, interpolation by spline functions is presented, and their most relevant properties are studied. In the last chapter we study the B-Spline Functions, and their theoretical importance. Also, several numerical examples and applications are carried out throughout the report.

The main aim of this work is to introduce the concept of Spline Functions and to study its most relevant properties. These functions will provide solutions to us about to the problems that could be generated in the polynomial interpolation by applying Newton or Lagrange methods for large data tables, softening the behavior of our interpolating function. They will provide us with a basis that will allow us to construct and to work with simple piecewise polynomials of an efficient way.

Keywords: *Polynomial interpolation - Spline Functions - B-Spline Functions - Spline Interpolation.*

Contenido

| | |
|---|-----|
| Agradecimientos | III |
| Resumen/Abstract | V |
| Introducción | IX |
| 1. Interpolación polinómica | 1 |
| 1.1. El problema de la interpolación polinómica. | 1 |
| 1.2. Método de Newton para el cálculo del polinomio de interpolación. | 4 |
| 1.3. Método de Lagrange para el cálculo del polinomio de interpolación. | 8 |
| 1.4. Convergencia de la interpolación polinómica. Algunos resultados notables. | 10 |
| 1.5. Ventajas y desventajas de los métodos. | 12 |
| 2. Funciones Spline | 15 |
| 2.1. Funciones Spline. | 15 |
| 2.2. Funciones Spline cúbicas interpolantes. | 17 |
| 2.3. Funciones Spline de grado superior. | 24 |
| 3. Funciones B-Spline | 27 |
| 3.1. Funciones B-Spline. | 27 |
| 3.2. Propiedades de las funciones B-Spline. | 30 |
| 3.3. Aplicaciones de las funciones B-Spline. | 35 |
| Bibliografía | 37 |
| Apéndice | 39 |
| Poster | 41 |

Introducción

La interpolación polinómica es una herramienta recurrente en diferentes ramas de la Ciencia y la Ingeniería, ya que permite aproximar el comportamiento de los distintos estudios mediante una función polinómica, interpolando a partir de un número finito de datos, o bien, en el sentido de la mejor aproximación en normas euclídeas e infinito. Dada la importancia de este tema por sus numerosas aplicaciones, y tras lo estudiado en la asignatura *Métodos Numéricos II* [5] del actual Grado en Matemáticas de la Universidad de La Laguna, donde se imparte el problema de interpolación polinómica y el Teorema de Aproximación de Weierstrass, uno de los objetivos de este *Trabajo de Fin de Grado* es el profundizar y ampliar los conocimientos estudiados sobre la interpolación polinómica clásica, y analizar las ventajas e inconvenientes de los métodos más notables conocidos en la literatura.

La interpolación polinómica, aplicando el método de Newton o el de Lagrange, es útil en situaciones que requieren un número bajo de puntos para interpolar. Sin embargo, a medida que crece el número de nodos, también lo hace el grado del polinomio interpolador, y éste se convierte en una función muy oscilante. Además, en ambos métodos, la dificultad de los cálculos también aumenta al incrementar el número de nodos. Con el fin de evitar estas adversidades se recurre a otros tipos de funciones interpolantes, como pueden ser la interpolación racional (útil si la función a interpolar presenta singularidades) o trigonométrica (útil si ésta es una función periódica).

Una idea razonable para evitar las oscilaciones de la función interpoladora cuando el número de puntos a interpolar es elevado sería, en lugar de obtener un único polinomio de grado n definido en todo el intervalo $[a, b]$, dividir el intervalo en un número determinado de subintervalos, para luego buscar un polinomio de menor grado fijo que interpole a nuestra función en éstos. Esta es precisamente

la idea de la interpolación mediante **funciones spline**, que se desarrolló en la década de 1940 y que serán funciones con un cierto grado de regularidad definidas a trozos mediante polinomios. Esta interpolación da lugar a resultados similares requiriendo solamente el uso de polinomios de bajo grado, evitando así indeseables oscilaciones.

La facilidad de cálculo computacional de las funciones spline nos permiten representar gráficamente curvas con formas complicadas de manera sencilla y eficiente. Debido a esto la interpolación por funciones spline se ha convertido en una herramienta muy popular de la Matemática Aplicada y en el diseño de gráficos por ordenador.

Las fuentes bibliográficas más frecuentemente consultadas durante la realización de esta memoria han sido [2] y [5].

Interpolación polinómica

En este capítulo se tratará el problema de la interpolación polinómica clásica. Se estudiarán los métodos de Newton y de Lagrange y se realizarán algunos ejemplos ilustrativos sencillos. También analizaremos las ventajas y desventajas entre ambos métodos y sus problemas en aplicaciones reales. Motivaremos finalmente el uso de *funciones spline* en el problema de interpolación.

1.1. El problema de la interpolación polinómica.

En esta sección se caracterizará el siguiente problema: *Dada una tabla de $n + 1$ puntos (x_i, y_i) , con $i = 0, 1, \dots, n$, con $y_i \in \mathbb{R}$,*

$$\begin{array}{c|c|c|c|c|c} x & x_0 & x_1 & x_2 & \cdots & x_n \\ \hline y & y_0 & y_1 & y_2 & \cdots & y_n \end{array},$$

buscar un polinomio p con el menor grado posible, tal que $p(x_i) = y_i$, $0 \leq i \leq n$. Se dice que el polinomio p interpola los datos. A lo largo de este trabajo, \mathbb{P}_n denotará al espacio vectorial de polinomios con coeficientes reales y grado a lo sumo n y $\mathcal{C}^k([a, b])$ al espacio de funciones continuas en $[a, b]$ con derivadas continuas hasta orden k . Comenzamos enunciando y demostrando el teorema que garantiza la existencia y unicidad de este problema.

Nota. El desarrollo teórico es igualmente válido si $y_i \in \mathbb{C}$.

Teorema 1.1. *Sean x_0, x_1, \dots, x_n números reales distintos. Entonces, para valores arbitrarios reales y_0, y_1, \dots, y_n existe un único polinomio $p_n \in \mathbb{P}_n$, tal que $p_n(x_i) = y_i$, $0 \leq i \leq n$.*

Demostración. Veamos primero la unicidad. Supongamos que existen $p_n, q_n \in \mathbb{P}_n$ cumpliendo las condiciones anteriores. Sea $r_n := p_n - q_n \in \mathbb{P}_n$, con grado a

lo sumo n . Como $r_n(x_i) = 0, \forall i = 0, 1, \dots, n$, se concluye que $r_n \equiv 0$.

Ofreceremos ahora dos demostraciones para la existencia, inspiradas en los métodos para el cálculo del polinomio interpolador que estudiaremos en los próximos apartados 1.2 y 1.3.

Newton:

Para probar la existencia procedemos por inducción sobre n . Para $n = 0$ la existencia es obvia: $p_0(x) \equiv y_0$. Supongamos ahora que hemos obtenido un polinomio $p_{k-1} \in \mathbb{P}_{k-1}$, con $p_{k-1}(x_i) = y_i$, para $0 \leq i \leq k-1$. Intentemos construir p_k de la siguiente manera:

$$p_k(x) = p_{k-1}(x) + c(x-x_0)(x-x_1)\cdots(x-x_{k-1}) \in \mathbb{P}_k, \quad c \in \mathbb{R}.$$

Es evidente que p_k interpola los datos que interpola p_{k-1} dado que $p_k(x_i) = p_{k-1}(x_i) = y_i, 0 \leq i \leq k-1$. Procedamos ahora a determinar el coeficiente desconocido c a partir de la condición $p_k(x_k) = y_k$. Esto lleva a la ecuación

$$p_{k-1}(x_k) + c(x_k - x_0)(x_k - x_1)\cdots(x_k - x_{k-1}) = y_k, \quad (1.1)$$

de la cual se despeja unívocamente el valor de c dado que $x_i \neq x_j$ si $i \neq j$. Concluimos así la primera demostración.

Lagrange:

Las condiciones de interpolación $p(x_i) = y_i$, para $0 \leq i \leq n$, conducen a un sistema lineal de $n+1$ ecuaciones con $n+1$ incógnitas c_0, c_1, \dots, c_n , donde $p_n(x) = \sum_{i=0}^n c_i x^i$:

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix} \cdot \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}. \quad (1.2)$$

La matriz asociada a este sistema, que denotaremos por $V(x_0, \dots, x_n)$, se conoce como matriz de **Vandermonde** y es fácil comprobar que

$$\det(V(x_0, \dots, x_n)) = \prod_{i < j} (x_i - x_j).$$

Por tanto, la elección de nodos distintos garantiza que para cualquier conjunto de datos y_0, \dots, y_n , el problema tendrá solución única. \square

Para sistemas grandes la resolución del sistema (1.2) resulta costosa y es posible que presente inestabilidad numérica, por lo que desde el punto de vista computacional esta fórmula no es una vía eficiente.

Enunciamos a continuación un resultado que permite estimar el error que se comete al aproximar una función f con ciertas condiciones de regularidad que se evalúa en $(n + 1)$ puntos por su polinomio interpolador.

Teorema 1.2. *Sea $f \in \mathcal{C}^{n+1}([a, b])$, y sea $p \in \mathbb{P}_n$ el polinomio que interpola a f en $x_0, x_1, \dots, x_n \in [a, b]$, con $x_i \neq x_j$ para $i \neq j$. Entonces, $\forall x \in [a, b]$, existe $\xi_x \in (a, b)$ tal que:*

$$f(x) - p(x) = \frac{1}{(n + 1)!} f^{(n+1)}(\xi_x) \prod_{i=0}^n (x - x_i).$$

Obsérvese que en particular se cumple que:

$$|f(x) - p(x)| \leq \frac{M}{(n + 1)!} \prod_{i=0}^n |x - x_i|, \quad M = \max_{x \in [a, b]} |f^{(n+1)}(x)|.$$

Demostración. Nótese que el resultado es trivialmente cierto si el punto x es uno de los nodos x_i , dado que ambos miembros de la igualdad son nulos. Podemos tomar por tanto un punto $t \in [a, b]$ con $t \neq x_i, \forall i = 0, \dots, n$. Entonces definimos

$$E(x) = f(x) - p(x), \quad \text{y} \quad G(x) = E(x) - \frac{\pi(x)}{\pi(t)} E(t), \quad \forall x \in [a, b],$$

con $\pi(x) = \prod_{i=0}^n (x - x_i)$.

La función $G(x) \in \mathcal{C}^{n+1}([a, b])$, ya que $E(x)$ y $\pi(x)$ lo son. Además,

$$G(x_i) = E(x_i) - \frac{\pi(x_i)}{\pi(t)} E(t) = 0, \quad \forall i = 0, \dots, n, \quad \text{y} \quad G(t) = E(t) - E(t).$$

Por tanto, G tiene $n + 2$ ceros distintos en $[a, b]$, aplicando el Teorema del Valor Medio, G' tendrá $n + 1$ ceros distintos. Repitiendo este argumento, $G^{(j)}(x)$ tiene $n + 2 - j$ ceros en $[a, b], \forall j = 0, \dots, n + 1$. Sea ξ_x un cero de $G^{(n+1)}(x)$,

$$G^{(n+1)}(\xi_x) = 0.$$

Dado que,

$$E^{(n+1)}(x) = f^{(n+1)}(x), \quad \text{y} \quad \pi^{(n+1)}(x) = (n + 1)!,$$

tenemos que

$$G^{(n+1)}(x) = f^{(n+1)}(x) - \frac{(n + 1)!}{\pi(t)} E(t).$$

Sustituyendo $x = \xi_x$ y despejando $E(t)$,

$$E(t) = \frac{\pi(t)}{(n + 1)!} f^{(n+1)}(\xi_x)$$

y obtenemos el resultado deseado. □

Un resultado importante que justifica el aproximar funciones continuas en $[a, b]$ por polinomios es el **teorema de aproximación de Weierstrass** (véase la demostración por ejemplo en [1, pág. 198]). Este resultado establece que el espacio $\mathbb{P}([a, b])$ de polinomios definidos en $[a, b]$ es denso en $\mathcal{C}([a, b])$ con respecto a la norma uniforme.

Teorema 1.3 (Weierstrass). *Sea $f \in \mathcal{C}([a, b])$ y $\varepsilon > 0$. Entonces existe un polinomio p que satisface $\|f - p\|_\infty := \max_{x \in [a, b]} |f(x) - p(x)| < \varepsilon$.*

Aunque solo existe un único polinomio interpolador de grado mínimo para una tabla de datos dada, existen diferentes formas de calcularlo. En las dos secciones siguientes abordaremos los dos métodos más conocidos en la literatura.

1.2. Método de Newton para el cálculo del polinomio de interpolación.

El método de Newton es un proceso recursivo que permite computar el polinomio interpolador a partir de una tabla de datos. Los polinomios p_0, p_1, \dots, p_n generados que cumplen $p_k(x_i) = y_i, \forall 0 \leq i \leq k$ poseen la propiedad de que para todo $k \geq 1$, cada p_k se obtiene añadiendo un sumando al anterior p_{k-1} , siendo por tanto un proceso dinámico:

$$\begin{aligned} p_0(x) &\equiv c_0 = y_0, \\ p_k(x) &= p_{k-1}(x) + c_k(x - x_0) \cdots (x - x_{k-1}), \quad \forall k \geq 1. \end{aligned} \quad (1.3)$$

El coeficiente c_k de esta ecuación se corresponde con el coeficiente c de la ecuación (1.1), que puede calcularse según

$$c_k = \frac{y_k - p_{k-1}(x_k)}{(x_k - x_0)(x_k - x_1) \cdots (x_k - x_{k-1})}. \quad (1.4)$$

La forma compacta para la expresión anterior que genera sucesivamente los polinomios $p_k \in \mathbb{P}_k$ viene dada por:

$$p_k(x) = \sum_{i=0}^k c_i \prod_{j=0}^{i-1} (x - x_j), \quad (1.5)$$

donde por convenio, $\prod_{j=0}^m (x - x_j) = 1$, si $m < 0$. Estos polinomios reciben el nombre de **polinomios de interpolación en la forma de Newton**.

Como durante el proceso aparece la evaluación de polinomios, debemos recordar que una manera eficiente para hacerlo es el método conocido como **multiplicación anidada** o **Algoritmo de Horner**, ya que este minimiza el

número de operaciones a realizar. Puede ser explicado de manera sencilla para una expresión general de la forma

$$u = \sum_{i=0}^k c_i \prod_{j=0}^{i-1} d_j = c_0 + c_1 d_0 + c_2 d_0 d_1 + \cdots + c_k d_0 d_1 \cdots d_{k-1}.$$

La idea es escribirla como:

$$u = c_0 + d_0(c_1 + \cdots + d_{k-3}(c_{k-2} + d_{k-2}(c_{k-1} + d_{k-1} \cdot c_k)) \cdots).$$

Si se procediera a evaluar el polinomio monomio a monomio, el número de operaciones necesarias sería del orden de n sumas y $\frac{n \cdot (n+1)}{2}$ productos, mientras que aplicando el algoritmo de Horner, como se describe a continuación:

$$\begin{cases} b_0 = c_k, \\ b_i = c_{k-i} + d_{k-i} \cdot b_{i-1}, \quad \forall i = 1, \dots, k, \end{cases}$$

tendríamos que $u = b_k$, y el número de operaciones realizadas se reduce a n sumas y n productos.

A continuación veamos un ejemplo.

Ejemplo 1. Encontrar el polinomio de menor grado que interpola la siguiente tabla.

| | | | | |
|-----|----|-----|---|---|
| x | 3 | 7 | 1 | 2 |
| y | 12 | 146 | 2 | 1 |

Nota. Obsérvese que la tabla no tiene por qué estar ordenada de la forma $x_i < x_j$ si $i < j$.

Solución. Para resolverlo utilizaremos las fórmulas (1.4) y (1.5) vistas anteriormente.

$$c_0 = y_0;$$

$$p_0(x) = y_0 = 12$$

$$c_1 = \frac{146-12}{7-3} = \frac{134}{4} = \frac{67}{2};$$

$$p_1(x) = 12 + \frac{67}{2}(x-3) \implies p_1(1) = -55$$

$$c_2 = \frac{2+55}{(1-3)(1-7)} = \frac{57}{12} = \frac{19}{4};$$

$$p_2(x) = 12 + \frac{67}{2}(x-3) + \frac{19}{4}(x-3)(x-7) \implies p_2(2) = \frac{9}{4}$$

$$c_3 = \frac{1 - \frac{9}{4}}{(2-3)(2-7)(2-1)} = \frac{-1}{4};$$

$$\begin{aligned} p_3(x) &= 12 + \frac{67}{2}(x-3) + \frac{19}{4}(x-3)(x-7) - \frac{1}{4}(x-3)(x-7)(x-1) \\ &= -\frac{1}{4}x^3 + \frac{15}{2}x^2 - \frac{87}{4}x + \frac{33}{2}. \end{aligned}$$

El cálculo de los coeficientes c_j , $0 \leq j \leq n$ del método de Newton puede efectuarse mediante el cálculo de las **diferencias divididas de Newton**, definidas por:

$$\begin{aligned} f[x_i] &= f(x_i), \quad 0 \leq i \leq n; \\ f[x_i, x_{i+1}, \dots, x_{i+j}] &= \frac{f[x_{i+1}, \dots, x_{i+j}] - f[x_i, \dots, x_{i+j-1}]}{x_{i+j} - x_i}, \quad 0 \leq i + j \leq n. \end{aligned} \quad (1.6)$$

Las diferencias $f[x_i]$, $f[x_i, x_{i+1}]$, $f[x_i, x_{i+1}, x_{i+2}]$, $f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$, ... son respectivamente las diferencias divididas de orden 0, 1, 2, 3, etcétera (diferencias divididas de orden superior). Estas cantidades proporcionan los coeficientes del método de Newton. Para su cálculo se suele proceder construyendo la tabla de diferencias divididas. Para el caso particular $n = 3$ (análogo para n en general) a partir de los pares $(x_i, f(x_i))$ de nuestro problema se construye según:

$$\begin{array}{l} \mathbf{x}_0 \mathbf{f}[\mathbf{x}_0] \quad | \quad f[x_0, x_1] \quad f[x_0, x_1, x_2] \quad f[x_0, x_1, x_2, x_3] \\ \mathbf{x}_1 \mathbf{f}[\mathbf{x}_1] \quad | \quad f[x_1, x_2] \quad f[x_1, x_2, x_3] \\ \mathbf{x}_2 \mathbf{f}[\mathbf{x}_2] \quad | \quad f[x_2, x_3] \\ \mathbf{x}_3 \mathbf{f}[\mathbf{x}_3] \quad | \end{array}$$

Algunas propiedades importantes de las diferencias divididas son las siguientes (véanse las demostraciones por ejemplo en [1, págs. 139-144]):

Teorema 1.4. *La diferencia dividida es una función simétrica en sus argumentos. Es decir, si (z_0, z_1, \dots, z_n) es una permutación de (x_0, x_1, \dots, x_n) entonces*

$$f[z_0, z_1, \dots, z_n] = f[x_0, x_1, \dots, x_n].$$

Esta propiedad justifica, por ejemplo, que para el cálculo del polinomio interpolador no sea necesario partir de una tabla de datos ordenada de datos, $x_i < x_j$ si $i < j$.

Teorema 1.5. *Sea $p \in \mathbb{P}_n$ el polinomio que interpola a la función f en los $n+1$ nodos distintos x_0, x_1, \dots, x_n . Si t es un punto diferente de los nodos, entonces*

$$f(t) - p(t) = f[x_0, x_1, \dots, x_n, t] \prod_{j=0}^n (t - x_j).$$

Este resultado nos permite obtener una estimación del error que se comete al aproximar el valor de la función f por evaluaciones en su polinomio interpolador.

Teorema 1.6. *Sea $f \in \mathcal{C}^n([a, b])$ y x_0, x_1, \dots, x_n puntos distintos en $[a, b]$. Entonces hay un punto $\xi_x \in (a, b)$ tal que*

$$f[x_0, x_1, \dots, x_n] = \frac{1}{n!} f^{(n)}(\xi_x).$$

Este enunciado permite en particular establecer la cota

$$f[x_0, x_1, \dots, x_n] \leq \frac{M}{n!}, \quad M = \max_{x \in [a, b]} |f^{(n)}(x)|,$$

útil para el Teorema 1.5. (Véase el Teorema 1.2).

Ejemplo 2. Consideremos la misma tabla del Ejemplo 1, y calculemos el polinomio interpolador empleando la fórmula de Newton, pero esta vez mediante diferencias divididas.

Solución.

| | | | | |
|----------|------------|------|------|------|
| 3 | 12 | 67/2 | 19/4 | -1/4 |
| 7 | 146 | 24 | 5 | |
| 1 | 2 | -1 | | |
| 2 | 1 | | | |

$$f[x_0, x_1] = \frac{146-12}{7-3} = \frac{67}{2} = c_1;$$

$$f[x_1, x_2] = \frac{2-146}{1-7} = 24;$$

$$f[x_2, x_3] = \frac{1-2}{2-1} = -1;$$

$$f[x_0, x_1, x_2] = \frac{24-\frac{67}{2}}{1-3} = \frac{19}{4} = c_2;$$

$$f[x_1, x_2, x_3] = \frac{-1-24}{2-7} = 5;$$

$$f[x_0, x_1, x_2, x_3] = \frac{5-\frac{19}{4}}{2-3} = -\frac{1}{4} = c_3;$$

$$p_3(x) = 12 + \frac{67}{2}(x-3) + \frac{19}{4}(x-3)(x-7) - \frac{1}{4}(x-3)(x-7)(x-1)$$

$$= -\frac{1}{4}x^3 + \frac{15}{2}x^2 - \frac{87}{4}x + \frac{33}{2}.$$

Finalizamos esta sección añadiendo otro ejemplo ilustrativo:

Ejemplo 3. Escribir el polinomio de interpolación de Newton para la siguiente tabla:

| | | | | |
|--------|----|----|---|----|
| x | 4 | 2 | 0 | 3 |
| $f(x)$ | 63 | 11 | 7 | 28 |

$$\text{Solución.} \quad \begin{array}{ccc|ccc} 4 & 63 & & 26 & 6 & 1 \\ 2 & 11 & & 2 & 5 & \\ 0 & 7 & & 7 & & \\ \hline 3 & 28 & & & & \end{array}$$

$$f[x_0, x_1] = \frac{11-63}{2-4} = 26 = c_1;$$

$$f[x_1, x_2] = \frac{7-11}{0-2} = 2;$$

$$f[x_2, x_3] = \frac{28-7}{3-0} = 7;$$

$$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} = \frac{2-26}{0-4} = 6 = c_2;$$

$$f[x_1, x_2, x_3] = \frac{f[x_2, x_3] - f[x_1, x_2]}{x_3 - x_1} = \frac{7-2}{3-2} = 5;$$

$$f[x_0, x_1, x_2, x_3] = \frac{f[x_1, x_2, x_3] - f[x_0, x_1, x_2]}{x_3 - x_0} = \frac{5-6}{3-4} = 1 = c_3;$$

$$p_3(x) = 63 + 26(x-4) + 6(x-4)(x-2) + (x-4)(x-2)x = x^3 - 2x + 7.$$

1.3. Método de Lagrange para el cálculo del polinomio de interpolación.

Veamos ahora un método alternativo para hallar el polinomio de interpolación p asociado a una tabla de datos (x_i, y_i) , con $0 \leq i \leq n$. Recordemos que si asumimos que los nodos x_i son distintos, existirá un único polinomio de interpolación en \mathbb{P}_n para datos arbitrarios y_0, \dots, y_n .

El método de interpolación de Lagrange proporcionará una vía alternativa a la de Newton para el cálculo de éste, expresando el polinomio p en forma de Lagrange:

$$p(x) = \sum_{i=0}^n y_i l_i(x), \quad (1.7)$$

donde $l_i \in \mathbb{P}_n \setminus \mathbb{P}_{n-1}, \forall i = 0, 1, \dots, n$. A estos polinomios l_i se les conoce como **polinomios fundamentales de Lagrange**, solamente dependen de los nodos, y tienen la propiedad $l_i(x_j) = \delta_{i,j}$ siendo $\delta_{i,j}$ la función delta de Kronecker.

Sea l_0 un polinomio de grado exacto n que toma el valor 0 en todo x_i para $1 \leq k \leq n$ y el valor 1 en x_0 . Entonces l_0 debe ser de la forma:

$$l_0(x) = c \prod_{j=1}^n (x - x_j).$$

El valor de c que garantiza $l_0(x_0) = 1$ se obtiene haciendo $x = x_0$, luego

$$1 = c \prod_{j=1}^n (x_0 - x_j) \implies c = \prod_{j=1}^n (x_0 - x_j)^{-1}.$$

Por tanto, podemos escribir:

$$l_0(x) = \prod_{j=1}^n \frac{x - x_j}{x_i - x_j}.$$

Razonando de igual manera obtenemos el resto de los polinomios fundamentales de Lagrange:

$$l_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j}, \quad \forall 0 \leq i \leq n. \quad (1.8)$$

Esto nos permite calcular el polinomio de interpolación de Lagrange según la fórmula (1.7).

Ejemplo 4. Consideremos la tabla de datos del Ejemplo 1, y calculamos ahora el polinomio interpolador mediante la fórmula de Lagrange.

Solución. Para ello utilizaremos las fórmulas (1.7) y (1.8) vistas anteriormente:

$$\begin{aligned} l_0 &= \frac{(x-7)(x-1)(x-2)}{(3-7)(3-1)(3-2)}, & l_1 &= \frac{(x-3)(x-1)(x-2)}{(7-3)(7-1)(7-2)}, \\ l_2 &= \frac{(x-3)(x-7)(x-2)}{(1-3)(1-7)(1-2)}, & l_3 &= \frac{(x-3)(x-7)(x-1)}{(2-3)(2-7)(2-1)}, \end{aligned}$$

por lo que

$$\begin{aligned} p(x) &= y_0 l_0 + y_1 l_1 + y_2 l_2 + y_3 l_3 = 12l_0 + 146l_1 + 2l_2 + l_3 \\ &= -\frac{1}{4}x^3 + \frac{15}{2}x^2 - \frac{87}{4}x + \frac{33}{2}. \end{aligned}$$

Al igual que en la sección anterior finalizamos con otro ejemplo ilustrativo:

Ejemplo 5. Encontrar el polinomio de menor grado que interpola estos puntos.

| | | | | | | |
|-----|-----|-----|-----|------|------|------|
| x | 1,5 | 2,7 | 3,1 | -2,1 | -6,6 | 11,0 |
| y | 0,0 | 0,0 | 0,0 | 1,0 | 0,0 | 0,0 |

Solución. $p(x) = y_0 l_0 + y_1 l_1 + y_2 l_2 + y_3 l_3 + y_4 l_4 + y_5 l_5 = l_3$, por tanto

$$\begin{aligned} p(x) &= \frac{(x-1,5)(x-2,7)(x-3,1)(x+6,6)(x-11,0)}{(-2,1-1,5)(-2,1-2,7)(-2,1-3,1)(-2,1+6,6)(-2,1-11,0)} \\ &\approx 0,00019(x-1,5)(x-2,7)(x-3,1)(x+6,6)(x-11,0) \\ &\approx 0,00019x^5 - 0,00221x^4 - 0,00442x^3 + 0,0835x^2 - 0,22353x + 0,17208. \end{aligned}$$

1.4. Convergencia de la interpolación polinómica. Algunos resultados notables.

La interpolación polinómica presenta mayores problemas de estabilidad cuanto mayor sea el número de nodos. En el Teorema 1.2 se ha establecido una cota del error que se comete al aproximar la función por su polinomio interpolador, pero ¿qué podemos decir sobre la convergencia?

De dicho teorema tenemos que $|f(x) - p(x)| \leq \frac{M_n}{(n+1)!} |\pi(x)|$, siendo $\pi(x) = \prod_{i=0}^n (x - x_i)$ y $M_n = \max_{x \in [a,b]} |f^{(n+1)}(x)|$. Luego, parece razonable pensar que la convergencia dependerá de la elección de nodos, y que nuestro objetivo ha de ser escogerlos de tal forma que se minimice $|\pi(x)|$.

A pesar de que no es recomendable trabajar con un número grande de nodos en la interpolación polinómica clásica, podría plantearse la siguiente cuestión: ¿Aumentar el número de nodos hará que el error cometido tienda a cero? Debemos tener en cuenta que el error depende del comportamiento de las derivadas de f , así que el aumento en el número de nodos no implica necesariamente una mejor aproximación.

Cuando $[a, b] = [-1, 1]$ se conoce que la elección de nodos óptima, es tomar los nodos de Chebyshev de 1ª especie:

$$x_k = \cos \left(\frac{(2k+1) \cdot \pi}{2(n+1)} \right), \quad \forall 0 \leq k \leq n.$$

Utilizando estos nodos y realizando una transformación lineal se comprueba que la fórmula del error resultante viene dada por

$$|f(x) - p(x)| \leq \frac{M_n \cdot (b-a)^{n+1}}{2^{2n+1} \cdot (n+1)!}.$$

La convergencia dependerá por tanto de que el límite de la expresión que se encuentra a la derecha de la desigualdad tienda a cero.

Si se escogen nodos equiespaciados, el error cometido es del orden de

$$|f(x) - p(x)| \leq \frac{M_n}{4(n+1)} \cdot \left(\frac{b-a}{n} \right)^{n+1}.$$

La convergencia nuevamente dependerá por tanto de que el límite de la expresión que se encuentra a la derecha de la desigualdad tienda a cero, en particular, para garantizar la convergencia uniforme será necesario imponer que

$$\lim_{n \rightarrow \infty} \frac{M_n \cdot (b-a)^n}{n!} = 0.$$

A continuación se enunciarán una serie de resultados destacables en la literatura sobre la convergencia en la interpolación polinómica clásica.

Una cuestión que debemos tener clara es la siguiente: dada f una función continua en el intervalo $[-1, 1]$, y nodos equiespaciados. ¿Esto implica convergencia puntual o uniforme en dicho intervalo cuando $n \rightarrow \infty$? La respuesta es que no, como se deduce del famoso **Contraejemplo de Bernstein**. Sea $f(x) = |x|$ en $[-1, 1]$. Esta función es derivable en todo el intervalo salvo en el origen. Sin embargo, los polinomios interpoladores para nodos equiespaciados divergen en $0 < |x| < 1$.

Otro famoso resultado es el **Ejemplo de Runge**: Sea $f(x) = \frac{1}{1+x^2}$ definida en el intervalo $[-5, 5]$, f analítica. Se demuestra que existe $c \approx 3,63$ tal que los polinomios interpoladores convergen si $|x| \leq c$, y divergen en caso contrario. La extensión de $f(x) \in \mathbb{R}$ al plano complejo influye en la convergencia.

Teorema 1.7. *Sea $f(z)$ función entera, real evaluada si $z \in \mathbb{R}$. Entonces los polinomios interpoladores convergen uniformemente a f , para cualquier sistema de nodos.*

Nota. Observar que la función del Ejemplo de Runge no cumple las hipótesis del Teorema 1.7.

Teorema 1.8 (Marcinkiewicz). *Sea f continua en $[a, b]$. Entonces existe un sistema de nodos tal que los polinomios interpoladores convergen uniformemente a f en $[a, b]$.*

Lamentablemente, la demostración de este resultado no es constructiva, es una prueba de existencia. Así que no nos proporciona una forma de conocer dicho sistema de nodos que resultaría de gran utilidad.

Teorema 1.9 (Faber). *Dada una tabla de nodos en $[a, b]$. Entonces existe f continua en $[a, b]$ tal que los polinomios interpoladores no convergen uniformemente a f .*

Teorema 1.10. *Si $f \in \mathcal{C}^1([a, b])$ y consideremos el conjunto de nodos de Chebyshev. Entonces los polinomios interpoladores convergen uniformemente a f en $[a, b]$.*

Nota. Observar el hecho de que si una de las condiciones falla en un solo punto hace que no se cumpla, como ocurre en el Contraejemplo de Bernstein.

1.5. Ventajas y desventajas de los métodos.

El método de Newton es un mecanismo dinámico. Permite añadir un nodo a la tabla y utilizar la información ya calculada, algo que no es posible con el método de Lagrange debido a que la variación del número de nodos obliga a recalcular todos los polinomios fundamentales de Lagrange. Numéricamente hablando, el método de Newton para el cálculo del polinomio de interpolación mediante diferencias divididas junto con el algoritmo de Horner es más eficiente y rápido computacionalmente. Sin embargo, el método de Lagrange se simplifica considerablemente cuando las ordenadas asociadas a los nodos son nulas (como se vio en el Ejemplo 5). Además tiene un especial interés teórico en numerosos procesos numéricos, como por ejemplo, en la construcción de fórmulas de cuadratura.

En efecto, si deseamos aproximar la integral $I = \int_a^b f(x)dx$, siendo $f \in \mathcal{C}([a, b])$ y p denota al polinomio que interpola a f en $(n+1)$ nodos $x_i \in [a, b]$, $i = 0, \dots, n$. Como $p \approx f$, parece razonable proceder reemplazando en el integrando de I la función f por el polinomio de interpolación p en su forma de Lagrange, obteniendo así **fórmulas de cuadratura de tipo interpolatorio**:

$$I \approx \int_a^b \sum_{i=0}^n f(x_i) \cdot l_i(x) dx = \sum_{i=0}^n f(x_i) \cdot \underbrace{\int_a^b l_i(x) dx}_{\lambda_i} = \sum_{i=0}^n \lambda_i \cdot f(x_i).$$

Los polinomios interpoladores son funciones aproximantes que tienen un carácter global dado que tienen una misma expresión en todo el intervalo $[a, b]$ en el que se considera el problema de interpolación. No obstante, ya sabemos que en general la interpolación polinómica puede carecer de buenas propiedades de convergencia, y de hecho, al aumentar el número de nodos de interpolación, se asume el riesgo de que el polinomio presente altas oscilaciones.

Una alternativa más flexible a la interpolación polinómica clásica consiste en considerar una partición Δ del intervalo $[a, b]$ e interpolar en cada subintervalo de la partición mediante polinomios de grado bajo, exigiendo naturalmente, que el aproximante a trozos obtenido tenga suficiente regularidad en los puntos de la partición. Surgen así las denominadas funciones spline (polinómicas), que de modo general son funciones polinómicas de grado k a trozos en una determinada partición del intervalo $[a, b]$, con la condición de que sus derivadas sucesivas sean continuas hasta cierto orden m , $\forall 0 \leq m < k$ (claramente si se exige $m = k$ se obtiene un polinomio definido globalmente en todo intervalo). Por simplicidad en la exposición, asumiremos en nuestro desarrollo que $m = k - 1$. La teoría de funciones spline ha encontrado interesantes aplicaciones, como por ejemplo en el diseño geométrico asistido por ordenador o en la regularización de soluciones

numéricas a Ecuaciones en Derivadas Parciales obtenidas a través del Método de Elementos Finitos.

Funciones Spline

En este capítulo se introducirá el concepto de *Función Spline*. Se estudiará cómo construir tales funciones y sus propiedades más relevantes en relación con la interpolación.

2.1. Funciones Spline.

Comenzamos dando la definición formal de *función spline*:

Definición 2.1. Sean $n + 1$ puntos distintos $t_0 < t_1 < \dots < t_n$ que componen una partición Δ y $k \in \mathbb{N}$ fijo. Una **función spline de grado k** con nodos en t_0, t_1, \dots, t_n es un función S que satisface las siguientes condiciones:

- i) $S \in \mathbb{P}_k$ en cada intervalo $[t_{i-1}, t_i]$.
- ii) $S \in \mathcal{C}^{k-1}([t_0, t_n])$.

Por tanto, S es un polinomio de grado a lo sumo k a trozos en $[t_0, t_n]$ continuo y que tiene derivadas continuas hasta orden $k - 1$.

Denotaremos por \mathcal{S}^k al espacio de funciones spline de grado k definidas en el intervalo $[t_0, t_n]$.

Los splines de grado 0 son funciones constantes a trozos, por lo que vienen dados explícitamente por

$$S(x) = \begin{cases} S_0(x) = c_0, & x \in [t_0, t_1), \\ S_1(x) = c_1, & x \in [t_1, t_2), \\ \vdots & \vdots \\ S_{n-1}(x) = c_{n-1}, & x \in [t_{n-1}, t_n]. \end{cases}$$

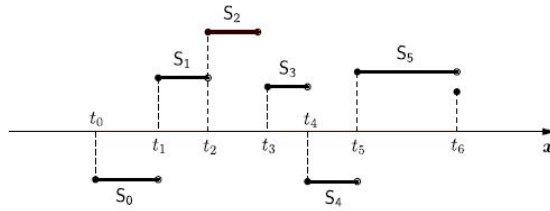


Figura 2.1. Función Spline de grado 0.

Una función spline de grado 1 se puede expresar de la forma

$$S(x) = \begin{cases} S_0(x) = a_0x + b_0, & x \in [t_0, t_1), \\ S_1(x) = a_1x + b_1, & x \in [t_1, t_2), \\ \vdots & \vdots \\ S_{n-1}(x) = a_{n-1}x + b_{n-1}, & x \in [t_{n-1}, t_n], \end{cases}$$

donde los nodos t_i y los coeficientes a_i, b_i son conocidos. La función spline se puede definir en toda la recta real. Por conveniencia podemos utilizar la expresión S_0 sobre el intervalo $(-\infty, t_1)$ y la expresión S_{n-1} sobre el intervalo $[t_{n-1}, \infty)$. La función S es continua, por lo que los coeficientes a_i, b_i no son realmente arbitrarios, dado que debe cumplirse que $\lim_{x \rightarrow t_i^-} S_{i-1} = S_i(t_i), \forall i = 1, \dots, n - 1$.

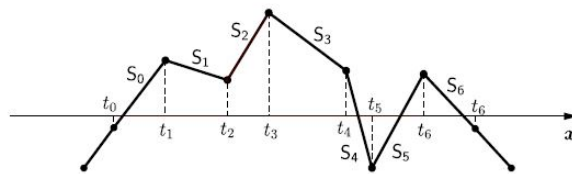


Figura 2.2. Función Spline de grado 1.

Plantaremos con fines ilustrativos el problema de interpolación de una función f definida en el intervalo $[a, b]$ por medio de una *función spline lineal* ($k = 1$). Para resolverlo, en primer lugar, se define una partición Δ del intervalo $[a, b]$ de tal forma que $a = t_0 < t_1 < \dots < t_n = b$. A partir de las $n + 1$

condiciones de interpolación $S(t_i) = f(t_i), \forall i = 0, \dots, n$, y de regularidad en los nodos interiores deducimos que

$$S(x) = f(t_i) + (f(t_{i+1}) - f(t_i)) \frac{x - t_i}{t_{i+1} - t_i}, \forall i = 0, \dots, n-1 \text{ con } x \in [t_i, t_{i+1}]. \quad (2.1)$$

Por otra parte, además la expresión (2.1) es única, luego $S|_{[t_i, t_{i+1}]}$ coincide con el polinomio interpolador de f de grado 1 en los nodos t_i y $t_{i+1}, \forall i = 0, \dots, n-1$. Notemos además que, en este caso, al aumentar el número de nodos no varía el grado de los interpolantes, que es uno. El resultado es una gráfica en forma de línea poligonal como en la Figura 2.2.

Debido a su frecuente uso en diversas áreas del Análisis Numérico y en problemas aplicados, en la siguiente sección abordaremos el problema de interpolación basado en *funciones spline cúbicas* ($k = 3$).

2.2. Funciones Spline cúbicas interpolantes.

En esta sección presentaremos la construcción de una función spline cúbica interpolante, así como la demostración de su propiedad más relevante. Supongamos que partimos de una tabla de valores dada:

$$\begin{array}{c|c|c|c|c|c} t & t_0 & t_1 & t_2 & \cdots & t_n \\ \hline y & y_0 & y_1 & y_2 & \cdots & y_n \end{array}$$

y que se desea construir un spline cúbico S definido en $[t_0, t_n]$ que interpole dicha tabla. Sobre cada intervalo $[t_0, t_1], [t_1, t_2], \dots, [t_{n-1}, t_n]$, S está definida como un polinomio cúbico diferente:

$$S(x) = \begin{cases} S_0(x), & x \in [t_0, t_1), \\ S_1(x), & x \in [t_1, t_2), \\ \vdots & \vdots \\ S_{n-1}(x), & x \in [t_{n-1}, t_n]. \end{cases} \quad (2.2)$$

Al ser S continua, los polinomios S_{i-1} y S_i interpolan el mismo valor en el punto t_i y por ello:

$$S_{i-1}(t_i) = y_i = S_i(t_i), \quad \forall 1 < i < n-1. \quad (2.3)$$

Además S' y S'' también deben ser continuas. ¿Se podrá definir bajo estas condiciones de interpolación y de regularidad el spline cúbico interpolante S de manera única?

Cada polinomio S_i que compone a la función spline cúbica posee 4 coeficientes por lo que hay un total de $4n$ coeficientes a determinar. Las condiciones

de interpolación y de continuidad de S en los nodos interiores nos proporcionan $2(n-1)$, condiciones como hemos visto en (2.3), y una por cada uno de los nodos extremos. Así pues tenemos un total de $2n$ condiciones. La continuidad de S' proporciona una condición en cada nodo interior, $S'_{i-1}(t_i) = S'_i(t_i)$ y de igual manera, la continuidad de S'' nos proporciona otras $(n-1)$ condiciones. Por tanto, tenemos $4n-2$ condiciones para determinar los $4n$ coeficientes, quedando así dos *grados de libertad*, que nos permiten fijar 2 condiciones adicionales para nuestro beneficio.

La necesidad de imponer dos condiciones adicionales para la determinación de S , da lugar a ciertos splines cúbicos interpolantes particulares:

1. *natural*: $S''(t_0) = S''(t_n) = 0$.
2. *periódico*: $S'(t_0) = S'(t_n)$ y $S''(t_0) = S''(t_n)$, cuando $y_0 = y_n$.
3. *completo*: $S'(t_0) = f'(t_0)$ y $S'(t_n) = f'(t_n)$.

Nos centramos ahora en la obtención de S . Para ello, procederemos partiendo de sus *momentos* $\{S''(t_i) := M_i\}_{i=0}^n$. Obviamente, los momentos M_i existen, $\forall 0 \leq i \leq n$ y debe cumplirse que $\lim_{x \rightarrow t_j^-} S''_{i-1}(x) = M_i = \lim_{x \rightarrow t_j^+} S''_i(x)$, $\forall 0 \leq j \leq n$, por la continuidad de S'' en los nodos interiores. Dado que S_i es un polinomio cúbico en $[t_i, t_{i+1})$, se deduce que S''_i es una función lineal en dicho intervalo tal que $S''_i(t_i) = M_i$ y $S''_i(t_{i+1}) = M_{i+1}$, S''_i que se corresponde con la segmento que une (t_i, M_i) con (t_{i+1}, M_{i+1}) :

$$S''_i(x) = \frac{M_i}{h_i}(t_{i+1} - x) + \frac{M_{i+1}}{h_i}(x - t_i), \quad x \in [t_i, t_{i+1}),$$

donde $h_i := t_{i+1} - t_i$. Si integramos dos veces esta expresión obtendremos:

$$S_i(x) = \frac{M_i}{6h_i}(t_{i+1} - x)^3 + \frac{M_{i+1}}{6h_i}(x - t_i)^3 + C_i(x - t_i) + D_i(t_{i+1} - x), \quad (2.4)$$

donde C_i y D_i son constantes. Imponiendo las condiciones de interpolación $S_i(t_i) = y_i$ y $S_i(t_{i+1}) = y_{i+1}$ en (2.4) determinamos C_i y D_i , $\forall 0 \leq i \leq n-1$:

$$\begin{aligned} S_i(x) &= \frac{M_i}{6h_i}(t_{i+1} - x)^3 + \frac{M_{i+1}}{6h_i}(x - t_i)^3 \\ &+ \left(\frac{y_{i+1}}{h_i} - \frac{M_{i+1}h_i}{6} \right) (x - t_i) + \left(\frac{y_i}{h_i} - \frac{M_i h_i}{6} \right) (t_{i+1} - x). \end{aligned} \quad (2.5)$$

El problema estaría por tanto resuelto, a falta de calcular los momentos $\{M_i\}_{i=0}^n$. Para hallar los valores M_1, \dots, M_{n-1} recurrimos a las condiciones de continuidad impuestas sobre S' , $S'_{i-1}(t_i) = S'_i(t_i)$. Derivando la ecuación (2.5) obtenemos

$$S'_i(x) = -\frac{M_i}{2h_i}(t_{i+1} - x)^2 + \frac{M_{i+1}}{2h_i}(x - t_i)^2 + \frac{y_{i+1}}{h_i} - \frac{M_{i+1}h_i}{6} - \frac{y_i}{h_i} + \frac{M_i h_i}{6}.$$

Sustituyendo $x = t_i$ y simplificando tenemos que

$$S'_i(t_i) = -\frac{h_i}{3}M_i - \frac{h_i}{6}M_{i+1} + \frac{y_{i+1}}{h_i} - \frac{y_i}{h_i}. \quad (2.6)$$

Procediendo de la misma forma, si utilizamos la ecuación (2.5) para obtener S'_{i-1} , llegamos a que:

$$S'_{i-1}(t_i) = \frac{h_{i-1}}{3}M_i + \frac{h_{i-1}}{6}M_{i-1} + \frac{y_i}{h_{i-1}} - \frac{y_{i-1}}{h_{i-1}}. \quad (2.7)$$

Igualando las ecuaciones (2.6) y (2.7) podemos escribir

$$M_{i-1}h_{i-1} + 2(h_i + h_{i-1})M_i + M_{i+1}h_i = \frac{6}{h_i}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1}). \quad (2.8)$$

Esto genera un sistema de $n - 1$ ecuaciones lineales con $n - 1$ incógnitas, M_1, \dots, M_{n-1} . Las cantidades M_0 y M_n pueden elegirse de manera arbitraria (dos grados de libertad), o recurrir a los diferentes tipos de funciones spline mencionados anteriormente para determinarlas.

Supongamos que nuestra elección es construir un spline cúbico natural ($S''(t_0) = S''(t_n) = 0$). El sistema lineal (2.8) para $1 \leq i \leq n - 1$ con $M_0 = 0$ y $M_n = 0$ puede expresarse en la forma

$$\begin{pmatrix} u_1 & h_1 & 0 & 0 & \cdots & 0 \\ h_1 & u_2 & h_2 & 0 & \cdots & 0 \\ 0 & h_2 & u_3 & h_3 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & h_{n-3} & u_{n-2} & h_{n-2} \\ 0 & 0 & 0 & \cdots & h_{n-2} & u_{n-1} \end{pmatrix} \cdot \begin{pmatrix} M_1 \\ M_2 \\ M_3 \\ \vdots \\ M_{n-2} \\ M_{n-1} \end{pmatrix} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_{n-2} \\ \alpha_{n-1} \end{pmatrix}, \quad (2.9)$$

donde

$$u_i = 2(h_i + h_{i-1}) \quad \text{y} \quad \alpha_i = \frac{6}{h_i}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1}), \quad \forall i = 1, \dots, n - 1.$$

Recordemos que los nodos están ordenados ($t_i < t_{i+1}$), por tanto $h_i = t_{i+1} - t_i > 0$, $\forall i = 0, \dots, n - 1$. Vemos pues que las cantidades α_i están bien definidas y que la matriz asociada al sistema (2.9) es simétrica, tridiagonal y diagonal dominante, concluyendo por tanto del Teorema de Gershgorin (véase [1, Teorema 9.1]) que es no singular y que el sistema tiene por tanto solución

única.

Una vez hemos determinado los coeficientes M_0, M_1, \dots, M_n , puede calcularse cualquier valor $S(x)$ de la función spline cúbica (2.2) a partir de la ecuación (2.5), determinando previamente el índice i para el cual $x \in [t_i, t_{i+1})$. Recordemos que por conveniencia podemos utilizar la expresión S_0 sobre el intervalo $(-\infty, t_1)$ y la expresión S_{n-1} sobre el intervalo $[t_{n-1}, \infty)$, definiendo de este modo a la función S en todo \mathbb{R} .

Ejemplo 6. Hallar el spline cúbico natural, considerando la siguiente tabla de datos.

| | | | | | | | | | | |
|-----|------|------|-----|---|---|----|-------|-------|------|----|
| t | -2 | -0,5 | 0,5 | 1 | 2 | 4 | 6 | 7 | 8,8 | 10 |
| y | 1,25 | 0,6 | 1 | 2 | 0 | -2 | -5,25 | -1,75 | 0,75 | 2 |

Solución. Esto genera un sistema de 8 ecuaciones lineales con 8 incógnitas, M_1, \dots, M_8 . Las cantidades $M_0 = 0 = M_9$ no son incógnitas, por ser el spline cúbico natural. Luego estamos ante un sistema del tipo $A \cdot X = B$ como el presentado a continuación:

$$\begin{pmatrix} u_1 & h_1 & 0 & 0 & \cdots & 0 \\ h_1 & u_2 & h_2 & 0 & \cdots & 0 \\ 0 & h_2 & u_3 & h_3 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & h_6 & u_7 & h_7 \\ 0 & 0 & 0 & \cdots & h_7 & u_8 \end{pmatrix} \cdot \begin{pmatrix} M_1 \\ M_2 \\ M_3 \\ \vdots \\ M_7 \\ M_8 \end{pmatrix} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_7 \\ \alpha_8 \end{pmatrix},$$

donde

$$u_i = 2(h_i + h_{i-1}) \quad y \quad \alpha_i = \frac{6}{h_i}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1}), \quad \forall i = 1, \dots, 8.$$

Procedemos a calcular los valores correspondientes a h_i, u_i y α_i a partir de los datos de la tabla como se indicó en el desarrollo anterior. Resultando así el siguiente sistema:

$$\begin{pmatrix} 5,0 & 1,0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1,0 & 3,0 & 0,5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0,5 & 3,0 & 1,0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1,0 & 6,0 & 2,0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2,0 & 8,0 & 2,0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2,0 & 6,0 & 1,0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,0 & 5,6 & 1,8 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1,8 & 6,0 \end{pmatrix} \cdot \begin{pmatrix} M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \\ M_6 \\ M_7 \\ M_8 \end{pmatrix} \approx \begin{pmatrix} 5,0 \\ 9,6 \\ -24,0 \\ 6,0 \\ -3,75 \\ 30,75 \\ 12,67 \\ -2,08 \end{pmatrix}.$$

Para resolverlo este sistema aplicaremos el **algoritmo de Thomas**, es decir, nuestra matriz A admite una descomposición LU mediante la descomposición de Crout (que se ha obtenido comparando elemento a elemento las entradas de A y las de LU , donde L es una matriz triangular inferior y U triangular superior con 1 en las entradas de la diagonal principal).

$$\begin{pmatrix} 5,0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1,0 & 2,8 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0,5 & 2,91 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1,0 & 5,66 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2,0 & 7,29 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2,0 & 5,45 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,0 & 5,42 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1,8 & 5,4 \end{pmatrix} \cdot \begin{pmatrix} 1,0 & 0,2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1,0 & 0,18 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1,0 & 0,34 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1,0 & 0,35 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1,0 & 0,27 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,0 & 0,18 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1,0 & 0,33 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1,0 \end{pmatrix}.$$

Obteniendo así un sistema del tipo $L \cdot U \cdot X = B$. Que se resolverá en dos pasos:
1 Resolver $L \cdot Y = B$, mediante sustitución hacia adelante.

$$\begin{pmatrix} 5,0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1,0 & 2,8 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0,5 & 2,91 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1,0 & 5,66 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2,0 & 7,29 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2,0 & 5,45 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,0 & 5,42 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1,8 & 5,4 \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \end{pmatrix} \approx \begin{pmatrix} 5,0 \\ 9,6 \\ -24,0 \\ 6,0 \\ -3,75 \\ 30,75 \\ 12,67 \\ -2,08 \end{pmatrix},$$

luego $Y^T = (1,0 \quad 3,07 \quad -8,77 \quad 2,61 \quad -1,23 \quad 6,09 \quad 1,21 \quad -0,79)$.

2 Resolver $U \cdot X = Y$, mediante sustitución hacia atrás.

$$\begin{pmatrix} 1,0 & 0,2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1,0 & 0,18 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1,0 & 0,34 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1,0 & 0,35 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1,0 & 0,27 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,0 & 0,18 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1,0 & 0,33 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1,0 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{pmatrix} \approx \begin{pmatrix} 1,0 \\ 3,07 \\ -8,77 \\ 2,61 \\ -1,23 \\ 6,09 \\ 1,21 \\ -0,79 \end{pmatrix},$$

la solución es $X^T = (0,03 \quad 4,87 \quad -9,99 \quad 3,59 \quad -2,80 \quad 5,82 \quad 1,47 \quad -0,79)$.

Ya se han determinado los momentos de nuestro spline cúbico, es decir, los coeficientes M_0, M_1, \dots, M_9 , sólo falta para resolver el ejercicio hacer uso de

la ecuación (2.5), para hallar la expresión (2.2) del spline buscado. Obtenemos así el siguiente spline cúbico:

$$S(x) = \begin{cases} S_0(x) = 0,003x^3 + 0,02x^2 - 0,4x + 0,395, & x \in [2, -0,5), \\ S_1(x) = 0,807x^3 + 1,225x^2 + 0,198x + 0,494, & x \in [-0,5, 0,5), \\ S_2(x) = -4,953x^3 + 9,865x^2 - 4,129x + 1,217, & x \in [0,5, 1), \\ S_3(x) = 2,263x^3 - 11,785x^2 + 17,512x - 5,99, & x \in [1, 2), \\ S_4(x) = -0,532x^3 + 4,99x^2 - 16,03x + 16,36, & x \in [2, 4), \\ S_5(x) = 0,718x^3 - 10,02x^2 + 43,982x - 63,58, & x \in [4, 6), \\ S_6(x) = -0,725x^3 + 15,96x^2 - 111,905x + 248,22, & x \in [6, 7), \\ S_7(x) = -0,209x^3 + 5,129x^2 - 40,307x + 100,834, & x \in [7, 8,8), \\ S_8(x) = 0,110x^3 - 3,292x^2 + 33,800x - 116,559, & x \in [8,8, 10]. \end{cases}$$

Tras concluir la construcción de las funciones spline cúbicas naturales interpolantes, recordemos que el objetivo de considerar funciones spline interpolantes era con el propósito de minimizar las grandes oscilaciones que producen los polinomios ordinarios cuando son de grado elevado. La propiedad que enunciamos y demostramos a continuación, junto con la nota que le acompaña, justifica tal consideración. Este resultado nos viene a decir que es precisamente un spline cúbico natural el que genera la función interpolante en $[t_0, t_1]$ menos oscilatoria posible.

Teorema 2.2. *Sea $f \in \mathcal{C}^2([a, b])$ y tomemos $a = t_0 < t_1 < \dots < t_n = b$. Si S es el spline cúbico natural que interpola a f en los nodos $t_i, \forall 0 \leq i \leq n$, entonces:*

$$\int_a^b [S''(x)]^2 dx \leq \int_a^b [f''(x)]^2 dx.$$

Demostración. Sea $g \equiv f - S$, luego $g(t_i) = 0, \forall 0 \leq i \leq n$ y $f'' \equiv g'' + S''$. Entonces,

$$\int_a^b (f'')^2 dx = \int_a^b (S'')^2 dx + \int_a^b (g'')^2 dx + 2 \int_a^b S'' g'' dx.$$

El teorema queda demostrado si probamos que $\int_a^b S'' g'' dx \geq 0$, en particular veremos que dicha integral es nula. Para ello realizamos el cálculo utilizando la integración por partes, usando la condición $S'''(t_0) = S'''(t_n) = 0$, y teniendo en cuenta que S''' es la función constante c_i en el intervalo $[t_{i-1}, t_i), \forall 1 \leq i \leq n$.

$$\begin{aligned}
 \int_a^b S'' g'' dx &= \sum_{i=1}^n \int_{t_{i-1}}^{t_i} S'' g'' dx \\
 &= \sum_{i=1}^n \left[(S'' g')(t_i) - (S'' g')(t_{i-1}) - \int_{t_{i-1}}^{t_i} S''' g' dx \right] \\
 &= - \sum_{i=1}^n \int_{t_{i-1}}^{t_i} S''' g' dx = - \sum_{i=1}^n c_i \int_{t_{i-1}}^{t_i} g' dx \\
 &= - \sum_{i=1}^n c_i [g(t_i) - g(t_{i-1})] = 0. \quad \square
 \end{aligned}$$

Nota. La curvatura de la curva descrita por la ecuación $y = f(x)$ viene dada por $|f''(x)| [1 + (f'(x))^2]^{-3/2}$. Suponiendo que el término no lineal se aproxime a 1, podemos asociar $|f''(x)|$ como una aproximación de la curvatura de dicha curva. Teniendo en cuenta esto y el Teorema 2.2 concluimos que si S es el spline cúbico natural que interpola a $f \in \mathcal{C}^2([a, b])$ en los nodos $t_i, \forall 0 \leq i \leq n$, entonces S es, de todas las funciones en $\mathcal{C}^2([a, b])$ que interpolan a f en estos nodos, la curva aproximadamente *más suave* posible, es decir, con aproximadamente la curvatura mínima sobre cada intervalo.

A continuación se muestra una gráfica, donde se compara la función spline cúbica generada en el Ejercicio 6 frente al polinomio interpolador construido para la misma tabla de datos.

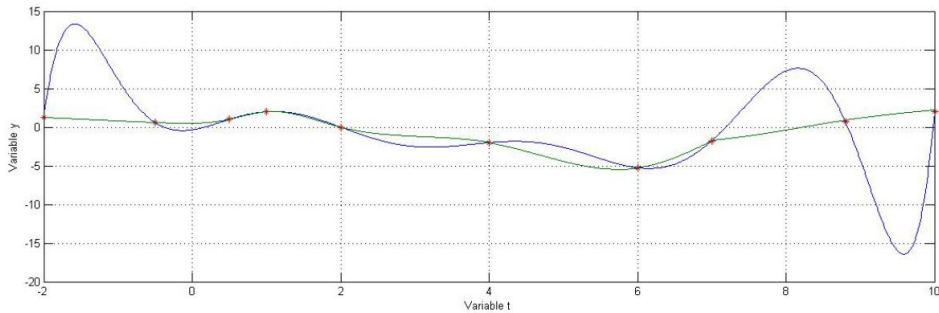


Figura 2.3. Función spline cúbica vs polinomio interpolador.

En la gráfica se ha representado, en color rojo los puntos de interpolación, en trazo azul el polinomio interpolador, que en este caso se ha obtenido haciendo uso de la fórmula de Lagrange, y en trazo verde el spline cúbico interpolador obtenido anteriormente. Se puede apreciar claramente como efectivamente la

función spline (verde) presenta menores oscilaciones que el polinomio interpolador (azul) minimizando la curvatura como indica el Teorema (2.2). La gráfica se ha obtenido haciendo uso de Matlab, cuyo código se adjunta en el Apéndice.

2.3. Funciones Spline de grado superior.

Para concluir este capítulo, a continuación enunciaremos algunos resultados teóricos sobre funciones spline de grado superior y sobre funciones spline naturales.

Una función que conviene utilizar es la conocida como *función de potencias truncada*, denotada por $(x - t_i)_+^n \in \mathcal{C}^{(n-1)}(\mathbb{R})$ y definida como:

$$(x - t_i)_+^n = \begin{cases} (x - t_i)^n & \text{si } x \geq t_i, \\ 0 & \text{si } x < t_i. \end{cases}$$

Recordemos que \mathcal{S}^k denota al espacio de funciones spline de grado k definidas en el intervalo $[t_0, t_n]$.

Proposición 2.3. *La dimensión de \mathcal{S}^k es a lo sumo $k + n$.*

Demostración. La idea de la demostración es bastante intuitiva, dado que un spline $S(x) \in \mathcal{S}^k$ queda determinado por $n \cdot (k + 1)$ coeficientes, mientras que las $(n - 1) \cdot k$ condiciones de regularidad en los nodos interiores reducen el número de parámetros libres a $n + k$, para fijar las condiciones de interpolación y las restantes a elegir según conveniencia. Para demostrar la proposición procederemos encontrando un conjunto de $k + n$ funciones que generan el espacio \mathcal{S}^k . En efecto, si $S \in \mathcal{S}^k$, entonces se puede expresar como:

$$S(x) = \sum_{j=0}^k a_j x^j + \sum_{i=1}^{n-1} b_i (x - t_i)_+^k, \quad \text{con } a_j, b_i \in \mathbb{R}. \quad (2.10)$$

Para comprobar la igualdad dada en la ecuación (2.10), comenzamos estudiando qué ocurre en el intervalo $[t_0, t_1)$, en el que todas las funciones de potencias truncadas valen 0. En $[t_0, t_1)$, $S(x)$ es un polinomio de grado k , al que denotamos por p_0 . Entonces, tenemos que $p_0 = \sum_{j=0}^k a_j x^j$, expresión que determina los coeficientes a_j . En $[t_1, t_2)$, $S(x)$ es otro polinomio p_1 de grado k . Teniendo en cuenta las condiciones de regularidad en t_1 , tenemos:

$$(p_1 - p_0)^{(r)}(t_1) = 0, \quad 0 \leq r \leq k - 1.$$

Dado que $p_0 - p_1 \in \mathbb{P}_k$, podemos concluir que $(p_0 - p_1)(x) = b_1(x - t_1)^k$ para un cierto b_1 . Esto nos permite escribir

$$S(x) = \sum_{j=0}^k a_j x^j + b_1(x - t_1)_+^k, \quad t_0 \leq x \leq t_2.$$

Este mismo argumento se repite de manera iterativa en todos los nodos interiores t_2, t_3, \dots, t_{n-1} , justificando así la expresión (2.10).

□

De la demostración de la Proposición 2.3 no solo se deduce su enunciado sino que también vemos una expresión alternativa para expresar una función spline de orden superior empleando funciones de potencias truncadas. Se trata por tanto de una demostración constructiva. En el siguiente capítulo demostraremos que la dimensión de \mathcal{S}^k es exactamente $n + k$.

Las funciones spline naturales de orden superior sólo tienen sentido para grados impares, debido a cómo se eligen las condiciones. En efecto, sea $S \in \mathcal{S}^k$. Tras fijar las condiciones de regularidad y de interpolación nos quedan $k - 1$ parámetros libres. Para hacer posible imponer el mismo número de condiciones en t_0 y en t_n , $k - 1$ tiene que ser par, y por tanto k impar. Asumimos pues que $S \in \mathbb{P}_k$ con $k = 2m + 1$, luego tenemos $2m$ parámetros libres distribuidos de tal manera que impongamos m condiciones en t_0 y otras m en t_n . ¿Cuáles son esas condiciones en t_0 y t_n para definir un spline natural de orden superior? Estas van a ser

$$S^{(m+r)}(t_0) = S^{(m+r)}(t_n) = 0, \quad \text{con } r = 1, \dots, m. \quad (2.11)$$

Cuando $m = 1$ tenemos los splines cúbicos naturales que acabamos de estudiar. Así pues, bajo estas condiciones se puede demostrar de manera análoga al Teorema 2.2 el siguiente resultado general:

Teorema 2.4. *Sean $m \leq n$ y $f \in \mathcal{C}^{m+1}([a, b])$. Sea S el spline natural de grado $2m + 1$ que interpola a f en los nodos $a = t_0 < t_1 < \dots < t_n = b$. Entonces:*

$$\int_a^b [S^{(m+1)}(x)]^2 dx \leq \int_a^b [f^{(m+1)}(x)]^2 dx.$$

Para finalizar esta sección veremos brevemente dos resultados más para splines naturales de grado superior. Para afrontar la teoría en general es conveniente proceder de una manera ligeramente diferente. Al igual que antes, se parte de un conjunto de nodos: $t_0 < t_1 < \dots < t_n$. Un spline natural de grado $2m + 1$ consistirá en una función $S \in \mathcal{C}^{2m}(\mathbb{R})$ que se reduce a un polinomio de grado $\leq 2m + 1$ en cada intervalo $[t_0, t_1), [t_1, t_2), \dots, [t_{n-1}, t_n]$ cumpliendo las condiciones definidas en (2.11) y a un polinomio de grado a lo sumo m en los intervalos $(-\infty, t_0)$ y (t_n, ∞) . Con esta definición, un spline cúbico natural se reduce a polinomios lineales en $(-\infty, t_0)$ y (t_n, ∞) y a un polinomio cúbico en

cada intervalo $[t_0, t_1), [t_1, t_2), \dots, [t_{n-1}, t_n]$ tal que $S''(t_0) = S''(t_n) = 0$. El subespacio vectorial de las funciones spline naturales de grado $2m + 1$ se denotará por \mathcal{N}_n^{2m+1} .

Teorema 2.5. *Cada función en \mathcal{N}_n^{2m+1} se puede expresar como:*

$$S(x) = \sum_{i=0}^m a_i x^i + \sum_{i=0}^n b_i (x - t_i)_+^{2m+1},$$

con $\sum_{i=0}^n b_i (t_i)^j = 0, \quad \forall 0 \leq j \leq m$.

Teorema 2.6. *Sea $t_0 < t_1 < \dots < t_n$ un conjunto de nodos y tomemos $0 \leq m \leq n$. Existe un único spline natural de grado $2m + 1$ que ajusta los valores dados en los nodos.*

Pueden verse las demostraciones de los Teoremas 2.4 - 2.6 en [2, págs. 333-336].

Funciones B-Spline

Este capítulo se dedicará al estudio de las funciones B-spline, que serán funciones spline con notables propiedades. En concreto, serán un caso particular de funciones no negativas que se anulan en toda la recta real excepto en unos pocos intervalos $[t_i, t_{i+1}]$ contiguos. Las funciones B-spline nos proporcionarán una base para construir fácilmente funciones spline generales. Su estudio proporciona además distinguidas aportaciones teóricas en Matemática Aplicada.

3.1. Funciones B-Spline.

En la práctica las funciones spline se construyen a partir de un sistema de nodos dados tales que $t_0 < t_1 < \dots < t_n$, pero en lo que respecta al desarrollo teórico, es más sencillo considerar que los nodos t_i , estén ordenados sobre toda la recta real, es decir, que formen un conjunto infinito que se extiende desde $-\infty$ hasta $+\infty$:

$$\dots < t_{-1} < t_0 < t_1 < \dots < t_{n-1} < t_n < t_{n+1} < \dots$$

Partiendo de esta premisa, a continuación vamos a ver cómo se definen formalmente los B-splines de grado 0, que denotaremos por B_i^0 :

$$B_i^0 = \begin{cases} 1, & \text{si } t_i \leq x < t_{i+1}, \\ 0, & \text{si } x < t_i \text{ ó } x \geq t_{i+1}. \end{cases}$$

Los B-splines de grado 0 constituyen una sucesión infinita, $\{B_i^0 : i \in \mathbb{Z}\}$. Algunas de sus propiedades más destacables que se deducen directamente de su definición son:

- (i) El soporte de B_i^0 , definido como el conjunto de valores x donde $B_i(x)^0 \neq 0$, es el intervalo $[t_i, t_{i+1})$.

- (ii) $B_i^0(x) \geq 0, \forall i \in \mathbb{Z} \text{ y } \forall x \in \mathbb{R}.$
- (iii) B_i^0 es continua por la derecha en toda la recta real.
- (iv) $\sum_{i=-\infty}^{\infty} B_i^0(x) \equiv 1, \forall x \in \mathbb{R}.$

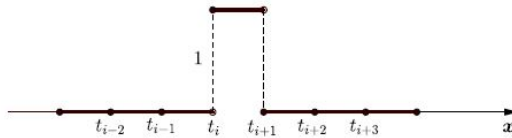


Figura 3.1. B-Spline de grado 0, B_i^0

Resulta evidente que las funciones B-splines B_i^0 constituyen una base para todos los splines de grado 0 definidos sobre la sucesión de nodos dados, de manera que sean continuos por la derecha. Por tanto, $S(x) = \sum_{i=-\infty}^{\infty} c_i B_i^0(x), \forall i \in \mathbb{Z}.$ De esta manera tenemos una base de Schauder: cada vector en el espacio tiene un representación única en términos de una serie doblemente infinita.

Las funciones B_i^0 son el punto de partida para una definición recursiva de todas las funciones B-spline de orden superior. Consideremos para ello la relación.

$$B_i^k(x) = \left(\frac{x - t_i}{t_{i+k} - t_i} \right) B_i^{k-1}(x) + \left(\frac{t_{i+k+1} - x}{t_{i+k+1} - t_{i+1}} \right) B_{i+1}^{k-1}(x), \quad k \geq 1, \quad (3.1)$$

que relaciona B-splines de orden k con B-splines de órdenes $k - 1$. Considerando las funciones lineales auxiliares

$$V_i^k(x) = \frac{x - t_i}{t_{i+k} - t_i}, \quad (3.2)$$

podemos escribir la relación de la siguiente manera:

$$B_i^k(x) = V_i^k B_i^{k-1} + (1 - V_{i+1}^k) B_{i+1}^{k-1}. \quad (3.3)$$

Dado que B_i^0 es un polinomio a trozos de grado 0, y que V_i^k es lineal, B_i^1 es un polinomio a trozos de grado ≤ 1 . Análogamente, de forma general tenemos que B_i^k es un polinomio a trozos de grado $\leq k$.

Utilizando la ecuación (3.1) se puede dar la expresión formal para B_i^1 :

$$\begin{aligned}
 B_i^1(x) &= \left(\frac{x - t_i}{t_{i+1} - t_i} \right) B_i^0(x) + \left(\frac{t_{i+2} - x}{t_{i+2} - t_{i+1}} \right) B_{i+1}^0(x) \\
 &= \begin{cases} 0, & \text{si } x < t_i \text{ ó } x \geq t_{i+2}, \\ \frac{x - t_i}{t_{i+1} - t_i}, & \text{si } t_i \leq x < t_{i+1}, \\ \frac{t_{i+2} - x}{t_{i+2} - t_{i+1}}, & \text{si } t_{i+1} \leq x < t_{i+2}. \end{cases}
 \end{aligned}$$

Algunas de sus propiedades más destacables son:

- (i) El soporte de B_i^1 , es (t_i, t_{i+2}) .
- (ii) $B_i^1(x) \geq 0, \forall i \in \mathbb{Z}$ y $\forall x \in \mathbb{R}$.
- (iii) B_i^1 es continua y diferenciable para todo punto excepto en t_i, t_{i+1} y t_{i+2} .
- (iv) $\sum_{i=-\infty}^{\infty} B_i^1(x) \equiv 1, \forall x \in \mathbb{R}$.

Esta última igualdad se puede comprobar de la siguiente manera:

- Si fijamos $x \in (t_i, t_{i+1})$, entonces $B_{i-1}^1(x) \neq 0, B_i^1(x) \neq 0$ y $B_j^1(x) = 0, \forall j \neq \{i - 1, i\}$. Por tanto,

$$\sum_{i \in \mathbb{Z}} B_i^1(x) = B_{i-1}^1(x) + B_i^1(x) = \frac{t_{i+1} - x}{t_{i+1} - t_i} + \frac{x - t_i}{t_{i+1} - t_i} = 1.$$

- Si ahora $x \in (t_{i+1}, t_{i+2})$, entonces $B_i^1(x) \neq 0, B_{i+1}^1(x) \neq 0$ y $B_j^1(x) = 0, \forall j \neq \{i, i + 1\}$. Por tanto,

$$\sum_{i \in \mathbb{Z}} B_i^1(x) = B_i^1(x) + B_{i+1}^1(x) = \frac{t_{i+2} - x}{t_{i+2} - t_{i+1}} + \frac{x - t_{i+1}}{t_{i+2} - t_{i+1}} = 1.$$

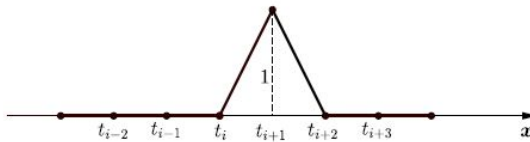


Figura 3.2. B-Spline de grado 1. B_i^1

En la siguiente sección generalizamos estas propiedades para funciones B-splines de orden superior, haciendo uso especialmente de las relaciones (3.1) - (3.3).

3.2. Propiedades de las funciones B-Spline.

A continuación se demostrarán una serie de lemas donde recogeremos las propiedades más importantes de las funciones B_i^k , con $i \in \mathbb{Z}$ y $k \in \mathbb{N}$.

Lema 3.1. *Si $k \geq 1$ y $x \notin (t_i, t_{i+k+1})$, entonces $B_i^k(x) = 0$.*

Demostración. Ya se ha visto que esto es cierto para $k = 1$, Así pues, procediendo por inducción, suponemos que es cierto para $k - 1$ y vemos que también es cierto para k . Si $x \notin (t_i, t_{i+k+1})$, entonces $x \notin (t_i, t_{i+k})$ y $x \notin (t_{i+1}, t_{i+k+1})$. Por hipótesis de inducción, $B_i^{k-1}(x) = 0$ y $B_{i+1}^{k-1}(x) = 0$. Por tanto, de la ecuación (3.1) se deduce que $B_i^k(x) = 0$. \square

Nótese que el lema anterior no es cierto para $k = 0$ dado que $B_i^0(t_i) = 1$.

Lema 3.2. *Sea $k \geq 0$. Si $x \in (t_i, t_{i+k+1})$, entonces $B_i^k(x) > 0$.*

Demostración. Previamente hemos visto que esta propiedad se cumple para $k = 0$ y $k = 1$. Como en la demostración anterior procedemos por inducción, suponiendo que es cierto para $k - 1$, con $k \geq 2$. De la ecuación (3.1) se tiene pues que si $x \in (t_i, t_{i+k+1})$, entonces $B_i^k(x) > 0$. \square

Teniendo en cuenta que uno de los objetivos principales del estudio de las funciones B-splines B_i^k es usarlos como base para generar funciones spline de grado k , analicemos a continuación su comportamiento en combinaciones lineales de la forma $\sum_{i=-\infty}^{\infty} c_i B_i^k(x)$.

Lema 3.3. *Bajo las condiciones anteriores se cumple*

$$\sum_{i=-\infty}^{\infty} c_i B_i^k = \sum_{i=-\infty}^{\infty} [c_i V_i^k + c_{i-1}(1 - V_i^k)] B_i^{k-1}.$$

Demostración. Aplicando la ecuación (3.3) a la serie obtenemos:

$$\begin{aligned} \sum_{i=-\infty}^{\infty} c_i B_i^k &= \sum_{i=-\infty}^{\infty} c_i [V_i^k B_i^{k-1} + (1 - V_{i+1}^k) B_{i+1}^{k-1}] \\ &= \sum_{i=-\infty}^{\infty} [c_i V_i^k + c_{i-1}(1 - V_i^k)] B_i^{k-1}. \end{aligned}$$

\square

El Lema 3.3 permite expresar combinaciones lineales de $\{B_i^k\}$ en términos de $\{B_i^{k-1}\}$. Los coeficientes c_i pueden ser constantes o incluso funciones $c_i = C_i(x)$. Por tanto, podremos evaluar una función $f(x)$ expresada en la forma:

$$f(x) = \sum_{i=-\infty}^{\infty} C_i^k(x)B_i^k(x),$$

siendo C_i^k funciones conocidas.

Para ello, sea

$$C_i^{k-1}(x) = C_i^k(x)V_i^k(x) + C_{i-1}^k(x)[1 - V_i^k(x)]. \quad (3.4)$$

Mediante esta fórmula y el Lema 3.3 obtenemos:

$$\sum_{i=-\infty}^{\infty} C_i^k(x)B_i^k(x) = \sum_{i=-\infty}^{\infty} C_i^{k-1}(x)B_i^{k-1}(x).$$

Reiterando este argumento ($k - 1$) veces llegamos a expresar f en términos de la familia de B-splines de orden cero:

$$\sum_{i=-\infty}^{\infty} C_i^k(x)B_i^k(x) = \sum_{i=-\infty}^{\infty} C_i^0(x)B_i^0(x).$$

En el lado derecho de esta igualdad aparecen las funciones B-splines B_i^0 y funciones C_i^0 , por lo que se trata de una expresión fácil de calcular, ya que es precisamente $B_i^0 = 1$ si $x \in [t_i, t_{i+1})$ y se anula en el resto de intervalos. Faltaría pues establecer cómo obtener las funciones C_i^0 a partir de C_i^k . Esto se realiza de manera idéntica teniendo en cuenta que la ecuación (3.4) se puede reescribir utilizando la ecuación (3.2) como:

$$\begin{aligned} C_i^{k-1}(x) &= C_i^k(x) \frac{x - t_i}{t_{i+k} - t_i} + C_{i-1}^k(x) \left[1 - \frac{x - t_i}{t_{i+k} - t_i} \right] \\ &= [(x - t_i)C_i^k(x) + (t_{i+k} - x)C_{i-1}^k(x)] / (t_{i+k} - t_i). \end{aligned} \quad (3.5)$$

Con el fin de seguir generalizando las propiedades de B-splines que hemos para $k = 0, 1$ establecemos ahora el siguiente

Lema 3.4. $\sum_{i=-\infty}^{\infty} B_i^k(x) \equiv 1, \forall k \in \mathbb{N}$.

Demostración. Sea $\sum_{i=-\infty}^{\infty} C_i^k B_i^k(x)$, con $C_i^k = 1, \forall i \in \mathbb{Z}$. Si fijamos x , podemos proceder usando la ecuación (3.5) para calcular

$$\begin{aligned} C_i^{k-1} &= [(x - t_i)C_i^k + (t_{i+k} - x)C_{i-1}^k] / (t_{i+k} - t_i) \\ &= [x - t_i + t_{i+k} - x] / (t_{i+k} - t_i) \\ &= 1. \end{aligned}$$

Repitiendo el argumento observamos que $C_i^j = 1, \forall j = k, k - 1, \dots, 0$. Por lo tanto, $\sum_{i=-\infty}^{\infty} B_i^k(x) = \sum_{i=-\infty}^{\infty} B_i^0(x) \equiv 1$.

□

La siguiente propiedad nos describe cómo se comporta la derivada de una función B-spline B_i^k , en términos de B-splines de orden $k - 1$. (Véase la demostración en [2, págs. 346-348].

Lema 3.5. Para $k \geq 2$,

$$\frac{d}{dx} B_i^k(x) = \left(\frac{k}{t_{i+k} - t_i} \right) B_i^{k-1}(x) - \left(\frac{k}{t_{i+k+1} - t_{i+1}} \right) B_{i+1}^{k-1}(x).$$

Cuando $k = 1$, la expresión es válida para todo x excepto $x = t_i, t_{i+1}, t_{i+2}$.

Nota. Aplicando el Lema 3.5 a la expresión $\sum_{i=-\infty}^{\infty} c_i B_i^k$ y teniendo en cuenta el Lema 3.3, se obtiene la siguiente fórmula, que resulta de gran utilidad en la diferenciación numérica:

$$\frac{d}{dx} \sum_{i=-\infty}^{\infty} c_i B_i^k(x) = k \sum_{i=-\infty}^{\infty} \left(\frac{c_i - c_{i-1}}{t_{i+k} - t_i} \right) B_i^{k-1}(x), \quad k \geq 2. \quad (3.6)$$

Lema 3.6. Para $k \geq 1$, las funciones B-splines cumplen $B_i^k \in \mathcal{C}^{k-1}(\mathbb{R})$.

Demostración. Es evidente que B_i^1 es continuo: $B_i^1 \in \mathcal{C}^0(\mathbb{R})$. Procediendo por inducción, supongamos ahora que $B_i^k \in \mathcal{C}^{k-1}(\mathbb{R})$. Por el Lema 3.5 tenemos que $\frac{d}{dx} B_i^{k+1} \in \mathcal{C}^{k-1}(\mathbb{R})$, ya que la derivada es combinación lineal de B_i^k y B_{i+1}^k . Por tanto, $B_i^{k+1} \in \mathcal{C}^k(\mathbb{R})$. □

A continuación vemos una propiedad que nos describe cómo se comporta la integral de una función B-spline B_i^k , y posteriormente algunos resultados que permitirán presentar a un conjunto de B-splines como una base para otras funciones spline S^k .

Lema 3.7. $\int_{-\infty}^x B_i^k(s) ds = \left(\frac{t_{i+k+1} - t_i}{k+1} \right) \cdot \sum_{j=i}^{\infty} B_j^{k+1}(x)$.

Demostración. Utilizando la ecuación (3.6) podemos verificar que las derivadas a ambos lados de la expresión son iguales si tomamos

$$c_j = \begin{cases} 0, & \text{si } j < i, \\ 1, & \text{si } j \geq i. \end{cases}$$

Se observa que $c_j - c_{j-1}$ es cero excepto cuando $j=i$, de donde se deduce que:

$$B_i^k(x) = \left(\frac{t_{i+k+1} - t_i}{k+1} \right) \cdot (k+1) \cdot \left(\frac{1}{t_{i+k+1} - t_i} \right) B_i^k(x).$$

Con el fin de asegurarse de que las funciones a ambos lados de la ecuación no difieren en una constante, observe que las dos se reducen a cero cuando $x = t_i$.

□

Sean los B-splines $B_0^k, B_1^k, \dots, B_k^k$. Cuando éstos se restringuen a un sólo intervalo entre los nodos (t_j, t_{j+1}) , el resultado es un conjunto de polinomios de grado $\leq k$, estos son precisamente todos los que al evaluarlos en dicho intervalo, son estrictamente positivos. Un resultado interesante es que este conjunto de funciones restringido al intervalo (t_j, t_{j+1}) forman una base del espacio polinomial \mathbb{P}_k .

Lema 3.8. *El conjunto de funciones B-spline $\{B_j^k, B_{j+1}^k, \dots, B_{j+k}^k\}$ es linealmente independiente en (t_{k+j}, t_{k+j+1}) .*

Demostración. Procedamos por inducción sobre k . Sea $k = 0$, el lema afirma que $\{B_j^0\}$ es linealmente independiente en el intervalo (t_j, t_{j+1}) . Esto es obviamente cierto. Supongamos ahora que el enunciado es cierto para $k - 1$. Sea $S = \sum_{i=0}^k c_{j+i} B_{j+i}^k$, y supongamos que $S|_{(t_{k+j}, t_{k+j+1})} = 0$. De la ecuación (3.6) y teniendo en cuenta que $B_{j+k+1}^{k-1} = 0$ y $B_j^{k-1} = 0$ en (t_{k+j}, t_{k+j+1}) tenemos que:

$$S'_{|(t_{k+j}, t_{k+j+1})} = 0 = k \sum_{i=1}^k \left(\frac{c_{j+i} - c_{j+i-1}}{t_{j+i+k} - t_{j+i}} \right) B_{j+i}^{k-1}|_{(t_{k+j}, t_{k+j+1})}.$$

Aplicando la hipótesis de inducción en $\{B_{j+1}^{k-1}, B_{j+2}^{k-1}, \dots, B_{j+k}^{k-1}\}$, tenemos que el conjunto es linealmente independiente en dicho intervalo. Por tanto, los coeficientes de la ecuación (3.6) deben ser nulos, es decir, $c_j = c_{j+1} = \dots = c_{j+k}$. Si denotamos $c_i = \lambda$, $\forall j \leq i \leq j+k$ y aplicando el Lema 3.4 tenemos que $S(x) = \lambda$ en (t_{k+j}, t_{k+j+1}) , y teniendo en cuenta que todos los términos son nulos en dicho intervalo salvo $B_j^k, B_{j+1}^k, \dots, B_{j+k}^k$, de haber supuesto que S se anula en (t_{k+j}, t_{k+j+1}) , concluimos que $\lambda = 0$. □

Lema 3.9. *El conjunto de funciones B-spline $\{B_{-k}^k, B_{-k+1}^k, \dots, B_{n-1}^k\}$ es linealmente independiente en (t_0, t_n) .*

Demostración. Sea $S = \sum_{i=-k}^{n-1} c_i B_i^k$ y supongamos que $S|_{(t_0, t_n)} = 0$. En el intervalo (t_0, t_1) sólo $B_{-k}^k, B_{-k+1}^k, \dots, B_0^k$, son no nulos, y por tanto:

$$S|_{(t_0, t_1)} = 0 = \sum_{i=-k}^0 c_i B_i^k|_{(t_0, t_1)}. \tag{3.7}$$

Por el Lema 3.8, el conjunto $\{B_{-k}^k, B_{-k+1}^k, \dots, B_0^k\}$, es linealmente independiente en (t_0, t_1) . Luego, de la ecuación (3.7) se deduce que $c_i = 0$ cuando $-k \leq i \leq 0$. Si $c_i = 0$, $\forall i$ la demostración habría concluido. Así pues, podemos suponer que c_j es el primer término no nulo para $j \geq 1$. Entonces, $(t_j, t_{j+1}) \subset (t_0, t_n)$. Para cualquier $x \in (t_j, t_{j+1})$ llegamos a una contradicción:

$$S(x) = 0 = \sum_{i=j}^{n-1} c_i B_i^k(x) = c_j B_j^k(x) \neq 0.$$

Por lo tanto, $c_i = 0, \forall i = -k, \dots, n - 1$. □

Del Lema 3.9 se sigue que la dimensión del conjunto de funciones B-spline de orden k definidos en $[t_0, t_n]$ es al menos $n + k$. Teniendo en cuenta la Proposición 2.3 se concluye el siguiente resultado

Teorema 3.10. *La dimensión de \mathcal{S}^k es $n + k$.*

Para finalizar esta sección analizaremos la convergencia de las funciones B-spline lineales y las B-splines cúbicas.

Sea Δ una partición arbitraria y B un B-spline lineal. Si $f \in \mathcal{C}^2([t_i, t_{i+1}])$, entonces se cumple la estimación de error

$$|f(x) - B(x)| \leq \frac{M_i}{8} h_i^2, \quad x \in [t_i, t_{i+1}],$$

con $h_i = \max_{0 \leq i \leq n-1} t_{i+1} - t_i$ y $M_i = \max_{x \in [t_i, t_{i+1}]} |f''(x)|$. En particular, si $f \in \mathcal{C}^2([a, b])$, entonces

$$|f(x) - B(x)| \leq \frac{M}{8} h^2, \quad x \in [a, b],$$

con $h = \max_{0 \leq i \leq n-1} t_{i+1} - t_i$ y $M = \max_{x \in [a, b]} |f''(x)|$.

De aquí podemos observar que, a diferencia de la interpolación polinómica en general, la interpolación de una función por medio de B-splines lineales garantiza una sucesión de interpolantes uniformemente convergentes, siempre que el diámetro de la partición $h = \|\Delta\|$ tienda a cero y $f \in \mathcal{C}^2([a, b])$.

Aunque el espacio de los splines lineales \mathcal{S}^1 proporciona interpolantes convergentes al aumentar el número de nodos, su uso en la práctica es reducido debido a su baja regularidad en los mismos. Teniendo en cuenta esto y que en la práctica se utilizan los splines cúbicos \mathcal{S}^3 , enunciaremos a continuación un resultado de convergencia para este tipo de B-splines.

Sea $f \in \mathcal{C}^4([a, b])$ y sea B un B-spline cúbico completo que interpola a f en Δ , con Δ una partición arbitraria para la que existe un $K \geq 1$ cumpliendo que

$$\frac{\|\Delta\|}{x_{j+1} - x_j} \leq K, \quad \forall j = 0, \dots, n - 1.$$

Dicha constante K es una medida de cuánto está de lejos Δ de ser equiespaciada.

Entonces, existen constantes C_k independientes de Δ , con $0 \leq k \leq 3$ y $0 < C_k \leq 2$, tales que

$$\|f^{(k)} - B^{(k)}\|_\infty \leq C_k \cdot K \cdot \|\Delta\|^{4-k} \cdot \|f^{(4)}\|_\infty, \quad 0 \leq k \leq 3.$$

Luego, si asumimos $\sup_{n \geq 1} \max_{0 \leq j \leq n-1} \frac{\|\Delta\|}{x_{j+1} - x_j}$ acotado, entonces el spline cúbico completo y sus tres primeras derivadas convergen uniformemente a f en $[a, b]$.

3.3. Aplicaciones de las funciones B-Spline.

En el capítulo anterior estudiamos las funciones spline de orden k S^k , y vimos que son funciones a trozos de clase C^{k-1} que se componen de polinomios de grado $\leq k$ en los intervalos $[t_0, t_1), [t_1, t_2), \dots, [t_{n-1}, t_n]$. Recordemos que denotamos por S^k a la familia de splines cuyo dominio es $[t_0, t_n]$.

Nuestro objetivo ahora es ver cómo se relacionan estas funciones con las funciones B-spline. Para ello restringimos el dominio de las funciones B-spline B_i^k de la recta real al intervalo $[t_0, t_n]$, y lo denotamos por $B_i^k|_{[t_0, t_n]}$.

Teorema 3.11. *Una base para el espacio S^k es*

$$\{B_i^k|_{[t_0, t_n]} : -k \leq i \leq n - 1\}. \tag{3.8}$$

Por tanto, la dimensión de S^k es $k + n$.

Demostración. Es evidente que las funciones $B_i^k|_{[t_0, t_n]}$ de la ecuación (3.8) pertenecen al espacio de las funciones S^k , ya que son funciones spline de grado k , definidos en los nodos $t_0 < t_1 < \dots < t_n$. Como se vio en la Proposición 2.3 del capítulo anterior la dimensión de S^k es a lo sumo $k + n$.

Por tanto, para concluir la demostración falta probar que la expresión (3.8) es una base, y aplicando el Lema 3.9 se concluye que las expresiones dadas en dicha ecuación son linealmente independientes, teniendo en cuenta que la dimensión es $k + n$. En efecto, es una base de S^k . □

Tras la demostración del Teorema 3.11 podemos decir que las funciones $1, x, x^2, \dots, (x - t_1)_+^k, (x - t_2)_+^k, \dots, (x - t_{n-1})_+^k$ generadas en la Proposición 2.3 de la Sección 2.3 forman una base de S^k . Sin embargo, emplear esta base puede

generar dificultades en los cálculos numéricos, debido a esto es recomendable emplear en su lugar la base de B-splines que se enuncia en el Teorema 3.11.

Las funciones spline se pueden emplear como funciones interpoladoras. Sea x_1, x_2, \dots, x_n un conjunto de puntos dados, nos gustaría interpolar datos arbitrarios en estos puntos mediante funciones spline de la forma $\sum_{j=1}^n c_j B_j^k$. Para que esto sea posible, la **matriz de interpolación** A dada por $A_{ij} = B_j^k(x_i), (1 \leq i, j \leq n)$ debe ser no singular.

Un importante teorema que nos proporciona las condiciones necesarias y suficientes para garantizar la solución a este problema es el *Teorema de Schoenberg-Whitney* que enunciamos a continuación. (Véase la demostración en [2, págs. 355-357]).

Teorema 3.12. *Sea x_1, x_2, \dots, x_n un conjunto de puntos dados. Para que la matriz A dada por $A_{ij} = B_j^k(x_i)$ sea no singular, es necesario y suficiente que no contenga elementos nulos en su diagonal.*

Ejemplo 7. Fijado como nodos el conjunto de puntos de los enteros (\mathbb{Z}). Si deseamos interpolar mediante combinaciones lineales de $B_1^2, B_2^2, \dots, B_5^2$, ¿podemos usar el conjunto de puntos $\{2,4, 4,2, 4,6, 5,5, 6,8\}$?

Solución. En efecto, podemos aplicar el Teorema 3.12, así que basta con comprobar que los elementos de la diagonal de $A_{ij} = B_j^2(x_i)$ son no nulos, y esto es cierto si los puntos están en el soporte de los $B_i^2, \forall i = 1, \dots, 5$, (Lema 3.2) veámoslo:

El soporte de $B_i^2 = (t_i, t_{i+3})$, si identificamos los nodos con la recta real, podemos suponer que estamos en el intervalo $[1, 8]$. Sea $A_{ij} = B_j^2(x_i)$, luego

$$\begin{aligned} a_{11} &= B_1^2(2,4) \quad \text{y} \quad 2,4 \in [1, 4) \quad \Rightarrow \quad a_{11} > 0. \\ a_{22} &= B_2^2(4,2) \quad \text{y} \quad 4,2 \in (2, 5) \quad \Rightarrow \quad a_{22} > 0. \\ a_{33} &= B_3^2(4,6) \quad \text{y} \quad 4,6 \in (3, 6) \quad \Rightarrow \quad a_{33} > 0. \\ a_{44} &= B_4^2(5,5) \quad \text{y} \quad 5,5 \in (4, 7) \quad \Rightarrow \quad a_{44} > 0. \\ a_{55} &= B_5^2(6,8) \quad \text{y} \quad 6,8 \in (5, 8] \quad \Rightarrow \quad a_{55} > 0. \end{aligned}$$

Concluimos por tanto, del Teorema de Schoenberg-Whitney que el problema de interpolación tendrá solución única, la cual puede obtenerse fácilmente resolviendo el correspondiente sistema de dimensión 5, y cuya solución nos brindarán los coeficientes de la combinación lineal de funciones $B_i^2, i = 1, \dots, 5$.

Bibliografía

- [1] **K.E. Atkinson.**- *An Introduction to numerical analysis*, John Wiley and Sons, 1989.
- [2] **D. Kincaid, W. Cheney.**- *Análisis Numérico. Las Matemáticas del Cálculo Científico*, Addison-Wesley Iberoamericana, 1994.
- [3] **P.M. Prenter.**- *Spline and Variational Methods*, John Wiley and Sons, 1991.
- [4] **J. Stoer, R. Bulirsch.**- *Introduction to Numerical Analysis*, Springer-Verlag, New York, 1993.
- [5] **Grado en Matemáticas ULL - Métodos Numéricos II.**-<https://e-guia.ull.es/matematicas/query.php?codigo=299343203>

Apéndice

Programación Matlab

```
# Código Matlab: Figura2.3.m
```

```
n=10;
syms x;
xn=[-2 -0.5 0.5 1 2 4 6 7 8.8 10];
yn=[1.25 0.6 1 2 0 -2 -5.25 -1.75 0.75 2];
plot(xn,yn,'*r')
hold on
p=0;
for i=1:n
    L=1;
    for j=1:n
        if j~=i
            L=L*(x-xn(j))/(xn(i)-xn(j));
        end
    end
    p=p+L*yn(i);
end
p=simplify(p)
pretty(p)

x=-2:0.01:10;
```

```

f=(10810479693733*x.^9)./104796710943724800 -
(85259461754711*x.^8)./23817434305392000 +
(462522803579339*x.^7)./9658683036288000 -
(4094711073018667*x.^6)./13609962460224000 +
(236190600087064699*x.^5)./299419174124928000 +
(80085127195097*x.^4)./439031047104000 -
(2288044281290226307*x.^3)./523983554718624000 +
(108509696483261191*x.^2)./23817434305392000 +
(793465203611429*x)./550402893612000 -
2450213544709/7088522114700

g0=0.003*x.^3+0.02*x.^2-0.4*x+0.395;
g1=0.807*x.^3+1.225*x.^2+0.198*x+0.494;
g2=-4.953*x.^3+9.865*x.^2-4.129*x+1.217;
g3=2.263*x.^3-11.785*x.^2+17.512*x-5.99;
g4=-0.532*x.^3+4.99*x.^2-16.03*x+16.36;
g5=0.718*x.^3-10.02*x.^2+43.982*x-63.58;
g6=-0.725*x.^3+15.96*x.^2-111.905*x+248.22;
g7=-0.209*x.^3+5.129*x.^2-40.307*x+100.834;
g8=0.11*x.^3-3.292*x.^2+33.8*x-116.559;
g=(x<-0.5).*g0+((x>=-0.5)&(x<0.5)).*g1+((x>=0.5)&(x<1)).*g2 +
((x>=1)&(x<2)).*g3+((x>=2)&(x<4)).*g4+((x>=4)&(x<6)).*g5 +
((x>=6)&(x<7)).*g6+((x>=7)&(x<8.8)).*g7+((x>=8.8)&(x<=10)).*g8;

plot(x,f,x,g);

```

Abstract

The polynomial interpolation consists in approximating a desired function by a polynomial function, generated from a finite number of known data. This tool is often used in many numerical processes of Applied Mathematics and in diverse areas of Engineering. Due to the importance of the subject because of its many applications, this report aims to strengthen the knowledge acquired about the classic polynomial interpolation, and introduce and study the concept of Spline Functions, a tool within the polynomial interpolation that fits conveniently to certain activities of the daily life.

The main aim of this work is to introduce the concept of Spline Functions and to study its most relevant properties. These functions will provide solutions to us about the problems that could be generated in the polynomial interpolation by applying Newton or Lagrange methods for large data tables, softening the behavior of our interpolating function. They will provide us with a basis that will allow us to construct and to work with simple piecewise polynomials of an efficient way.

1. Introduction

An interpolant polynomial can be obtained from the classical Newton or Lagrange methods. It is a useful tool when a low number of interpolant nodes is required. However, when this number increases, the interpolant will be a polynomial of high degree, and thus a very oscillating function. In addition to this drawback, tedious calculus arises in this situation in both procedures.

A reasonable idea to avoid such high oscillations of the interpolant function for a high number of interpolant nodes would be not to find a unique polynomial of degree n defined in the interval $[a, b]$, but to consider a partition Δ of this interval and then to find a polynomial of low fixed degree in each subinterval that interpolates our function. This is precisely the idea of interpolation by **spline functions**.

2. Outline of the first Chapter

Let us present the problem that characterizes the classical polynomial interpolation problem:

From a given table of $n + 1$ points (x_i, y_i) , with $i = 0, 1, \dots, n$, and $y_i \in \mathbb{R}$ (or $y_i \in \mathbb{C}$),

$$\begin{array}{c|c|c|c|c|c|c} x & x_0 & x_1 & x_2 & \dots & x_n \\ \hline y & y_0 & y_1 & y_2 & \dots & y_n \end{array},$$

find a polynomial p of less possible degree, such that $p(x_i) = y_i$, $0 \leq i \leq n$. The polynomial p is said to interpolate the data. The next known result guarantees the uniqueness and the existence of the interpolant polynomial:

Theorem Let x_0, x_1, \dots, x_n be a set of distinct real numbers. Then, for arbitrary real (or complex) values y_0, y_1, \dots, y_n , there exists a unique polynomial $p_n \in \mathbb{P}_n$, such that $p_n(x_i) = y_i$, for all $0 \leq i \leq n$.

A very important result in Approximation Theory, that justifies the use of polynomials in the approximation of continuous functions in $[a, b]$, is the well known **Weierstrass Approximation Theorem**, that states that the space $\mathbb{P}([a, b])$ of polynomials defined in $[a, b]$ is dense in $\mathcal{C}([a, b])$ with respect to the uniform norm.

The most famous known methods for the computation of an interpolant polynomial are the Newton and Lagrange procedures. Despite polynomials are nice functions for the approximation of continuous functions, it is not true that when the number of interpolant nodes increases, then the interpolant polynomial converges to the function. Some well known results in the literature in this respect are the Bernstein counterexample, Runge's example, or the Marcinkiewicz or Faber's Theorems. In order to avoid these drawbacks, it is standard to choose **spline functions**.

3. Outline of the second chapter

Spline functions are piecewise polynomial functions of degree k in a partition Δ of the interval $[a, b]$ with a certain regularity condition: they belong in $\mathcal{C}^{(m)}([a, b])$, for some $m < k$ (it is very common to take $m = k - 1$). The associated interpolation problem by using such kind of functions gives rise to similar results as in the polynomial interpolation, with the improvement that the interpolant function has now few oscillations.

Due to the common use in several areas of Numerical Analysis and applied problems in Mathematics, Physics, Engineering, ..., in this second chapter we mostly

concentrate in the study of **cubic spline functions**. In particular, we analyze their efficient construction and we prove their most important property:

Theorem Let $f \in \mathcal{C}^2([a, b])$ and take $a = t_0 < t_1 < \dots < t_n = b$. If S is the natural cubic spline that interpolates f at the nodes t_i , $\forall 0 \leq i \leq n$, then:

$$\int_a^b |S''(x)|^2 dx \leq \int_a^b |f''(x)|^2 dx.$$

Some numerical experiments are carried out, in order to illustrate this important result. Spline functions of high order are also analyzed not so deeply.

4. Outline of the third chapter

B-spline functions are spline functions with remarkable properties. In particular, they are a family of elementary functions $\{B_i^k(x)\}_{i \in \mathbb{Z}}$ that form a basis of the spline functions of order k considered in the previous chapter. So, their study becomes an important tool for our purposes. This chapter is devoted to the analysis of their most important properties, in particular:

Theorem A basis of the space of spline functions \mathcal{S}^k of degree k is given by

$$\{B_i^k(x)\}_{i \in \mathbb{Z}}, \quad -k \leq i \leq n - 1.$$

So, the dimension of \mathcal{S}^k is $k + n$. To end, we also characterize the interpolation problem by B-spline functions. It is convenient to mention that the study of spline functions can be directly done from the study of B-spline functions in an independent way.

References

[1] **K.E. Atkinson**. - *An Introduction to numerical analysis*, John Wiley and Sons, 1989.
 [2] **D. Kincaid, W. Cheney**. - *Análisis Numérico. Las Matemáticas del Cálculo Científico*, Addison-Wesley Iberoamericana, 1994.
 [3] **P.M. Prenter**. - *Spline and Variational Methods*, John Wiley and Sons, 1991.
 [4] **J. Stoer, R. Bulirsch**. - *Introduction to Numerical Analysis*, Springer-Verlag, New York, 1993.
 [5] **Grado en Matemáticas ULL - Métodos Numéricos II**. - <https://e-guia.ull.es/matematicas/query.php?codigo=299343203>