LECTURE 10: *B-Splines*

1.11.2 *B-Splines: a basis for splines*

Throughout our discussion of standard polynomial interpolation, we viewed $\mathcal{P}_n$ as a linear space of dimension $n+1$, and then expressed the unique interpolating polynomial in several different bases (monomial, Newton, Lagrange). The most elegant way to develop spline functions uses the same approach. A set of *basis splines*, depending only on the location of the knots and the degree of the approximating piecewise polynomials can be developed in a convenient, numerically stable manner. (Cubic splines are the most prominent special case.)

For example, each cubic basis spline, or *B-spline*, is a continuous piecewise-cubic function with continuous first and second derivatives. Thus any linear combination of such B-splines will inherit the same continuity properties. The coefficients in the linear combination are chosen to obey the specified interpolation conditions.

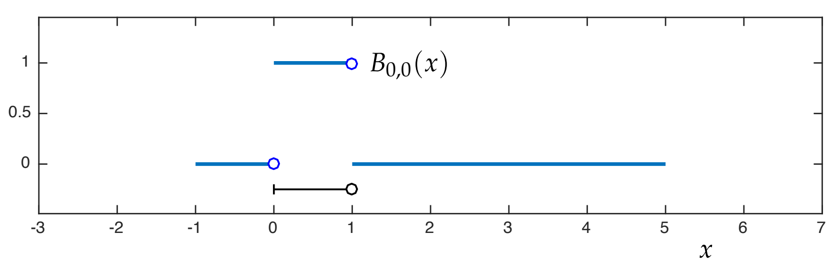
B-splines are built up recursively from constant B-splines. Though we are interpolating data at $n+1$ knots $x_0, \ldots, x_n$, to derive B-splines we need extra nodes outside $[x_0, x_n]$ as scaffolding upon which to construct the basis. Thus, add knots on either end of x_0 and x_n:

$$\cdots < x_{-2} < x_{-1} < x_0 < x_1 < \cdots < x_n < x_{n+1} < \cdots .$$

Given these knots, define the constant (zeroth-degree) B-splines:

$$B_{j,0}(x) = \begin{cases} 1 & x \in [x_j, x_{j+1}); \\ 0 & \text{otherwise.} \end{cases}$$

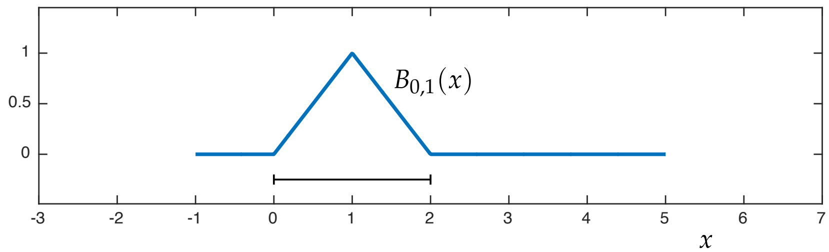
The following plot shows the basis function $B_{0,0}$ for the knots $x_j = j$. Note, in particular, that $B_{j,0}(x_{j+1}) = 0$. The line drawn beneath the spline marks the *support* of the spline, that is, the values of x for which $B_{0,0}(x) \neq 0$.

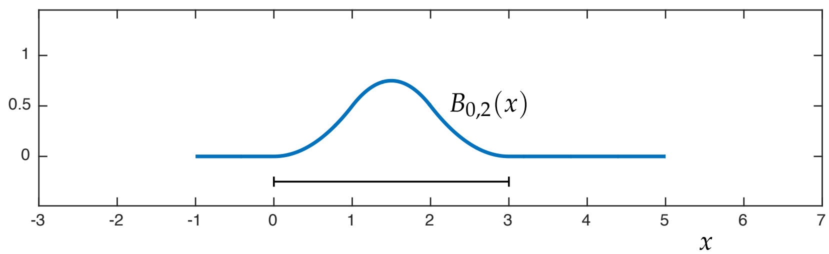


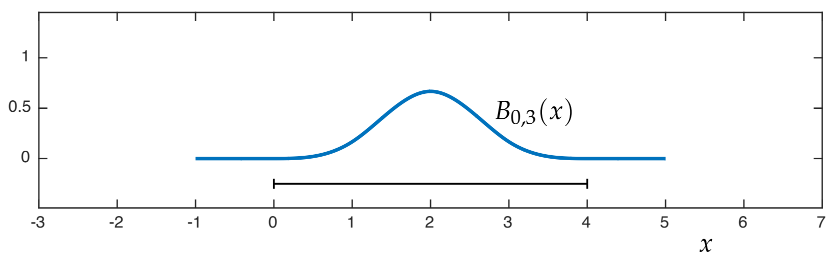
From these degree-0 B-splines, manufacture B-splines of higher degree via the recurrence

$$B_{j,k}(x) = \left(\frac{x - x_j}{x_{j+k} - x_j}\right) B_{j,k-1}(x) + \left(\frac{x_{j+k+1} - x}{x_{j+k+1} - x_{j+1}}\right) B_{j+1,k-1}(x). \tag{1.32}$$

While not immediately obvious from the formula, this construction ensures that $B_{j,k}$ has one more continuous derivative than does $B_{j,k-1}$. Thus, while $B_{j,0}$ is discontinuous (see previous plot), $B_{j,1}$ is continuous, $B_{j,2} \in C^1(\mathbb{R})$, and $B_{j,3} \in C^2(\mathbb{R})$. One can see this in the three plots below, where again $x_j = j$. As the degree increases, the B-spline $B_{j,k}$ becomes increasingly smooth. Smooth is good, but it has a consequence: the *support* of $B_{j,k}$ gets larger as we increase k. This, as we will see, has implications on the number of nonzero entries in the linear system we must ultimately solve to find the expansion of the desired spline in the B-spline basis.







From these plots and the recurrence defining $B_{j,k}$, one can deduce several important properties:

- $B_{j,k} \in C^{k-1}(\mathbb{R})$ (continuity);
- $B_{j,k}(x) = 0$ if $x \notin (x_j, x_{j+k+1})$ (compact support);
- $B_{j,k}(x) > 0$ for $x \in (x_j, x_{j+k+1})$ (positivity).

Finally, we are prepared to write down a formula for the spline that interpolates $\{(x_j, f_j)\}_{j=0}^{n}$ as a linear combination of the basis splines we have just constructed. Let $S_k(x)$ denote the spline consisting of piecewise polynomials in $\mathcal{P}_k$. In particular, S_k must obey the following properties:

- $S_k(x_j) = f_j$ for $j = 0, \ldots, n$;
- $S_k \in C^{k-1}[x_0, x_n]$ for $k \geq 1$.

The beauty of B-splines is that the second of these properties is automatically inherited from the B-splines themselves. (Any linear combination of $C^{k-1}(\mathbb{R})$ functions must itself be a $C^{k-1}(\mathbb{R})$ function.) The interpolation conditions give $n+1$ equations that constrain the unknown coefficients $c_{j,k}$ in the expansion of S_k:

(1.33) $$S_k(x) = \sum_j c_{j,k} B_{j,k}(x).$$

What limits should j have in this sum? For the greatest flexibility, let j range over all values for which

$$B_{j,k}(x) \neq 0 \quad \text{for some } x \in [x_0, x_n].$$

If $B_{j,k}(x) = 0$ for all $x \in [x_0, x_n]$, it cannot contribute to the interpolation requirement $S_k(x_j) = f_j$, $j = 0, \ldots, n$.

Figure 1.22 shows the B-splines of degree $k = 1, 2, 3$ that overlap the interval $[x_0, x_4]$ for $x_j = j$. For $k \geq 1$, $B_{j,k}(x)$ is supported on (x_j, x_{j+k+1}), and hence the limits on the sum in (1.33) take the form

(1.34) $$S_k(x) = \sum_{j=-k}^{n-1} c_{j,k} B_{j,k}(x), \qquad k \geq 1.$$

The sum involves $n+k$ coefficients $c_{j,k}$, which must be determined to

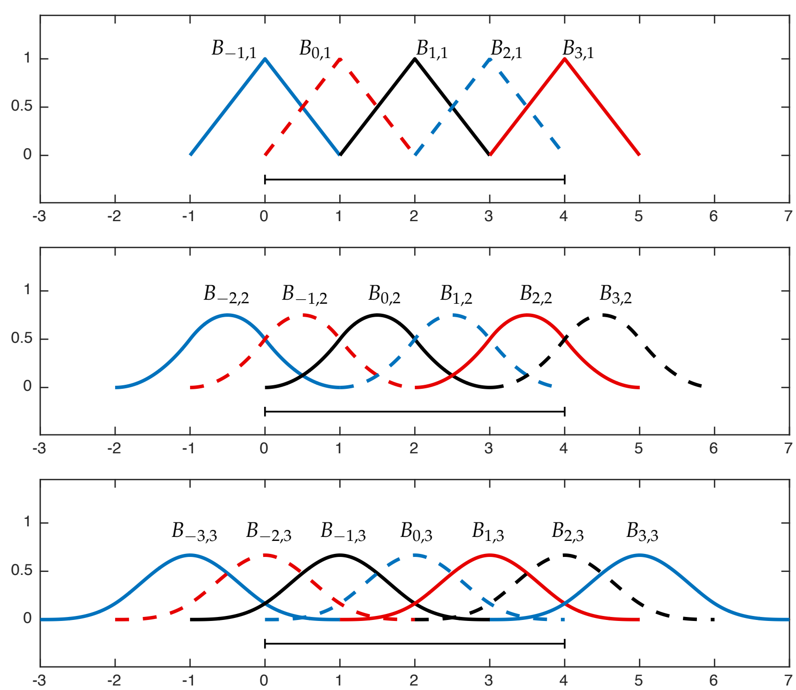


Figure 1.22: B-splines of degree $k = 1$ (top), $k = 2$ (middle), and $k = 3$ (bottom) that are supported on the interval $[x_0, x_n]$ for $x_j = j$ with $n = 4$. Note that $n + k$ B-splines are supported on $[x_0, x_n]$.

satisfy the $n+1$ interpolation conditions

$$f_\ell = S_k(x_\ell) = \sum_{j=-k}^{n-1} c_{j,k} B_{j,k}(x_\ell), \qquad \ell = 0, \ldots, n.$$

Before addressing general $k \geq 1$, we pause to handle the special case of $k = 0$, i.e., constant splines.

1.11.3 *Constant splines, $k = 0$*

The constant B-splines give $B_{n,0}(x_n) = 1$ and so, unlike the general $k \geq 1$ case, the $j = n$ B-spline must be included in the spline sum:

$$S_0(x) = \sum_{j=0}^{n} c_{j,0} B_{j,0}(x).$$

The interpolation conditions give, for $\ell = 0, \ldots, n$,

$$\begin{aligned} f_\ell = S_0(x_\ell) &= \sum_{j=0}^{n} c_{j,0} B_{j,0}(x_\ell) \\ &= c_{\ell,0} B_{\ell,0}(x_\ell) = c_{\ell,0}, \end{aligned}$$

since $B_{j,0}(x_\ell) = 0$ if $j \neq \ell$, and $B_{\ell,0}(x_\ell) = 1$ (recall the plot of $B_{0,0}(x)$ shown earlier). Thus $c_{\ell,0} = f_\ell$, and the degree $k = 0$ spline interpolant is simply

$$S_0(x) = \sum_{j=0}^{n} f_j B_{j,0}(x).$$

LECTURE 11: *Matrix Determination of Splines; Energy Minimization*

1.11.4 *General case, $k \geq 1$*

Now consider the general spline interpolant of degree $k \geq 1$,

$$S_k(x) = \sum_{j=-k}^{n-1} c_{j,k} B_{j,k}(x),$$

with constants $c_{-k,k}, \ldots, c_{n-1,k}$ determined to satisfy the interpolation conditions $S_k(\ell) = f_\ell$, i.e.,

$$\sum_{j=-k}^{n-1} c_{j,k} B_{j,k}(x_\ell) = f_\ell, \quad \ell = 0, \ldots, n.$$

By now we are accustomed to transforming constraints like this into matrix equations. Each value $\ell = 0, \ldots, n$ gives a row of the equation

$$(1.35) \quad \begin{bmatrix} B_{-k,k}(x_0) & B_{-k+1,k}(x_0) & \cdots & B_{n-1,k}(x_0) \\ B_{-k,k}(x_1) & B_{-k+1,k}(x_1) & \cdots & B_{n-1,k}(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ B_{-k,k}(x_n) & B_{-k+1,k}(x_n) & \cdots & B_{n-1,k}(x_n) \end{bmatrix} \begin{bmatrix} c_{-k,k} \\ c_{-k+1,k} \\ \vdots \\ c_{n-1,k} \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_n \end{bmatrix}.$$

Let us consider the matrix in this equation. The matrix will have $n+1$ rows and $n+k$ columns, so when $k > 1$ the system of equations will be *underdetermined*. Since B-splines have 'small support' (i.e., $B_{j,k}(x) = 0$ for most $x \in [x_0, x_n]$), this matrix will be *sparse*: most entries will be zero.

One could obtain an $(n+1) \times (n+1)$ matrix by arbitrarily setting $k-1$ certain values of $c_{j,k}$ to zero, but this would miss a great opportunity: we can constructively include all $n+k$ B-splines and impose k extra properties on S_k to pick out a unique spline interpolant from the infinitely many options that satisfy the interpolation conditions.

The following subsections will describe the particular form the system (1.35) takes for $k = 1, 2, 3$. In each case we will illustrate the resulting spline interpolant through the following data set.

(1.36)

j	0	1	2	3	4
x_j	0	1	2	3	4
f_j	1	3	2	−1	1

1.11.5 *Linear splines, $k = 1$*

Linear splines are simple to construct: in this case $n + k = n + 1$, so the matrix in (1.35) is square. Let us evaluate it: since

$$B_{j,1}(x_\ell) = \begin{cases} 1, & j = \ell; \\ 0, & j \neq \ell, \end{cases}$$

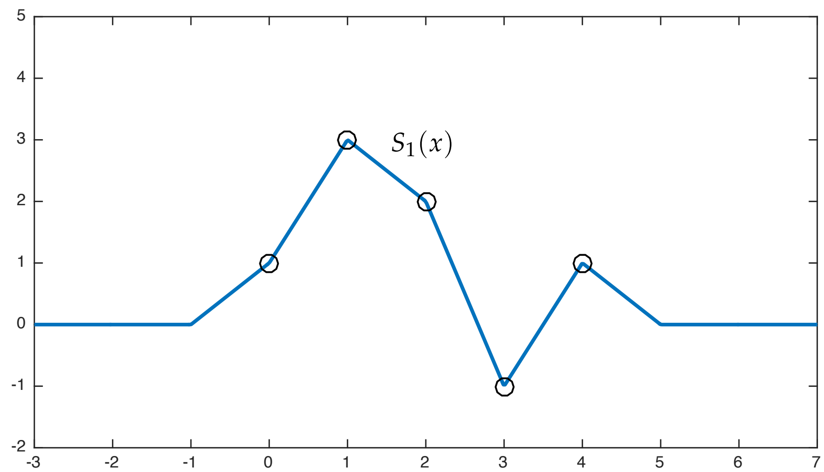


Figure 1.23: Linear spline S_1 interpolating 5 data points $\{(x_j, f_j)\}_{j=0}^4$.

the matrix is simply

$$\begin{bmatrix} B_{-1,1}(x_0) & B_{0,1}(x_0) & \cdots & B_{n-1,1}(x_0) \\ B_{-1,1}(x_1) & B_{0,1}(x_1) & \cdots & B_{n-1,1}(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ B_{-1,1}(x_n) & B_{0,1}(x_n) & \cdots & B_{n-1,1}(x_n) \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix} = \mathbf{I}.$$

The system (1.35) is thus trivial to solve, reducing to

$$\begin{bmatrix} c_{-1,1} \\ c_{-0,k} \\ \vdots \\ c_{n-1,k} \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_n \end{bmatrix},$$

which gives the unique linear spline

$$S_1(x) = \sum_{j=-1}^{n-1} f_{j+1} B_{j,1}(x).$$

This above discussion is a pedantic way to arrive at an obvious solution: Since the jth 'hat function' B-spline equals one at x_{j+1} and zero at all other knots, just write the unique formula for the interpolant immediately.

Figure 1.23 shows the unique piecewise linear spline interpolant to the data in (1.36), which is a linear combination of the five linear splines shown in Figure 1.22. Explicitly,

$$\begin{aligned} S_1(x) &= f_0 B_{-1,1}(x) + f_1 B_{0,1}(x) + f_2 B_{1,1}(x) + f_3 B_{2,1}(x) + f_4 B_{3,1}(x) \\ &= B_{-1,1}(x) + 3\,B_{0,1}(x) + 2\,B_{1,1}(x) - B_{2,1}(x) + B_{3,1}(x). \end{aligned}$$

Note that linear splines are simply C^0 functions that interpolate a given data set—between the knots, they are identical to the piecewise linear functions constructed in Section 1.10.1. Note that $S_1(x)$ is supported on (x_{-1}, x_{n+1}) with $S_1(x) = 0$ for all $x \notin (x_{-1}, x_{n+1})$. This is a general feature of splines: Outside the range of interpolation, $S_k(x)$ goes to zero as quickly as possible for a given set of knots while still maintaining the specified continuity.

1.11.6 *Quadratic splines, $k = 2$*

The construction of quadratic B-splines from the linear splines via the recurrence (1.32) forces the functions $B_{j,2}$ to have a continuous derivative, and also to be supported over three intervals per spline, as seen in the middle plot in Figure 1.22. The interpolant takes the form

$$S_2(x) = \sum_{j=-2}^{n-1} c_{j,2} B_{j,2}(x),$$

with coefficients specified by $n+1$ equations in $n+2$ unknowns:

$$\begin{bmatrix} B_{-2,2}(x_0) & B_{-1,2}(x_0) & \cdots & B_{n-1,2}(x_0) \\ B_{-2,2}(x_1) & B_{-1,2}(x_1) & \cdots & B_{n-1,2}(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ B_{-2,2}(x_n) & B_{-1,2}(x_n) & \cdots & B_{n-1,2}(x_n) \end{bmatrix} \begin{bmatrix} c_{-2,2} \\ c_{-1,2} \\ \vdots \\ c_{n-1,2} \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_n \end{bmatrix}. \tag{1.37}$$

Since there are more variables than constraints, we expect infinitely many quadratic splines that interpolate the data.

Evaluate the entries of the matrix in (1.37). First note that

$$B_{j,2}(x_\ell) = 0, \quad \ell \notin \{j+1, j+2\},$$

so the matrix is zero in all entries except the main diagonal ($B_{j,2}(x_{j+2})$) and the first superdiagonal ($B_{j,2}(x_{j+1})$). To evaluate these nonzero entries, recall that the recursion (1.32) for B-splines gives

$$B_{j,2}(x) = \Big(\frac{x - x_j}{x_{j+2} - x_j}\Big) B_{j,1}(x) + \Big(\frac{x_{j+3} - x}{x_{j+3} - x_{j+1}}\Big) B_{j+1,1}(x).$$

Evaluate this function at x_{j+1} and x_{j+2}, using our knowledge of the

$B_{j,1}$ linear B-splines ('hat functions'):

$$B_{j,2}(x_{j+1}) = \Big(\frac{x_{j+1}-x_j}{x_{j+2}-x_j}\Big)B_{j,1}(x_{j+1}) + \Big(\frac{x_{j+3}-x_{j+1}}{x_{j+3}-x_{j+1}}\Big)B_{j+1,1}(x_{j+1})$$

$$= \Big(\frac{x_{j+1}-x_j}{x_{j+2}-x_j}\Big)\cdot 1 + \Big(\frac{x_{j+3}-x_{j+1}}{x_{j+3}-x_{j+1}}\Big)\cdot 0 = \frac{x_{j+1}-x_j}{x_{j+2}-x_j};$$

$$B_{j,2}(x_{j+2}) = \Big(\frac{x_{j+2}-x_j}{x_{j+2}-x_j}\Big)B_{j,1}(x_{j+2}) + \Big(\frac{x_{j+3}-x_{j+2}}{x_{j+3}-x_{j+1}}\Big)B_{j+1,1}(x_{j+2})$$

$$= \Big(\frac{x_{j+2}-x_j}{x_{j+2}-x_j}\Big)\cdot 0 + \Big(\frac{x_{j+3}-x_{j+2}}{x_{j+3}-x_{j+1}}\Big)\cdot 1 = \frac{x_{j+3}-x_{j+2}}{x_{j+3}-x_{j+1}}.$$

Use these formulas to populate the superdiagonal and subdiagonal of the matrix in (1.37). In the (important) special case of uniformly spaced knots

$$x_j = x_0 + jh, \quad \text{for fixed } h > 0,$$

gives the particularly simple formulas

$$B_{j,2}(x_{j+1}) = B_{j,2}(x_{j+2}) = \frac{1}{2},$$

hence the system (1.37) becomes

$$\begin{bmatrix} 1/2 & 1/2 & & & \\ & 1/2 & 1/2 & & \\ & & \ddots & \ddots & \\ & & & 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} c_{-2,2} \\ c_{-1,2} \\ c_{0,2} \\ \vdots \\ c_{n-1,2} \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_n \end{bmatrix},$$

where the blank entries are zero. This $(n+1)\times(n+2)$ system will have infinitely many solutions, i.e., infinitely many splines that satisfy the interpolation conditions. How to choose among them? Impose *one* extra condition, such as $S_2'(x_0) = 0$ or $S_2'(x_n) = 0$.

As an example, let us work through the condition $S_2'(x_0) = 0$; it raises an interesting issue. Refer to the middle plot in Figure 1.22. Due to the small support of the quadratic B-splines, $B_{j,2}'(x_0) = 0$ for $j > 0$, so

$$S_2'(x_0) = c_{-2,2}B_{-2,2}'(x_0) + c_{-1,2}B_{-1,2}'(x_0) + c_{0,2}B_{0,2}'(x_0). \tag{1.38}$$

The derivatives of the B-splines at knots are tricky to compute. Differentiating the recurrence (1.32) with $k = 2$, we can formally write

$$B_{j,2}'(x) = \Big(\frac{1}{x_{j+2}-x_j}\Big)B_{j,1}(x) + \Big(\frac{x-x_j}{x_{j+2}-x_j}\Big)B_{j,1}'(x) - \Big(\frac{1}{x_{j+3}-x_{j+1}}\Big)B_{j+1,1}(x) + \Big(\frac{x_{j+3}-x}{x_{j+3}-x_{j+1}}\Big)B_{j+1,1}'(x).$$