Polynomial interpolation

Michael S. Floater

January 13, 2014

These notes start a study of interpolation by polynomials, treating existence and uniqueness and various choices of basis functions.

1 The interpolation problem

Recall that a polynomial $p : \mathbb{R} \to \mathbb{R}$ of degree $\le n$ is a function that can be expressed in the form

$$p(x) = \sum_{j=0}^{n} c_j x^j, \qquad x \in \mathbb{R}, \tag{1}$$

where $c_j \in \mathbb{R}$, $j = 0, 1, \ldots, n$. We denote the linear space of all such polynomials by π_n. The degree of p is the largest j for which $c_j \neq 0$.

Let $f : [a, b] \to \mathbb{R}$ be a real function and let $x_0, x_1, \ldots, x_n$ be distinct points in $[a, b]$. We are interested in finding a polynomial $p \in \pi_n$ that *interpolates* f, in the sense that

$$p(x_i) = f(x_i), \qquad i = 0, 1, \ldots, n. \tag{2}$$

We will show that there exists a unique solution to this interpolation problem.

Because the monomials $\{1, x, x^2, \ldots, x^n\}$ form a basis for π_n, each polynomial $p \in \pi_n$ is uniquely determined by its $n + 1$ coefficients $c_0, c_1, \ldots, c_n$ in (1). If $p \in \pi_n$ satisfies the interpolation conditions (2) then these coefficients satisfy the linear system of $n + 1$ equations,

$$c_0 + c_1 x_i + \cdots + c_n x_i^n = f(x_i), \qquad i = 0, 1, \ldots, n.$$

This equation system can be expressed in matrix and vector notation as

$$Mc = f, \tag{3}$$

where

$$M = \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{bmatrix}, \quad c = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{bmatrix}, \quad f = \begin{bmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_n) \end{bmatrix}.$$

There is therefore a unique solution to the interpolation problem if and only if there is a unique solution vector c to (3). From linear algebra we know that the latter is true if the matrix M is non-singular. We will show that M is indeed non-singular by showing that the only solution to the homogeneous equation

$$Mc = 0, \tag{4}$$

is $c = 0$. Suppose that c is a solution to this equation. Then the polynomial

$$q(x) = \sum_{j=0}^{n} c_j x^j$$

has the property that $q(x_i) = 0$, $i = 0, 1, \ldots, n$. But then q has degree at most n and at least $n+1$ roots, and by the fundamental theorem of algebra, $q = 0$, which means that $c_j = 0$, $j = 0, 1, \ldots, n$. This proves that M is indeed non-singular. In summary we have shown

Theorem 1 *There exists a unique solution $p \in \pi_n$ to the interpolation problem (2).*

The matrix M is known as a Vandermonde matrix. It can be shown that its determinant is equal to

$$\det(M) = \prod_{0 \le i < j \le n} (x_j - x_i) \neq 0,$$

which also shows that M is non-singular.

Once we have found the coefficients c_i in (1), we typically want to *evaluate* p, i.e., find $p(x)$ for a given $x \in \mathbb{R}$. Horner's rule is a computationally efficient way of doing this. For example, the rule for $n = 3$ expresses $p(x)$ as

$$p(x) = c_0 + x(c_1 + x(c_2 + xc_3)),$$

and similarly for arbitrary n. Due to the $n - 1$ nested parentheses, the evaluation of p requires only n multiplications and n additions.

2 Other bases

As we have seen, we can find the interpolant p in the form (1) by solving the linear system (3) to find the coefficients $c_0, c_1, \ldots, c_n$. The representation (1) is known as the *monomial* form of p. The polynomial p is expressed in terms of the *monomial basis*, $\{1, x, x^2, \ldots, x^n\}$. In practice, we might want to represent p with respect to some other basis of π_n. Any linearly independent set of $n+1$ polynomials $\phi_0, \phi_1, \ldots, \phi_n$ in π_n forms a basis for π_n, and any $p \in \pi_n$ can be uniquely represented as a linear combination

$$p(x) = \sum_{j=0}^{n} c_j \phi_j(x). \tag{5}$$

The interpolation problem (2) then leads to the linear system of equations

$$c_0\phi_0(x_i) + c_1\phi_1(x_i) + \cdots + c_n\phi_n(x_i) = f(x_i), \qquad i = 0, 1, \ldots, n,$$

which can also be expressed as (3), but now with the matrix

$$M = \begin{bmatrix} \phi_0(x_0) & \phi_1(x_0) & \cdots & \phi_n(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_n(x_1) \\ \vdots & \vdots & & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \cdots & \phi_n(x_n) \end{bmatrix}. \tag{6}$$

This matrix, like the Vandermonde matrix, is non-singular because if c satisfies the homogeneous equation $Mc = 0$, the polynomial

$$q(x) = \sum_{j=0}^{n} c_j \phi_j(x)$$

satisfies $q(x_i) = 0$, $i = 0, 1, \ldots, n$, and because $q \in \pi_n$, we have $q = 0$. Since $\phi_0, \ldots, \phi_n$ form a basis for π_n it follows that $c_0 = c_1 = \cdots = c_n = 0$.

3 Lagrange basis

One choice of basis that makes the interpolation problem easy to solve is the Lagrange basis, $L_0, L_1, \ldots, L_n$, defined by

$$L_j(x) = \prod_{\substack{k=0 \\ k \neq j}}^{n} \frac{x - x_k}{x_j - x_k}.$$

Since

$$L_j(x_i) = \delta_{ij} := \begin{cases} 1 & \text{if } i = j; \\ 0 & \text{if } i \neq j, \end{cases}$$

it follows that the interpolation problem can be solved directly:

$$p(x) = \sum_{j=0}^{n} f(x_j) L_j(x).$$

There is no need to solve the linear system (3) because the matrix M in this case is the identity matrix.

4 Iterative interpolation

Another way of finding the interpolant is through recursion. Suppose that $q \in \pi_{n-1}$ satisfies

$$q(x_i) = f(x_i), \qquad i = 0, 1, \ldots, n-1,$$

and that $r \in \pi_{n-1}$ satisfies

$$r(x_i) = f(x_i), \qquad i = 1, 2, \ldots, n.$$

Then

$$p(x) := \frac{x_n - x}{x_n - x_0} q(x) + \frac{x - x_0}{x_n - x_0} r(x)$$

is the polynomial in π_n that solves the interpolation problem (2). This can easily be checked by considering the three cases $x = x_0$, $x = x_n$, and $x = x_i$, $i = 1, \ldots, n-1$. Applying this formula recursively leads to the Neville algorithm. We initialize the algorithm by setting $p_{i,0} = f(x_i)$ for $i = 0, 1, \ldots, n$. Then, for a given $x \in \mathbb{R}$, and for $r = 1, \ldots, n$, and $i = 0, 1, \ldots n - r$, we compute

$$p_{i,r}(x) = \frac{x_{i+r} - x}{x_{i+r} - x_i} p_{i,r-1}(x) + \frac{x - x_i}{x_{i+r} - x_i} p_{i+1,r-1}(x),$$

and $p_{0,n}(x)$ is then the value at x of the interpolant satisfying (2). The algorithm is a triangular scheme, as shown below for the case $n = 3$.

$$\begin{array}{cccc}
p_{0,0}(x) & & & \\
 & p_{0,1}(x) & & \\
p_{1,0}(x) & & p_{0,2}(x) & \\
 & p_{1,1}(x) & & p_{0,3}(x) \\
p_{2,0}(x) & & p_{1,2}(x) & \\
 & p_{2,1}(x) & & \\
p_{3,0}(x) & & &
\end{array}$$

The entries in the scheme are computed column by column from left to right, which each entry being computed from the two entries in the previous column, one above and one below.