Polynomial Interpolation

The polynomial interpolation problem is the problem of constructing a polynomial that passes through or interpolates \( n+1 \) data points \((x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)\). In chapter 3 we are going to see several techniques for constructing interpolating polynomials.

Lagrange Interpolation

To construct a polynomial of degree \( n \) passing through \( n+1 \) data points \((x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)\) we start by constructing a set of basis polynomials \( L_{n,k}(x) \) with the property that

\[
L_{n,k}(x_j) = \begin{cases} 
1 & \text{when } j = k \\
0 & \text{when } j \neq k 
\end{cases}
\]

These basis polynomials are easy to construct. For example for a sequence of \( x \) values \( \{x_0, x_1, x_2, x_3\} \) we would have the four basis polynomials

\[
L_{3,0}(x) = \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)}
\]

\[
L_{3,1}(x) = \frac{(x - x_0)(x - x_2)(x - x_3)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)}
\]

\[
L_{3,2}(x) = \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)}
\]

\[
L_{3,3}(x) = \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)}
\]

Once we have constructed these basis functions, we can form the \( n^{th} \) degree Lagrange interpolating polynomial

\[
L(x) = \sum_{k=0}^{n} y_k L_{n,k}(x)
\]

This polynomial does what we want it to do, because when \( x = x_j \) every one of the basis functions vanishes, except for \( L_{n,j}(x) \), which has value 1. Thus \( L(x_j) = y_j \) for every \( j \) and the polynomial interpolates each one of the data points in the original data set.

An example

Here is a set of data points.

<table>
<thead>
<tr>
<th>( x )</th>
<th>( y )</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1168</td>
<td>0.213631</td>
</tr>
<tr>
<td>4.19236</td>
<td>0.214232</td>
</tr>
<tr>
<td>4.20967</td>
<td>0.21441</td>
</tr>
<tr>
<td>4.46908</td>
<td>0.218788</td>
</tr>
</tbody>
</table>
Here is a plot of these points showing that they line up along a curve, but the curve is not quite linear.

To construct the Lagrange interpolating polynomial of degree 3 passing through these points we first compute basis functions:

\[ L_{3,0}(x) = \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} \]


\[ L_{3,1}(x) = \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} \]


\[ L_{3,2}(x) = \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_1)(x_2-x_3)} \]

\[ L_{3,3}(x) = \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)} \]


From these we construct the interpolating polynomial:

\[ L(x) = y_0 L_{3,0}(x) + y_1 L_{3,1}(x) + y_2 L_{3,2}(x) + y_3 L_{3,3}(x) \]

\[ = -0.00355245 x^3 + 0.0695519 x^2 - 0.386008 x + 0.871839 \]

Here are the original data points plotted along with the interpolating polynomial.

---

**Error Estimate**

For each polynomial interpolation method we examine in chapter 3 we will want to also generate an estimate of how accurate the method the method is for a particular application. For example, suppose that the data points \((x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)\) we interpolate by this method are actually generated by an underlying function \(f(x)\). That is, \(y_k = f(x_k)\) for all \(k\). How large can we then expect the error

\[ |f(x) - L(x)| \]
to be over a range of \(x\) including the \(x\) values we interpolated?

The following theorem answers this question.

**Theorem** If the function \(f(x)\) has \(n+1\) continuous derivatives on some interval \([a,b]\) and the polynomial \(P(x)\) is the Lagrange interpolating polynomial constructed to interpolate a set of points \((x_0, f(x_0)), (x_1, f(x_1)), \ldots, (x_n, f(x_n))\) with \(x_k \in [a,b]\) for all \(k\) then for each \(x\) in \([a,b]\) there is a \(\xi(x)\) such that

\[
f(x) = P(x) + \frac{f^{(n+1)}(\xi(x))}{(n+1)!} (x - x_0)(x - x_1)\cdots(x - x_n)
\]

**Proof** If \(x\) is any one of the points \(x_k\) we have that

\[
f(x_k) = P(x_k) + \frac{f^{(n+1)}(\xi(x_k))}{(n+1)!} (x_k - x_0)(x_k - x_1)\cdots(x_k - x_n)\cdots(x_k - x_n)
\]

or

\[
f(x_k) = f(x_k) + 0
\]

For \(x \neq x_k\) for any \(k\), we define a function

\[
g(t) = f(t) - P(t) \cdot [f(x) - P(x)] \frac{(t - x_0)(t - x_1)\cdots(t - x_n)}{(x - x_0)(x - x_1)\cdots(x - x_n)}
\]

Note that \(g(t)\) vanishes as each of the \(n+1\) points \(t = x_k\). By construction, \(g(t)\) also vanishes at \(t = x\). This means that \(g(t)\) vanishes at a total of \(n+2\) distinct points. Note also that \(g(t)\) has \(n+1\) continuous derivatives on the interval \([a,b]\). By the generalized version of Rolle's theorem, there must be a \(\xi(x)\) in \([a,b]\) such that

\[
g^{(n+1)}(\xi(x)) = 0
\]

or

\[
0 = g^{(n+1)}(\xi(x))
\]

\[
= f^{(n+1)}(\xi(x)) - P^{(n+1)}(\xi(x))
\]

\[
- [f(x) - P(x)] \frac{d^{n+1}}{dt^{n+1}} \left( \frac{(t - x_0)(t - x_1)\cdots(t - x_n)}{(x - x_0)(x - x_1)\cdots(x - x_n)} \right)_{t = \xi(x)}
\]

or

\[
0 = f^{(n+1)}(\xi(x)) - [f(x) - P(x)] \frac{(n+1)!}{(x - x_0)(x - x_1)\cdots(x - x_n)}
\]

Solving this equation for \(f(x)\) gives
\[ f(x) = P(x) + \frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x - x_0)(x - x_1)\cdots(x - x_n) \]