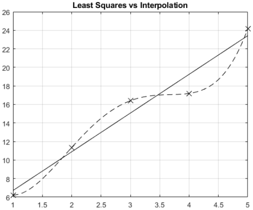
Interpolation

MATH-151: Mathematical Algorithms in Matlab

September 18, 2023

Fitting the Data

- Sometimes we are given some information and want to find a function that agrees with the data. This is a process called **interpolation**
  - Some of you have probably seen least-squares fits or regression before, that seeks to match the "trend" of the data, however with interpolation, we want to describe the functional relationship of the data **and** make sure all of our data points are included



- Most often we use polynomials as our **interpolants**, but different sets of functions can also be used!

WHEN IS INTERPOLATION USED?

- Interpolation is useful in many purposes when we are interested in representing data with a smooth function that captures changes in slope. For example
  - Plotting a smooth curve through data
  - Reading between entries of a data table
  - Differentiation or integration of function data
  - Approximating a complicated function with a simpler one
  - Using trigonometric functions to interpolate uniformly-spaced data points is called the Discrete Fourier Transform (or DFT) and commonly used in signal processing.

Lagrange Method

- Lagrange's method is centered around creating terms that equal $f(x_i)$ when $x = x_i$ and 0 when $x = x_j, j \neq i$.
- Let's start with the second part, how do we make our term be 0, when our x is any of our other points? We multiply by $(x - x_j)$ for all the $j \neq i$

$$\text{We write this } \prod_{j=1, j\neq i}^{N} (x - x_j)$$

- How do we make it correct when $x = x_i$? We take our product above and divide by $(x_i - x_j)$ for each $j \neq i$ so we get 1, then multiply by $f(x_i)$

$$\text{This gives us } f(x_i) \frac{\prod_{j=1, j\neq i}^{N} (x - x_j)}{\prod_{j=1, j\neq i}^{N} (x_i - x_j)}$$

- We repeat this for each of our points and add them together

$$p(x) = \sum_{i=1}^{N} f(x_i) \frac{\prod_{j=1, j\neq i}^{N} (x - x_j)}{\prod_{j=1, j\neq i}^{N} (x_i - x_j)}$$

Lagrange Method in Practice

- Lagrange's method looks a lot more complicated than it is. Let's see an example.
- Suppose we have 3 points. $(2, 5)$, $(7, 0)$, and $(8, 11)$.
- We start with $x_1 = 2, f(x_1) = 5$. Our term to make sure this works is

$$f(x_1)\frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} = 5\frac{(x - 7)(x - 8)}{(2 - 7)(2 - 8)}$$

- Repeating this process for $x_2 = 7$ and $x_3 = 8$ we can find our interpolating polynomial

$$p(x) = 5\frac{(x - 7)(x - 8)}{(2 - 7)(2 - 8)} + 0\frac{(x - 2)(x - 8)}{(7 - 2)(7 - 8)} + 11\frac{(x - 2)(x - 7)}{(8 - 2)(8 - 7)}$$

- We can simplify this, but we don't have to. The computer loves to grind through arithmetic!

Newton Divided Differences Method

- Another way that we can find our interpolating polynomial with less computational cost is to use **Newton's divided differences** method
- The divided differences are denoted by $f[]$ and defined using a recursive definition

$$f[x_1, x_2, \dots, x_n] = \frac{f[x_2, \dots, x_n] - f[x_1, \dots, x_{n-1}]}{x_n - x_1}$$

  where we start with base cases $f[x_n] = f(x_n)$
- And we use these to build our polynomial as follows

$$p(x) = f[x_1] + f[x_1, x_2](x - x_1) + \cdots + f[x_1, \dots, x_n](x - x_1) \times \cdots \times (x - x_{n-1})$$

- Just like Lagrange's method, this looks much more complicated when writing it abstractly than when performing it in practice, so lets see an example again..

Newton Divided Differences in Practice

- Using our points from earlier we can compute our interpolating polynomial using our divided differences.

x	$f(x)$		
2	$f[x_1] = 5$		
		$f[x_1, x_2] = \frac{0-5}{7-2} = -1$	
7	$f[x_2] = 0$		$f[x_1, x_2, x_3] = \frac{11-(-1)}{8-2} = 2$
		$f[x_2, x_3] = \frac{11-0}{8-7} = 11$	
8	$f[x_3] = 11$		

- Putting it all together we can find our interpolating polynomial is

$$f(x) = 5 + (-1)(x-2) + 2(x-2)(x-7)$$

or, if we want to be fancy, $f(x) = 5 + (x-2)\Big(-1 + (x-7)(2)\Big)$

- If we want to do the work, we can see this is the same answer that Lagrange's method gave us!

Why Use Each Method?

- Both of these methods get us to the same answer, so why show both of them?
- Each method has its own strengths and weaknesses.
  - Lagrange's method is easier to understand and more straightfoward. But it is less flexible to new data, and often less computationally efficient
  - Newton's divided differences are easier to add new data points to and can be written to be very fast, however it is harder to understand and implement