Polynomial interpolation

(using a PDF of Janet Peterson)

If we have a finite set of points

$$(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$$

we want to construct a polynomial that represents the points and also the points between the known points. this polynomial approximates the unknown function that generated the data points. We could use also polynomials to approximate more complicated functions f(x).

In calculus, when you constructed a Taylor polynomial this was a polynomial which approximated another function.

The way we ask the polynomial to represent our data results in different polynomials.

The problem we are going to consider initially is to find an interpolating polynomial for a set of data (x_i, y_i):

Given $n+1$ distinct points

$$(x_1, y_1), (x_2, y_2), \cdots (x_{n+1}, y_{n+1})$$

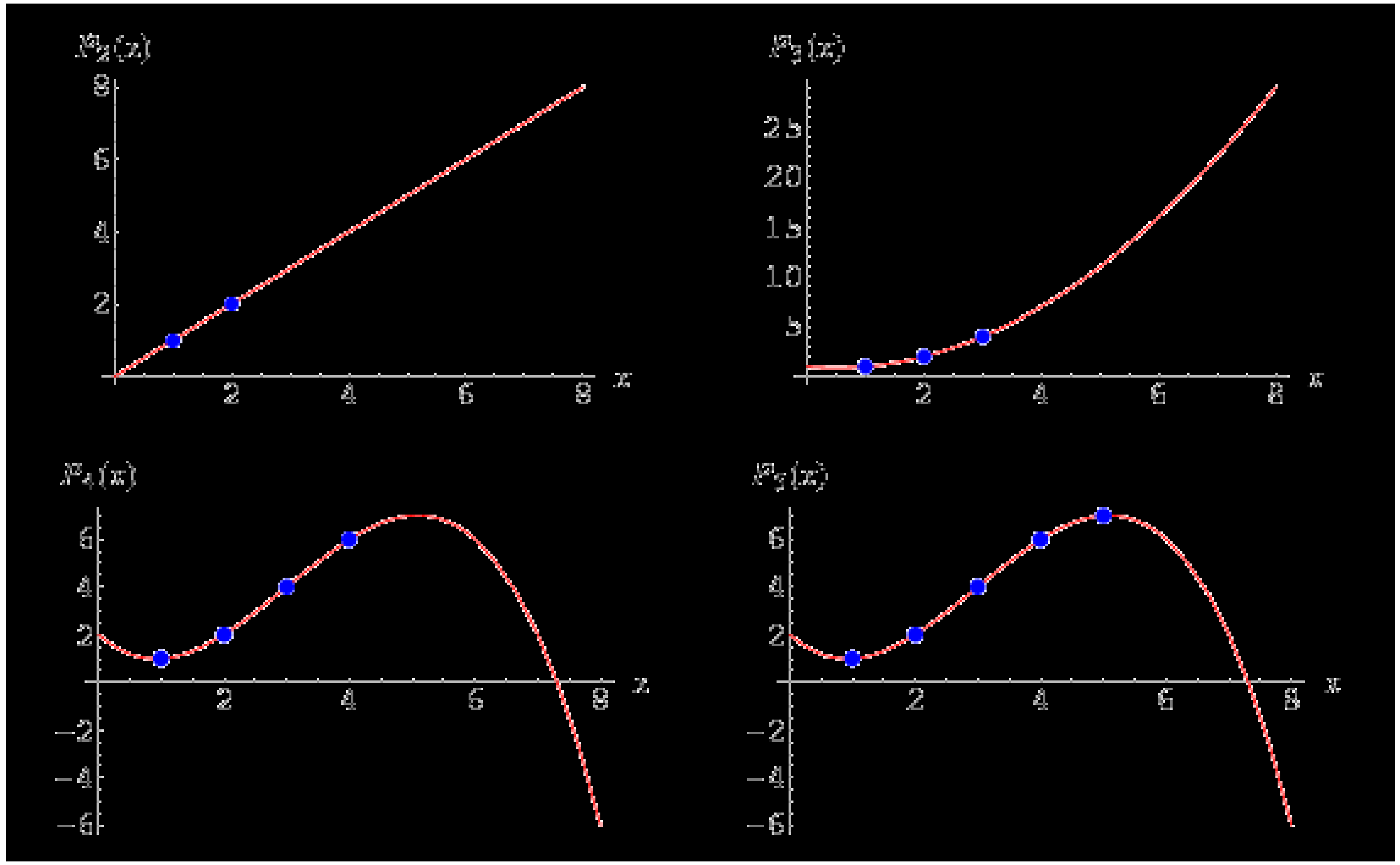
find a polynomial of degree n, say p_n, such that

$$p_n(x_i) = y_i, \quad \text{for } i = 1, 2, \ldots, n+1$$

- This means that the graph of the polynomial passes through our $n+1$ distinct data points.
- The general polynomial of degree n looks like

$$a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1} + a_n x^n$$

- It can be proved that there is a unique polynomial which satisfies our problem.



- In the first figure we have two data points so we find the linear polynomial which passes through these two points.
- In the second figure we have three data points so we find the quadratic polynomial which passes through these three points.
- In the third and fourth figures we have four and five points so we find a cubic and a quartic, respectively.

What do we want to accomplish?

1. First, we want to determine an efficient way to determine the polynomial, i.e., determine the coefficients $a_0, a_1, \cdots, a_n$.

2. Once we have determined the coefficients, we want to determine how to efficiently evaluate the polynomial at a point.

3. If we decide to add points to our data set, we would like to not have to start our calculation from scratch. Since the polynomial is unique we choose to evaluate it any way that works and is efficient.

4. We want to view this problem as being analogous to finding a polynomial which interpolates a given function at a set $\{x_i\}$.

5. We want to decide if this is a good approach to interpolating a set of data or a function. If not, we want to look at alternative ways to interpolate.

Some Simple Examples

- As a first example, find the linear polynomial (i.e., p_1) which passes through the two points $(1, -5)$, $(5, 3)$. Clearly if p_1 must pass through these two points we have

$$p_1(1) = -5 \qquad p_1(5) = 3$$

So if $p_1 = a_0 + a_1 x$ we must satisfy

$$a_0 + a_1(1) = -5 \qquad a_0 + 5a_1 = 3$$

Solving these two equations we get $a_0 = -7$ and $a_1 = 2$ so we have the polynomial $p_1 = 2x - 7$.

- As a second example, find the quadratic that passes through the 3 points $(1, 10)$, (2, 19), (-1,-14). Clearly if $p_2 = a_0 + a_1 x + a_2 x^2$ must pass through these three points we have

$$p_2(1) = 10 \qquad p_2(2) = 19 \qquad p_2(-1) = -14$$

or equivalently we must satisfy

$$a_0 + a_1(1) + a_2(1)^2 = 10 \quad a_0 + 2 \cdot a_1 + (2)^2 a_2 = 19 \quad a_0 + a_1(-1) + a_2(-1)^2 = -14$$

Solving these three equations we get the polynomial $p_2 = -x^2 + 12x - 1$.

- As you can see, if we proceed in this manner then to find a polynomial of degree n that passes through the given $n+1$ points we have to solve $n+1$ linear equations for the $n+1$ unknowns $a_0, a_1, \ldots, a_n$.

- However, there are more efficient ways to do this.

- There are basically two main efficient approaches for determining the interpolating polynomial. Each has advantages in certain circumstances.

- Since we are guaranteed that we will get the same polynomial using different approaches (since the polynomial is unique) it is just a question of which is a more efficient implementation.

Lagrange Form of the Interpolating Polynomial

- To motivate this form of the interpolating polynomial, let's revisit one of our examples.
- We found that the quadratic interpolating polynomial which interpolates the points $(1, 10)$, (2, 19), (-1,-14) is $p_2 = -x^2 + 12x - 1$.
- Instead of writing this polynomial in terms of the monomials $1, x, x^2$ (i.e., $p = a_0 \cdot 1 + a_1 x + a_2 x^2$) lets rewrite it as

$$p_2 = 10 * L_1(x) + 19 L_2(x) - 14 L_3(x)$$

where $L_i(x)$, $i = 1, 3$ are quadratic polynomials which have the properties

$$L_1(2) = L_1(-1) = 0, \qquad L_1(1) = 1$$

$$L_2(1) = L_2(-1) = 0, \qquad L_2(2) = 1$$

$$L_3(1) = L_3(2) = 0, \qquad L_3(-1) = 1$$

If we can do this, then clearly $p_2(1) = 10$, $p_2(2) = 19$ and $p_2(-1) = -14$ and once we have the $L_i(x)$ we have found our interpolating polynomial, just not reduced to its simplest form.

- At first it seems that we have traded finding one quadratic polynomial for finding 3 others. However, let's look at what $L_i(x)$, $i = 1, 3$ are in our example.

$$L_1(x) = \frac{(x-2)(x+1)}{(1-2)(1+1)} = \frac{(x-2)(x+1)}{-2}$$

- Clearly because we have the factor $(x-2)$ in the numerator, $L_1(2) = 0$. Similarly the factor $(x+1)$ in the numerator makes $L_1(-1) = 0$. When we evaluate the numerator at $x = 1$ we get the denominator which is just -2 so it satisfies $L_1(1) =$ 1

- Similarly

$$L_2(x) = \frac{(x-1)(x+1)}{(2-1)(2+1)} = \frac{(x-1)(x+1)}{3}$$

and

$$L_3(x) = \frac{(x-1)(x-2)}{(-1-1)(-1-2)} = \frac{(x-1)(x-2)}{6}$$

- The numerator is easily determined because for L_i we simply use factors of $(x - x_j)$ for $j \neq i$.
- How do we get the denominator? Because we want $L_i(x_i) = 1$, we simply choose the denominator to equal the numerator evaluated at x_i.
- For our problem, we can show that upon simplification we get the same polynomial as before $-x^2 + 12x - 1$

$$p_2 = 10\frac{(x-2)(x+1)}{-2}) + 19\frac{(x-1)(x+1)}{3} - 14\frac{(x-1)(x-2)}{6}$$

$$p_2 = \big(-5x^2 + 5x + 10\big) + \frac{19}{3}\big(x^2 - 1\big) - \frac{7}{3}\big(x^2 - 3x + 2\big)$$

$$p_2 = \Big(\frac{-15+19-7}{3}\Big)x^2 + \Big(\frac{15+21}{3}\Big)x + \Big(\frac{30-19-14}{3}\Big) = -x^2 + 12x - 1$$

Given $n+1$ distinct points

$$(x_1, y_1), (x_2, y_2), \cdots (x_{n+1}, y_{n+1})$$

The Lagrange form of the interpolating polynomial is

$$p_n = \sum_{i=1}^{n+1} y_i L_i(x)\,,$$

where

$$L_i(x) = \frac{(x-x_1)(x-x_2)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_{n+1})}{(x_i-x_1)(x_i-x_2)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_{n+1})}$$

The important properties of $L_i(x)$ are

1. $L_i(x_j) = 0$ for $j \neq i$
2. $L_i(x_i) = 1$

Newton Form of the Interpolating Polynomial

- This approach to determining the interpolating polynomial using divided differences and allows adding data without starting over.
- When we first looked at writing an interpolating polynomial we wrote it as $c_1 + c_2 x + c_3 x^2 + \cdots c_{n+1} x^n$ in terms of the monomials $1, x, x^2, \ldots, x^n$. This required the solution of $n+1$ linear equations.
- Recall that when we wrote the Lagrange form of say the third degree interpolating polynomial we wrote it in terms of sums of cubic polynomials. The Newton form takes a different approach.
- The Newton form of the line passing through (x_1, y_1), (x_2, y_2) is
$$p_1(x) = a_1 + a_2(x - x_1)$$
- The Newton form of the parabola passing through (x_1, y_1), (x_2, y_2), (x_3, y_3) is
$$p_2(x) = a_1 + a_2(x - x_1) + a_3(x - x_1)(x - x_2)$$

- The general form of the Newton polynomial passing through (x_i, y_i), $i = 1, \ldots, n+1$ is

  $$p_n(x) = a_1 + a_2(x - x_1) + a_3(x - x_1)(x - x_2) + \cdots + a_{n+1}(x - x_1)(x - x_2) \cdot (x - x_n)$$

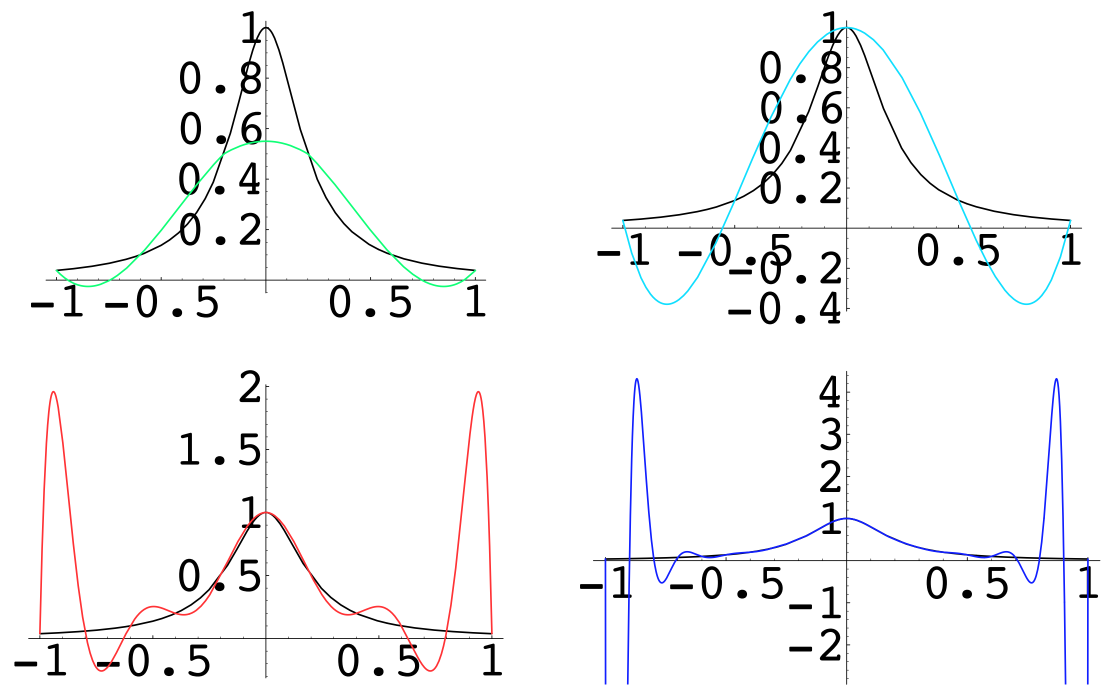
- As in the case of the Lagrange form of the interpolating polynomial we first find the coefficients a_i, $i = 1, \ldots, n = 1$ and then we use this formula to evaluation p_n at a point.

Runge's Example

- Consider the function

$$f(x) = \frac{1}{1+25x^2} \qquad -1 \le x \le 1$$

which we want to interpolate (using only function values) with an increasing number of points. We interpolate with evenly spaced points. The black curve represents the function.



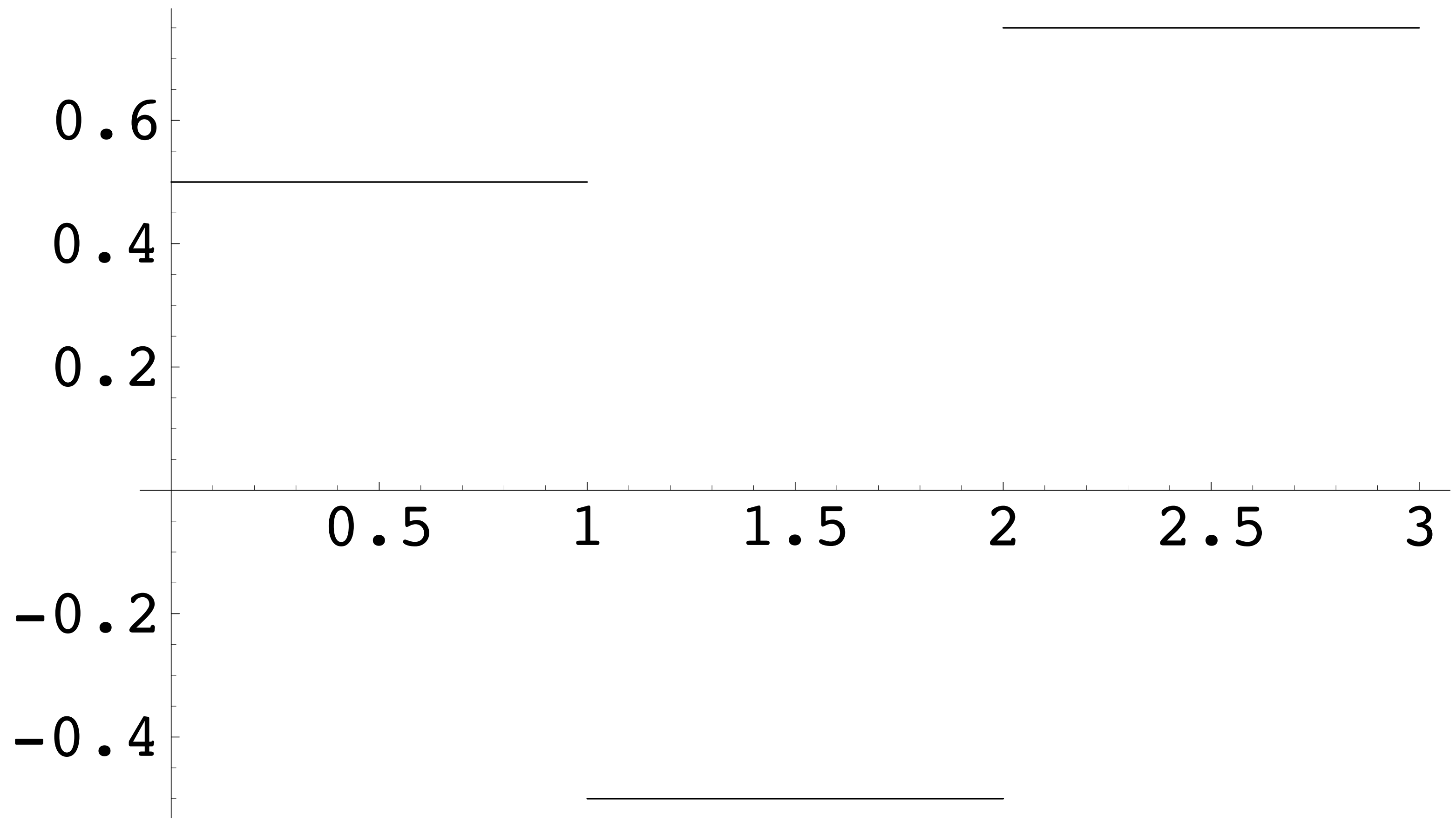
- As you can see, the interpolant gets more "wiggles" in it as it is required to interpolate more points.
- It is for this reason that one should not use a single polynomial to interpolate a lot of data.
- The general rule is that

  HIGH DEGREE POLYNOMIAL INTERPOLATION SHOULD BE AVOIDED

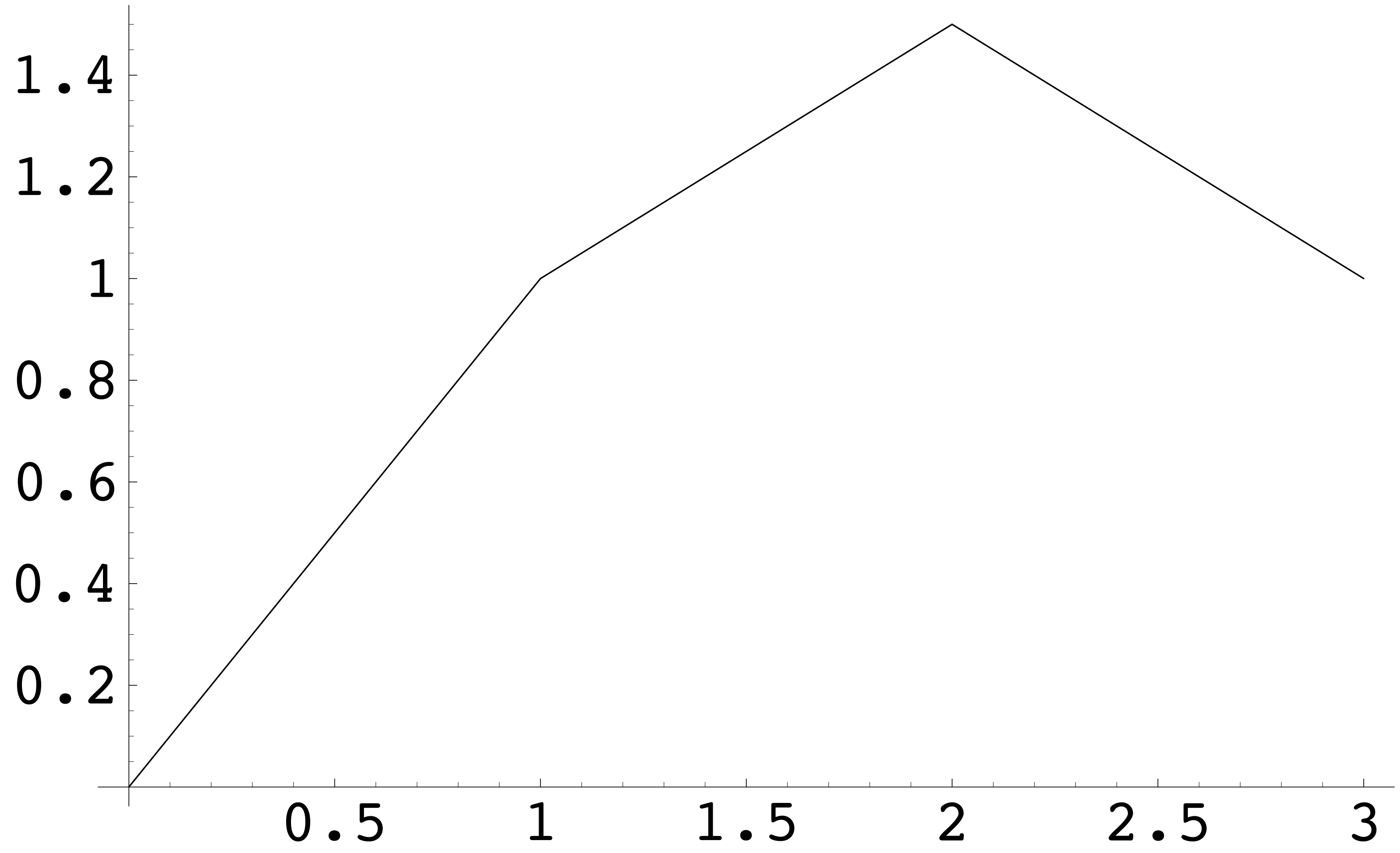
- What can we do instead?
- We use something called piecewise interpolation.
- When a graphing program plots a function like we did in the previous examples it connects closely spaced points with a straight line. If the points are close enough, then the result looks like a curve to our eye. This is called piecewise linear interpolation.

Examples of Piecewise Functions

Piecewise Constant Function

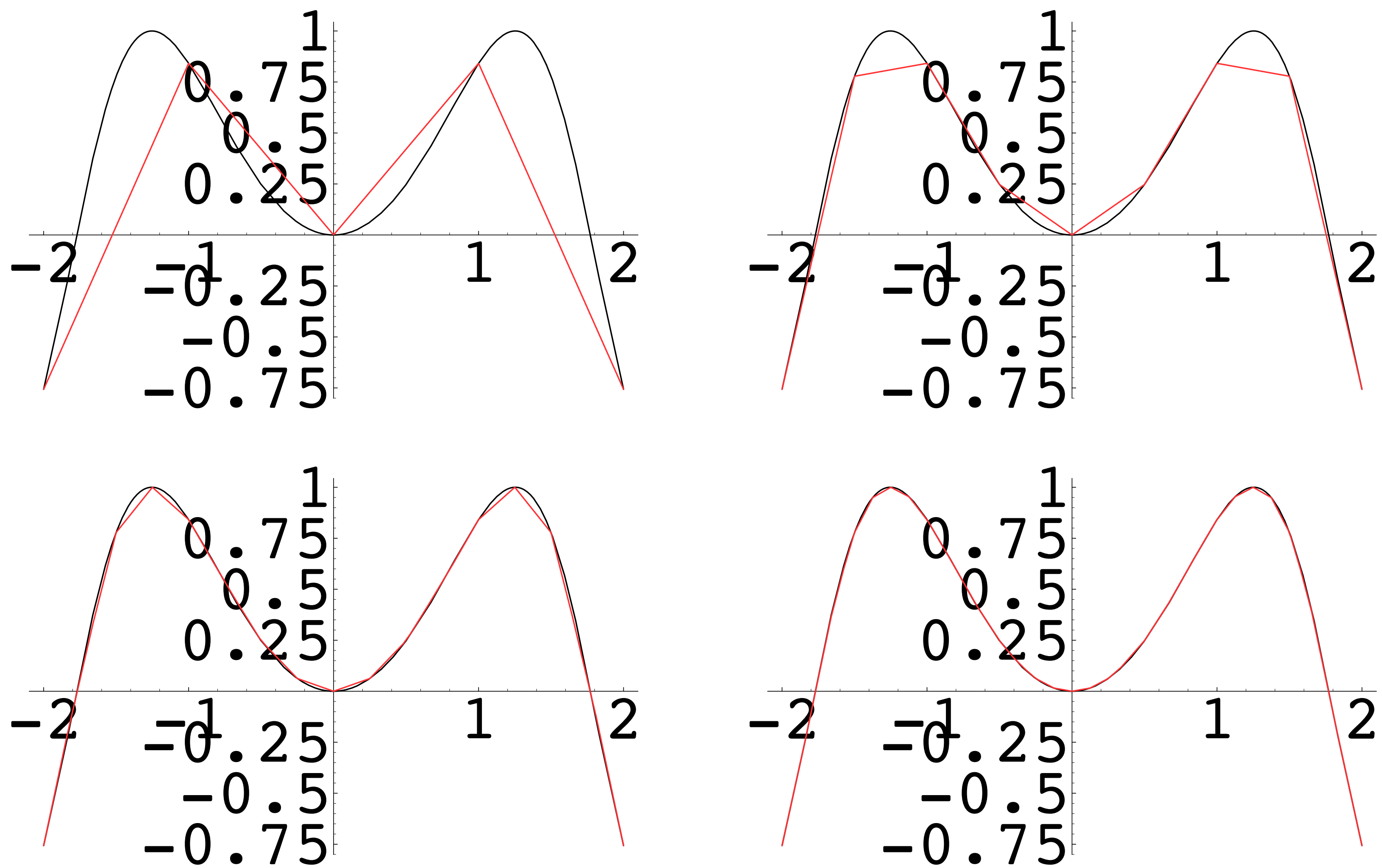


Continuous Piecewise Linear Function

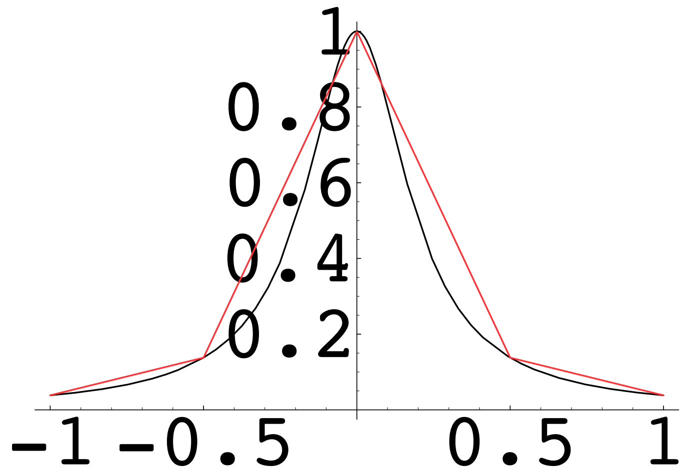


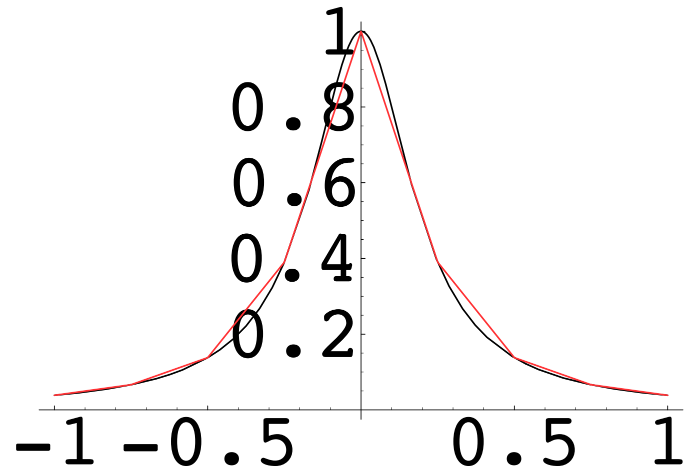
We will be interpolating data using a piecewise linear function on each subinterval so the linear function defined on the interval $[x_{i-1}, x_i]$ and the function defined on $[x_i, x_{i+1}]$ will match up at x_i; i.e., we are using continuous, piecewise linear functions.

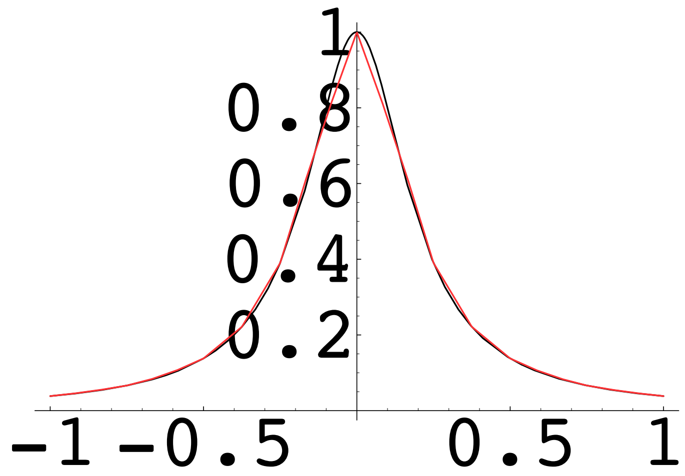
Below we see some examples of a piecewise linear interpolant to $\sin x^2$ on (-2,2). In each plot we are using uniform subintervals and doubling the number of subintervals from one plot to the next.



Below we see some examples of piecewise linear interpolation for Runge's example. Note that this form of interpolation does not have the "wiggles" we encountered by using a higher degree polynomial.

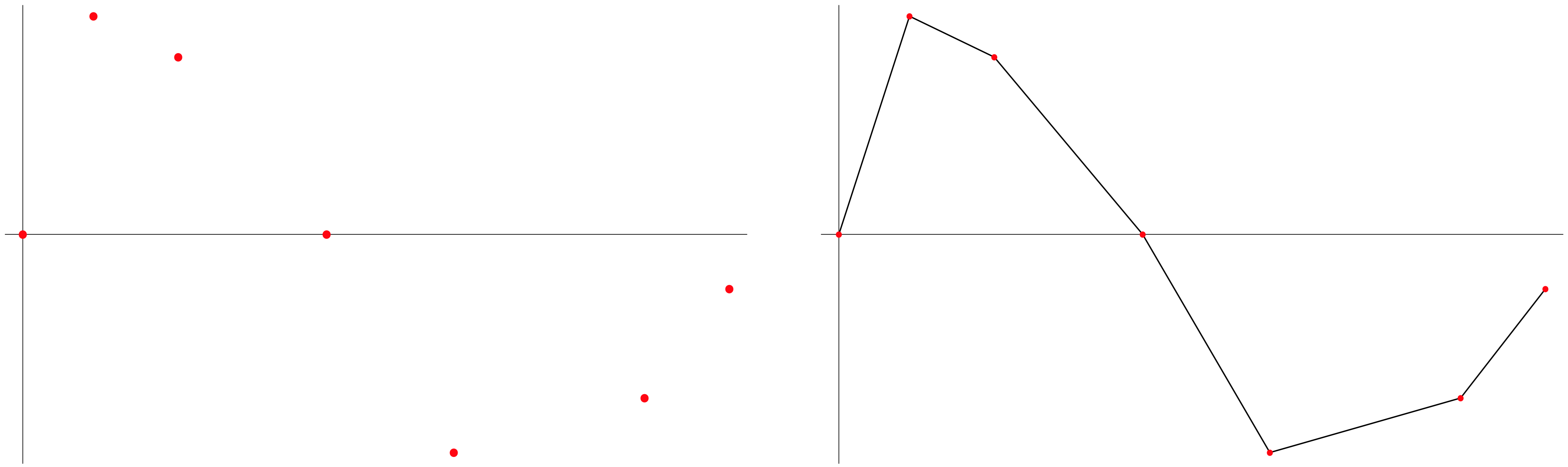


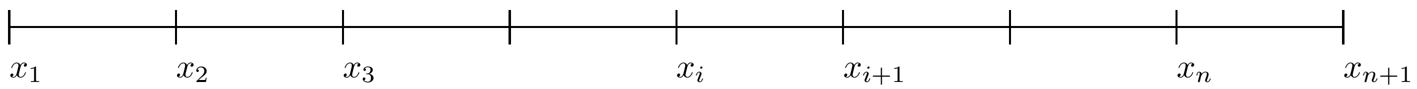




Piecewise Linear Interpolation

- Assume that we are given (x_1, y_1), (x_2, y_2), ..., (x_{n+1}, y_{n+1}). We want to construct the piecewise linear interpolant.





- We divide our domain into points x_i $i = 1, \ldots, n+1$
- The linear interpolation on the interval $[x_i, x_{i+1}]$ is just

$$\mathcal{L}_i(x) = a_i + b_i(x - x_i)$$

where

$$a_i = y_i \qquad y_{i+1} = y_i + b_i(x_{i+1} - x_i) \Rightarrow b_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}$$

- Thus our piecewise linear interpolant is given by

$$\mathcal{L}(x) = \begin{cases} \mathcal{L}_1(x) & \text{if } x_1 \leq x \leq x_2 \\ \mathcal{L}_2(x) & \text{if } x_2 \leq x \leq x_3 \\ \vdots & \vdots \\ \mathcal{L}_i(x) & \text{if } x_i \leq x \leq x_{i+1} \\ \vdots & \vdots \\ \mathcal{L}_n(x) & \text{if } x_n \leq x \leq x_{n+1} \end{cases}$$

- Note that $\mathcal{L}(x)$ is continuous but not differentiable everywhere.

Example

Determine the piecewise linear interpolant to $\sin x$ on $[0, \pi]$ found by using 2 equal subintervals; then evaluate it at the points $\frac{\pi}{3}, \frac{3\pi}{5}$. Compute error.

- Our intervals are $[0, \frac{\pi}{2}]$ and $[\frac{\pi}{2}, \pi]$.
- On the first interval the coefficients are

$$a_1 = \sin 0 = 0 \qquad b_1 = \frac{\sin \frac{\pi}{2} - \sin 0}{\frac{\pi}{2} - 0} = \frac{2}{\pi}$$

  so we have $\mathcal{L}_1 = 0 + \dfrac{2}{\pi}(x - 0) = \dfrac{2x}{\pi}$.

- On the second interval the coefficients are

$$a_2 = \sin \frac{\pi}{2} = 1 \qquad b_2 = \frac{\sin \pi - \sin \frac{\pi}{2}}{\pi - \frac{\pi}{2}} = \frac{0 - 1}{\frac{\pi}{2}} = -\frac{2}{\pi}$$

which gives $\mathcal{L}_2 = 1 - \dfrac{2}{\pi}(x - \dfrac{\pi}{2})$.

- The point $\frac{\pi}{3}$ is in the first interval so

$$\mathcal{L}_1(x) = a_1 + b_1(x - x_1) \Rightarrow \mathcal{L}_1(\frac{\pi}{3}) = 0 + \frac{2}{\pi}(\frac{\pi}{3} - 0) = \frac{2}{3}$$

  The actual value of $\sin\frac{\pi}{3}$ is 0.866025 so our error is approximately 0.2. The error is fairly large because the length of our subinterval is large $\frac{\pi}{2} \approx 1.57$.

- Now the point $\frac{3\pi}{5}$ is in the second interval so

$$\mathcal{L}_2(x) = a_2 + b_2(x - x_2) \Rightarrow \mathcal{L}_2(\frac{3\pi}{5}) = 1 - \frac{2}{\pi}(\frac{3\pi}{5} - \frac{\pi}{2}) = \frac{4}{5}$$

  The actual value of $\sin\frac{3\pi}{5}$ is 0.950157 so our error is approximately 0.151.

Implementing Piecewise Linear Interpolation

- Given the data (x_i, y_i), $i = 1, 2, \ldots, n+1$, we can determine the coefficients a_i, b_i, $i = 1, \ldots, n$ on each of the n subintervals by the formulas

$$a_i = y_i \qquad b_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}, \quad i = 1, 2, \ldots, n$$

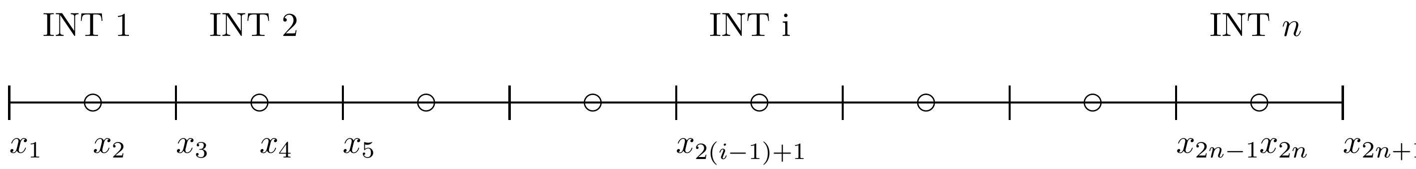
  This can be done once and stored just like we did for the nth degree interpolating polynomial.

- Unlike the case when we had a single nth degree polynomial, when we want to evaluate our piecewise linear interpolant at some point x then we have to decide which formula $\mathcal{L}_i(x)$ we need to use. That is, we have to decide the subinterval $[x_i, x_{i+1}]$ so that $x \in [x_i, x_{i+1}]$. Once we do this, then we simply use the formula $\qquad a_i + b_i(x - x_i) \qquad$ where our coefficients a_i, b_i are known.

- Before we look at the actual implementation, lets look at piecewise quadratic interpolation because it brings up some additional issues.

Piecewise Quadratic Interpolation

- We know that to determine the quadratic interpolation on an interval we need 3 conditions so we have to add another point.
- This is possible to do if we are approximating a given function by this piecewise quadratic function since we could just choose the midpoint.
- However, if we have discrete data, we might not be able to add another point. In that case we must have an even number of intervals and then we choose the intervals $[x_i, x_{i+2}]$ to perform the quadratic interpolation.
- For simplicity we will take the approach that we are going to approximate a given function so that we can generate another point in the interval and we choose the midpoint.



- We assume that we have n intervals and $2n+1$ points labeled x_i, $i = 1, \ldots, 2n+1$. For example, the intervals are

$$[x_1, x_3], \quad [x_3, x_5], \quad \cdots \quad [x_{2n-1}, x_{2n+1}]$$

- Then using the Newton form of the interpolating polynomial we have for the interval $[x_1, x_3]$ with x_2 the midpoint

$$\mathcal{Q}_1(x) = a_1 + b_1(x - x_1) + c_1(x - x_1)(x - x_2)$$

so when we evaluate $\mathcal{Q}_1(x_1)$, $\mathcal{Q}_1(x_2)$, and $\mathcal{Q}_1(x_3)$ we get

$$\mathcal{Q}_1(x_1) = y_1 \Rightarrow a_1 = y_1$$

$$\mathcal{Q}_1(x_2) = y_2 \Rightarrow y_2 = y_1 + b_1(x_2 - x_1) \Rightarrow b_1 = \frac{y_2 - y_1}{x_2 - x_1} = 2\frac{y_2 - y_1}{\Delta x_1}$$

$$\mathcal{Q}_1(x_3) = y_3 \Rightarrow y_3 = y_1 + b_1 \Delta x_1 + c_1 \Delta x_1 \left(\frac{\Delta x_1}{2}\right) \Rightarrow c_1 = 2\frac{y_3 - y_1 - b_1 \Delta x_1}{\Delta x_1^2}$$

where $\Delta x_1 = x_3 - x_1$.

- Thus our piecewise quadratic interpolant on the entire region $[x_1, x_{2n+1}]$ is given by a quadratic on each of the n intervals:

$$\mathcal{Q}(x) = \begin{cases} \mathcal{Q}_1(x) & \text{if } x_1 \le x \le x_3 \\ \mathcal{Q}_2(x) & \text{if } x_3 \le x \le x_5 \\ \quad\vdots & \qquad\vdots \\ \mathcal{Q}_i(x) & \text{if } x_i \le x \le x_{i+2} \\ \quad\vdots & \qquad\vdots \\ \mathcal{Q}_n(x) & \text{if } x_{2n-1} \le x \le x_{2n+1} \end{cases}$$

- Note that as in the case of the piecewise linear interpolant, $\mathcal{Q}(x)$ is continuous everywhere but not differentiable everywhere.
- So one thing that is different from the linear case is the relationship between the number of intervals and the number of points.