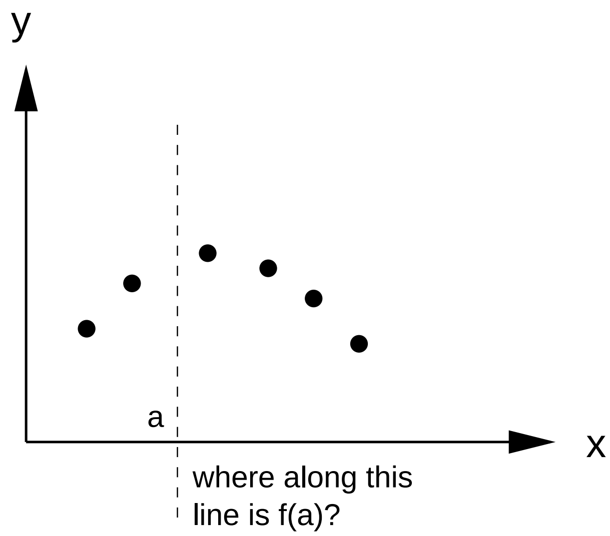
1 INTERPOLATION

1.1 Statement of the Problem

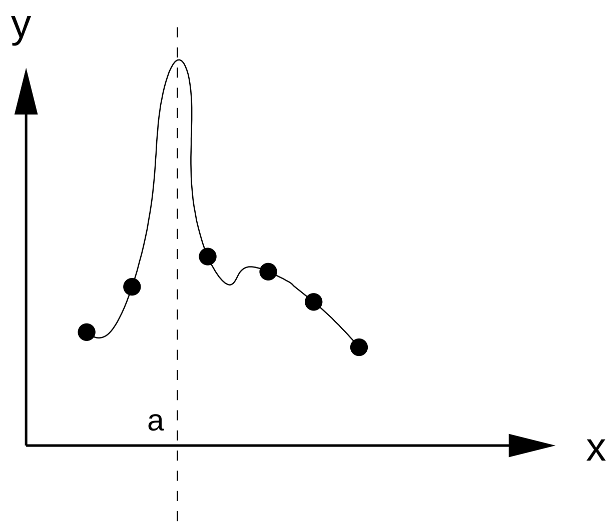
The first subsection in each section of these notes is a description or "statement" of the problem that is going to be solved in that section. Here, in the first section, we are discussing *interpolating* some data points.

Suppose some function $f(x)$ is only known at a discrete set of points x_i where $i = 1, 2, \ldots N$ as shown in the figure below. The problem is to find $f(x)$ at all values of x.



The strict answer to the question *What is the value of $f(a)$?* is that **YOU DON'T KNOW!!**

i.e. unless you know the function that the data points go through, there is nothing stopping a situation like this happening:

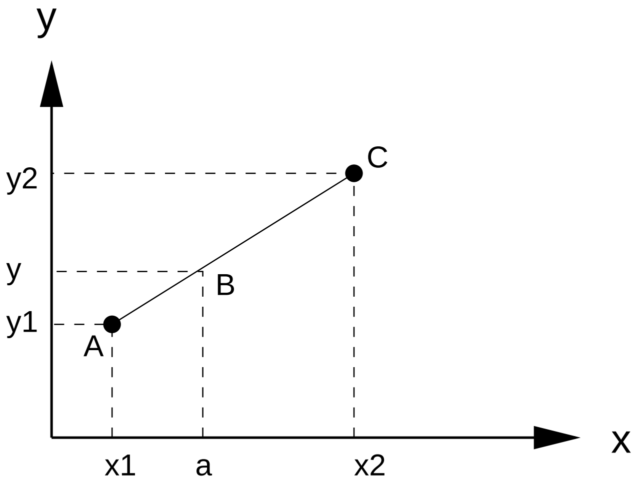


i.e. a completely unexpected thing happening.

If you do happen to know the function that goes through the data points, then you can use the **Curve Fitting** techniques that will be described later in Section 5.

1.2 Linear Interpolation

Below is a graphical summary of the *Linear Interpolation* method:



The coordinates (x,y) are given by

$$\frac{y_2 - y}{x_2 - x} = \frac{y_2 - y_1}{x_2 - x_1}$$

which is just saying that the gradient between B & C is equal to the gradient between A & C. Re-arranging, this gives

$$\begin{aligned} y &= \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x - x_2) + y_2 \\ &= y_1 \left(\frac{x - x_2}{x_1 - x_2}\right) + y_2 \left(\frac{x - x_1}{x_2 - x_1}\right) \\ &= y_1 \, A_1(x) \, + \, y_2 \, A_2(x) \end{aligned} \tag{1}$$

(You should check that you can derive this!) Note that the functions $A_1(x)$ and $A_2(x)$ have special properties! (See later!) They are defined

$$\begin{aligned} A_1(x) &= \frac{x - x_2}{x_1 - x_2} \\ A_2(x) &= \frac{x - x_1}{x_2 - x_1} \end{aligned}$$

There is another way of performing a linear interpolation using the **Taylor Expansion**:

$$f(x) = f(x_1) + (x - x_1) f'(x_1)$$

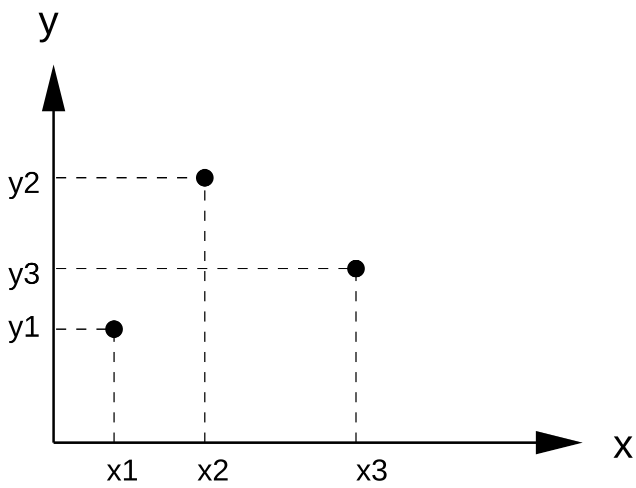
or, in other words, since $f'(x_1)$ is just the gradient of the straight line AC, i.e. $f'(x_1) = \frac{y_2 - y_1}{x_2 - x_1}$, we have

$$y = y_1 + (x - x_1)\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \tag{2}$$

Note that eq(1) **must** be the same as eq(2). (There will be a problem sheet question about this later...)

1.3 Quadratic (i.e. 2nd Order) Interpolation

In order to perform a *Quadratic Interpolation* we'll need 3 points! One way of seeing this is that a quadratic function has *three* coefficients, and to define these three coefficients unambiguously we're going to need three points!



Lagrange's answer is:

$$y = y_1B_1(x) + y_2B_2(x) + y_3B_3(x)$$

$$\begin{array}{rrcll} \text{where} & B_1(x) & = & 1 & \text{when} \quad x = x_1 \\ \text{and} & B_1(x) & = & 0 & \text{when} \quad x = x_2 \text{ or } x = x_3 \end{array}$$

Similarly for $B_2(x)$ and $B_3(x)$.

Recall from eq(1):

$$A_1(x) = \frac{x - x_2}{x_1 - x_2}$$

for the linear interpolation in sec. 1.2.

We can generalise this for quadratics as follows:

$$
\begin{aligned}
B_1(x) &= \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} \\
B_2(x) &= \frac{(x-x_3)(x-x_1)}{(x_2-x_3)(x_2-x_1)} \\
B_3(x) &= \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}
\end{aligned}
$$

Note that these B's have the important property that $B_i(x_i) = 1$ and $B_i(x_j) = 0$ (where $i \neq j$) as required. In other words, for example $B_2(x_1) = 0$, $B_3(x_3) = 1$ etc.

These are the same properties as the $A_1(x)$ and $A_2(x)$ in sec 1.2 have.

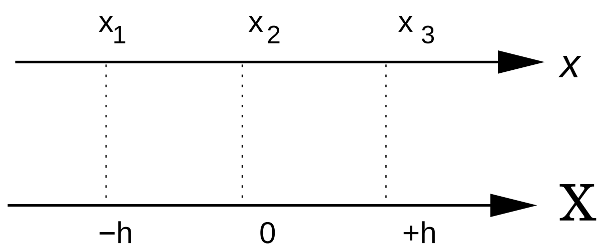
So, the full expression is

$$
\begin{aligned}
y &= y_1 \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} \\
&+ y_2 \frac{(x-x_3)(x-x_1)}{(x_2-x_3)(x_2-x_1)} \\
&+ y_3 \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)} \qquad (3)
\end{aligned}
$$

This is a polynomial of 2nd order (i.e. it's a **quadratic!**) which passes through all 3 points.

$\longrightarrow$ it's the required answer!

Now, let's redefine the x-axis:



As you can see, we've simply

- introduced a new $X-$axis which has an origin at the old $x = x_2$ position.
- assumed that x_1, x_2 and x_3 are all equi-spaced with spacing h.

Re-writing eq(3) in terms of the new X variable, we get

$$
\begin{aligned}
y &= y_1 \frac{X(X-h)}{-h \times -2h} + y_2 \frac{(X+h)(X-h)}{h \times -h} + y_3 \frac{(X+h)X}{2h \times h} \\
&= \frac{X^2}{h^2}(\frac{1}{2}y_1 - y_2 + \frac{1}{2}y_3) + \frac{X}{h}(-\frac{1}{2}y_1 + \frac{1}{2}y_3) + y_2 \qquad (4)
\end{aligned}
$$

You should check that you can derive this equation!

Equation(4) will be relevant for section 2.3 (2nd order numerical integration).

1.4 Higher Order Polynomials

The next natural thing to do after using linear and quadratic polynomials to interpolate functions is to proceed to higher order polynomials. Naturally, this requires adding more points as can be seen from the following table.

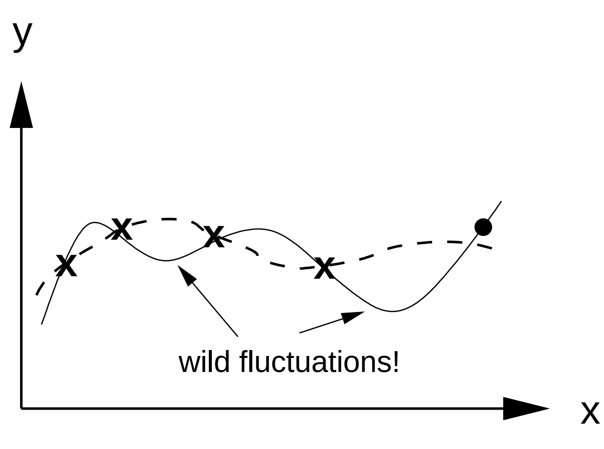
Order of Polynomial	Polynomial	Number of Coefficients in polynomial	Number of points required for interpolation
1 (i.e. linear)	$a_0 + a_1 x$	2	2
2 (i.e. quadratic)	$a_0 + a_1 x + a_2 x^2$	3	3
3 (i.e. cubic)	$a_0 + a_1 x + a_2 x^2 + a_3 x^3$	4	4
4 (i.e. quartic)	$a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4$	5	5
5 (i.e. quintic)	$a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5$	6	6

While it may seem that going to higher order polynomials will lead to a more accurate interpolation function, the reverse can be the case.

The problem is that the larger the order of the interpolating polynomial the wilder the fluctuations can be. This is depicted in the graph below.

The dashed line is a representation of a cubic (i.e. 3rd order) interpolation to the four points marked with a cross. When a quartic (i.e. 4th order) interpolation is performed to those four points plus an additional point marked with a filled circle, you get the solid line. This quartic interpolation formula, while going through all 5 points, has wild fluctuations which are very unreasonable.

An example VisualBasic programme which produces similar results appears in the **Consolidator** at the end of this section.

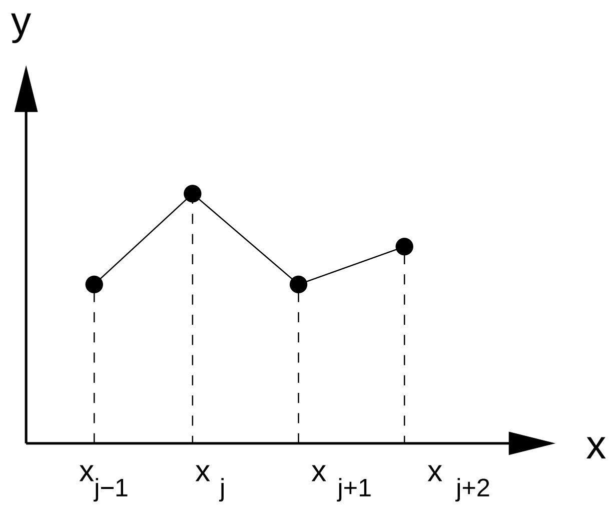


Note also, that the 5th point is to the right of the original four points (which are marked with a cross). Nevertheless, the quartic interpolation is very different to the cubic interpolation in this region! i.e. going to a higher order interpolation by adding an extra point can affect the interpolation even a long way from the extra point!

This motivates the need for a **local** algorithm which does not suffer from this problem. This is discussed in the next subsection.

1.5 Cubic Spline

Let's consider the set of points depicted in the following graph.



Now let's focus on the interval (x_j, x_{j+1}). Consider joining $x_j \to x_{j+1}$ with a line. This is just applying the linear interpolation formulas eq(1 or 2) locally.

Obviously, this has discontinuous 1st derivatives at x_j and x_{j+1}. This just means that the gradients (or first derivatives) of the two line segments that join at x_j are not equal. Similarly for the two line segments that join at x_{j+1}.

We want to improve on this.

AIM: Find a cubic polynomial for each interval (x_j, x_{j+1}) such that

1. it goes through points (x_j, y_j) and (x_{j+1}, y_{j+1})
2. has smooth 1st derivative
3. has continuous 2nd derivative

If we can find such a function, then it will be a very good interpolation formula.

Such a function exists which is called the **cubic spline**.

It is a very commonly used interpolation procedure - applications such as Excel have the cubic spline built in (it appears as one of the menu items!).

The cubic spline formula are quite complex, and since they are often built-in to the software that you'll be using, I won't detail them here.

1.6 Extrapolation

Extrapolation is estimating the value of a function beyond the region where the known data are! This is even more dangerous than interpolation!

The basic idea is to

1. interpolate the data first $\rightarrow$ *function* which goes through the data points (using one of the methods discussed above)

2. evaluate this function beyond the last known data point!

Just as in the case of interpolation, extrapolation becomes much more safe when the functional form which describes the data is known.