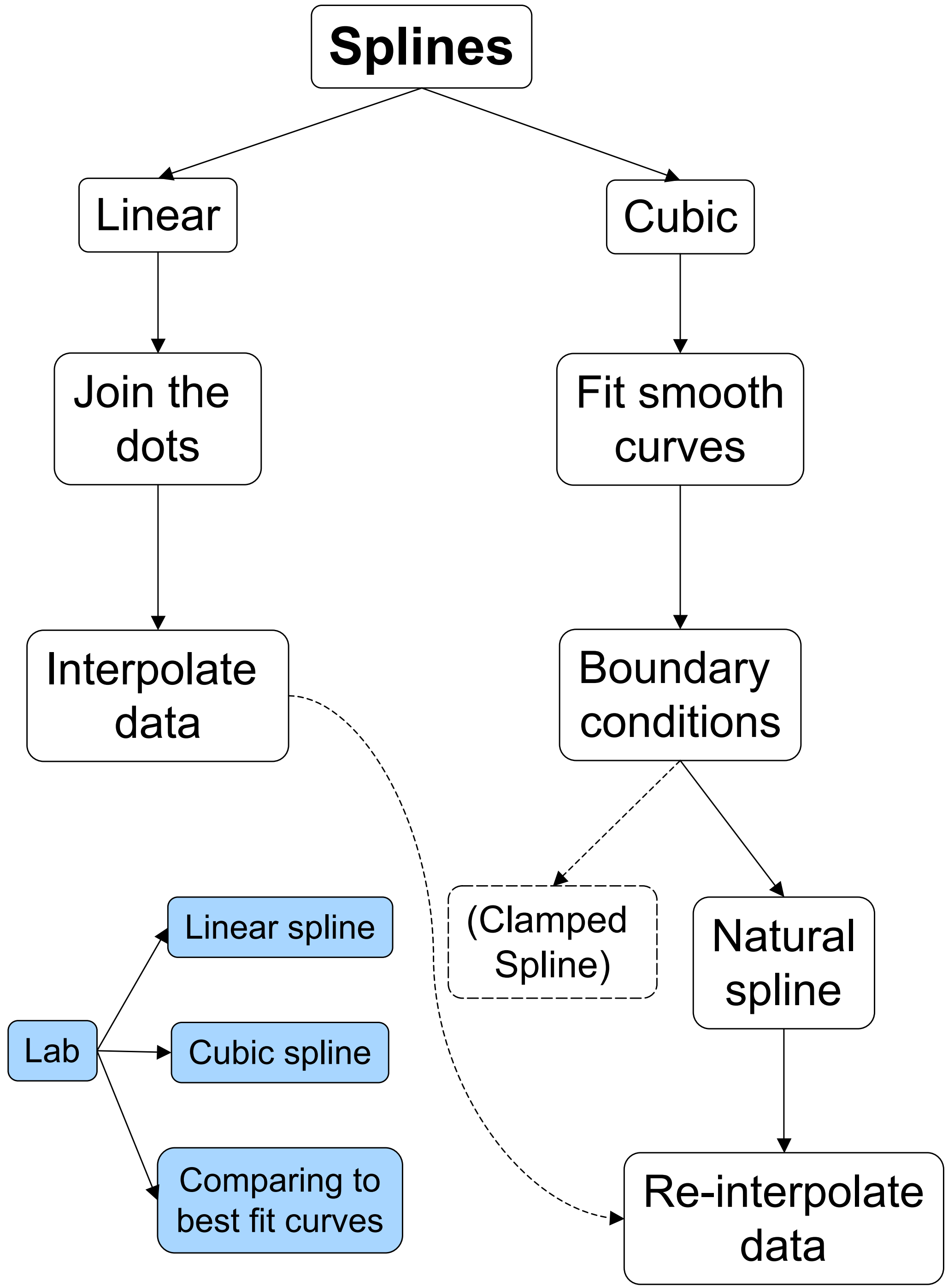
Splines

A type of model-fitting that

- Assumes the data is excellent
- Adapts the model to the data
- Is good for interpolating between data points.


Splines
Linear
Cubic
Join the dots
Fit smooth curves
Interpolate data
Boundary conditions
(Clamped Spline)
Natural spline
Re-interpolate data
Lab
Linear spline
Cubic spline
Comparing to best fit curves

An illustrative example

We use splines as an illustrative example as

- They're easy to use in Matlab
- We've used them in the previous lab
- It's important to understand the theory behind what the computer is doing.

Polynomial fitting

Polynomials are

- simple to apply
- easy to integrate and differentiate
- but very bad near the endpoints

  (oscillations and sensitivity to small changes in the data).

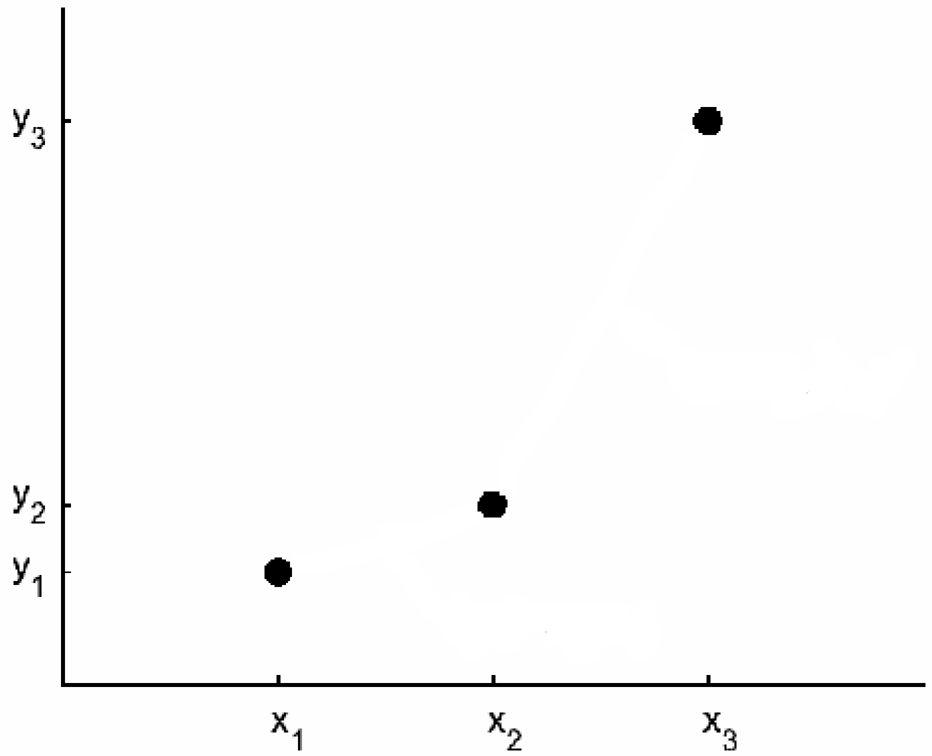
Advantages of splines

Splines

- capture the trend of the data regardless of the underlying relationship
- reduce the tendency toward oscillation
- are not sensitive to small changes in the data.

Linear splines

- Simplest method of connecting data points
- "Join-the-dots"
- Fits straight lines



This is what Matlab does when it plots.

A sudden disease increase

- Consider a disease that maintains low levels and suddenly increases sharply

x	1	2	3
y	5	8	25

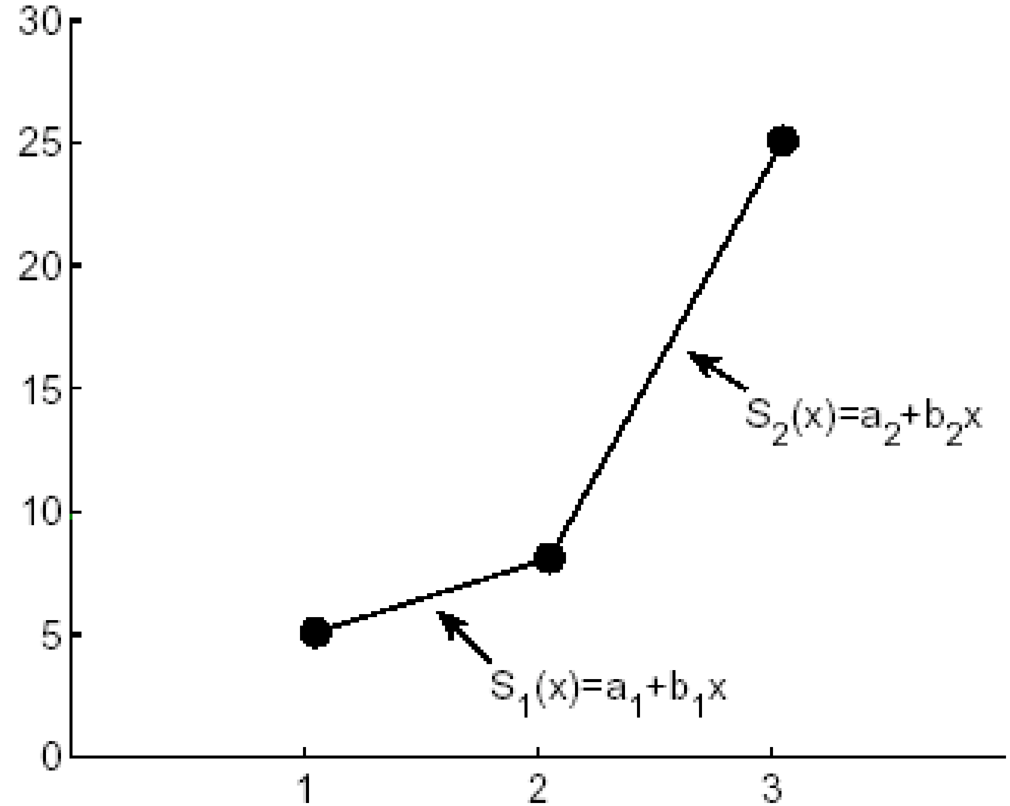
- For simplicity, we'll only fit two curves, but the method will generalise.

Two straight lines

- For $1 \leq x \leq 2$,

  $S_1(x) = a_1+b_1x$
- For $2 < x \leq 3$,

  $S_2(x) = a_2+b_2x$
- Both spline segments meet at the point (2,8).



Substituting to find $S_1(x)$

- Since $S_1(x)$ passes through (1,5) and (2,8), substitute these points in:

$$a_1+b_1(1) = 5$$

Substituting (1,5)

$$a_1+b_1(2) = 8$$

Substituting (2,8).

Remember $S_1(x)=a_1+b_1x$

Substituting to find $S_2(x)$

- Since $S_2(x)$ passes through (2,8) and (3,25), substitute these points in:

$$a_2+b_2(2) = 8$$

Substituting (2,8)

$$a_2+b_2(3) = 25$$

Substituting (3,25).

Remember $S_2(x)=a_2+b_2x$

How to solve Ac=b in Matlab

- If Ac=b and det(A)≠0, then $c=A^{-1}b$
- Matlab code:

```
A=[1 1 0 0;
    1 2 0 0;
    0 0 1 2;
    0 0 1 3];
b=[5;8;8;25];
c=A^(-1)*b
```

You can do this all on one line if you like

Remember, the semi-colon withholds the output

No period, because this is matrix multiplication

- Thus c = [2;3;-26;17].

Prediction with splines

- We've solved the linear spline:

Interval	Spline model
$1 \leq x \leq 2$	$S_1(x) = 2+3x$
$2 < x \leq 3$	$S_2(x) = -26+17x$

- Let's use it to predict. Eg

$$y(1.67) = 2+3(1.67) = 7.01$$

$$y(2.33) = -26+17(2.33) = 13.61.$$

A problem with the linear spline

The linear spline is

- easy to construct
- intuitively clear
- but isn't "smooth" (i.e. the derivatives at the midpoint are not continuous).

Cubic splines

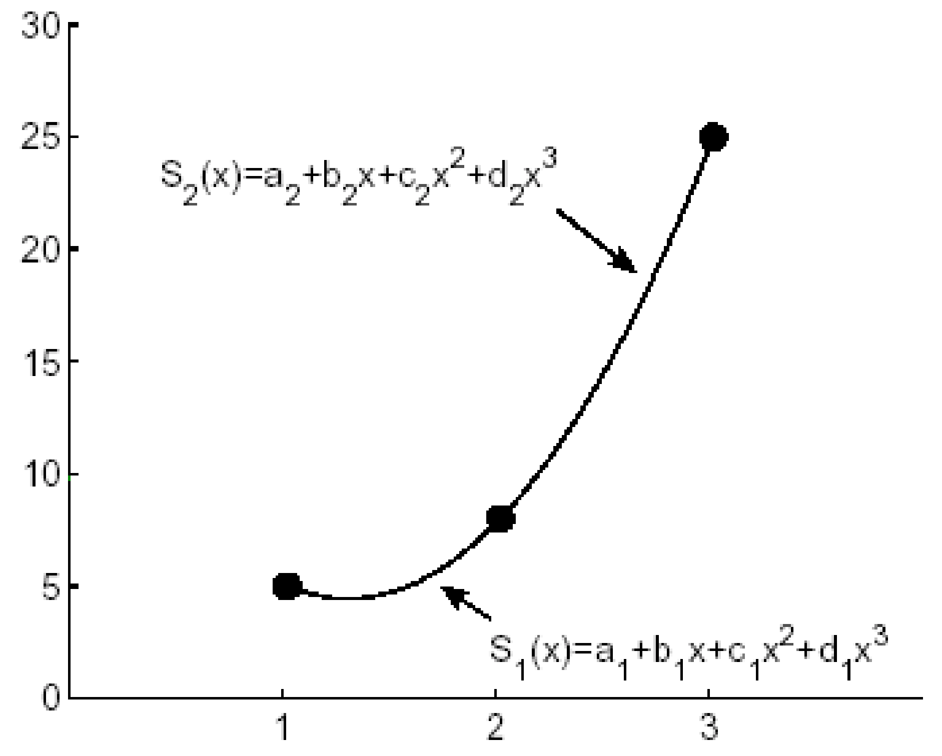
- For $1 \leq x \leq 2$,

$S_1(x)=a_1+b_1x+c_1x^2+d_1x^3$

- For $2 < x \leq 3$,

$S_2(x)=a_2+b_2x+c_2x^2+d_2x^3$

- Both spline segments meet at the point (2,8).



First and second derivatives

Since $S_1(x)=a_1+b_1x+c_1x^2+d_1x^3$,

- $S_1'(x)=b_1+2c_1x+3d_1x^2$
- $S_1''(x)=2c_1+6d_1x$

Similarly,

- $S_2'(x)=b_2+2c_2x+3d_2x^2$
- $S_2''(x)=2c_2+6d_2x$.

The derivative of x^n is nx^{n-1} and the derivative of a constant is zero

Smoothness

Cubic splines match up

- slopes (continuous first derivatives)
- curvatures (second derivatives)

We have eight unknowns (a_1,b_1,c_1,d_1, a_2,b_2,c_2,d_2), so we need eight equations.

Match points

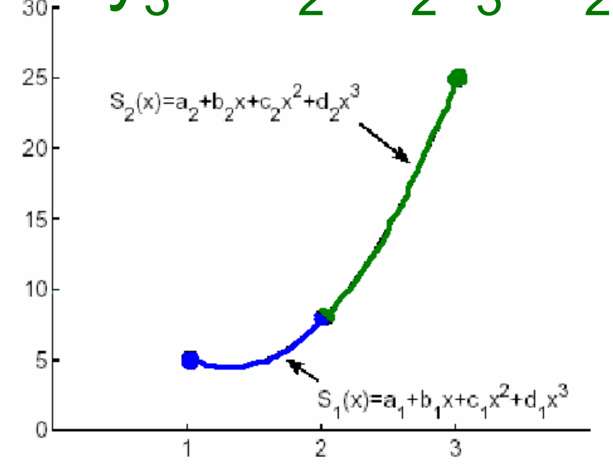
- First we "join-the-dots", ensuring our splines match the data points:

$$y_1 = a_1+b_1x_1+c_1x_1^2+d_1x_1^3 \quad \text{(first point of } S_1\text{)}$$

$$y_2 = a_1+b_1x_2+c_1x_2^2+d_1x_2^3 \quad \text{(last point of } S_1\text{)}$$

$$y_2 = a_2+b_2x_2+c_2x_2^2+d_2x_2^3 \quad \text{(first point of } S_2\text{)}$$

$$y_3 = a_2+b_2x_3+c_2x_3^2+d_2x_3^3 \quad \text{(last point of } S_2\text{)}$$



These are four of our eight equations.

Match slopes

- The slopes must match at the interior data point x_2
- i.e. $S_1'(x_2) = S_2'(x_2)$
- which means

$$b_1+2c_1x_2+3d_1x_2^2 = b_2+2c_2x_2+3d_2x_2^2$$

This our fifth out of eight equations.

Match curvatures

- The curvatures must match at the interior data point x_2
- i.e. $S_1''(x_2) = S_2''(x_2)$
- which means

$$2c_1+6d_1x_2 = 2c_2+6d_2x_2$$

This our sixth out of eight equations.

The last two equations

- We still need two equations, but we've exhausted information about the interior point
- Where can we find them?
- Answer: use the endpoints.

Natural spline

- In the absence of other information, let's make the first derivatives constant at the endpoints
- This means the second derivative must be zero
- This is called the *natural spline.*

Remember, the derivative of a constant is always zero

Derivatives at the endpoints

- Our endpoints are x_1 and x_3
- Thus

$$S_1''(x_1) = 2c_1+6d_1x_1 = 0$$

$$S_2''(x_3) = 2c_2+6d_2x_3 = 0$$

These our seventh and eighth equations.

(Clamped spline)

- If we had information about the endpoints, we could use that
- This would 'clamp' the spline to that information, hence a *clamped spline*
- We won't use this method, but…

  …it's important to realise the cubic spline method isn't unique.

An example

- Let's use the same data to construct the cubic spline through our three points.
- We'll construct by hand what the computer does.
- We don't normally do this in practice…

  …but for three data points it's just about manageable.

Substituting to find $S_1(x)$

- Since $S_1(x)$ passes through (1,5) and (2,8), substitute these points in:

$$a_1+b_1(1)+c_1(1^2)+d_1(1^3) = 5$$

Substituting (1,5)

$$a_1+b_1(2)+c_1(2^2)+d_1(2^3) = 8$$

Substituting (2,8).

$S_1(x)=a_1+b_1x+c_1x^2+d_1x^3$

Substituting to find $S_2(x)$

- Since $S_2(x)$ passes through (2,8) and (3,25), substitute these points in:

$$a_2+b_2(2)+c_2(2^2)+d_2(2^3) = 8$$

Substituting (2,8)

$$a_2+b_2(3)+c_2(3^2)+d_2(3^3) = 25$$

Substituting (3,25).

$S_2(x)=a_2+b_2x+c_2x^2+d_2x^3$

Matching derivatives at $x_2=2$

- $S_1'(2) = S_2'(2)$

  $$b_1+2c_1(2)+3d_1(2^2) = b_2+2c_2(2)+3d_2(2^2)$$

- $S_1''(2) = S_2''(2)$

  $$2c_1+6d_1(2) = 2c_2+6d_2(2).$$

$S_1'($ $S_1''(x)=2c_1+6d_1x$

$S_2'($ $S_2''(x)=2c_2+6d_2x$

Natural spline at the endpoints

- For a natural spline, $S_1''(1) = S_2''(3) = 0$

$$2c_1+6d_1(1) = 0$$

$$2c_2+6d_2(3) = 0.$$

$S_1''(x)=2c_1+6d_1x$

$S_2''(x)=2c_2+6d_2x$

Thus, our eight linear equations are

$$a_1+b_1+c_1+d_1 = 5$$

$$a_1+2b_1+4c_1+8d_1 = 8$$

$$a_2+2b_2+4c_2+8d_2 = 8$$

$$a_2+3b_2+9c_2+27d_2 = 25$$

$$b_1+4c_1+12d_1-b_2-4c_2-12d_2 = 0$$

$$2c_1+12d_1-2c_2-12d_2 = 0$$

$$2c_1+6d_1 = 0$$

$$2c_2+18d_2 = 0.$$

But this is really a matrix system

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 2 & 4 & 8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 2 & 4 & 8 \\ 0 & 0 & 0 & 0 & 1 & 3 & 9 & 27 \\ 0 & 1 & 4 & 12 & 0 & -1 & -4 & -12 \\ 0 & 0 & 2 & 12 & 0 & 0 & -2 & -12 \\ 0 & 0 & 2 & 6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 18 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \\ c_1 \\ d_1 \\ a_2 \\ b_2 \\ c_2 \\ d_2 \end{bmatrix} = \begin{bmatrix} 5 \\ 8 \\ 8 \\ 25 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

A c b

We can solve using Matlab

- Again, the solution to Ac=b is $c=A^{-1}b$.

```
A=[1 1 1 1 0 0 0 0;
1 2 4 8 0 0 0 0;
0 0 0 0  1 2 4 8;
0 0 0 0 1 3 9 27;
0 1 4 12 0 -1 -4 -12;
0 0 2 12 0 0 -2 -12;
0 0 2 6 0 0 0 0;
0 0 0 0 0 0 2 18];
b=[5;8;8;25;0;0;0;0];
c=A^(-1)*b
```

The solution

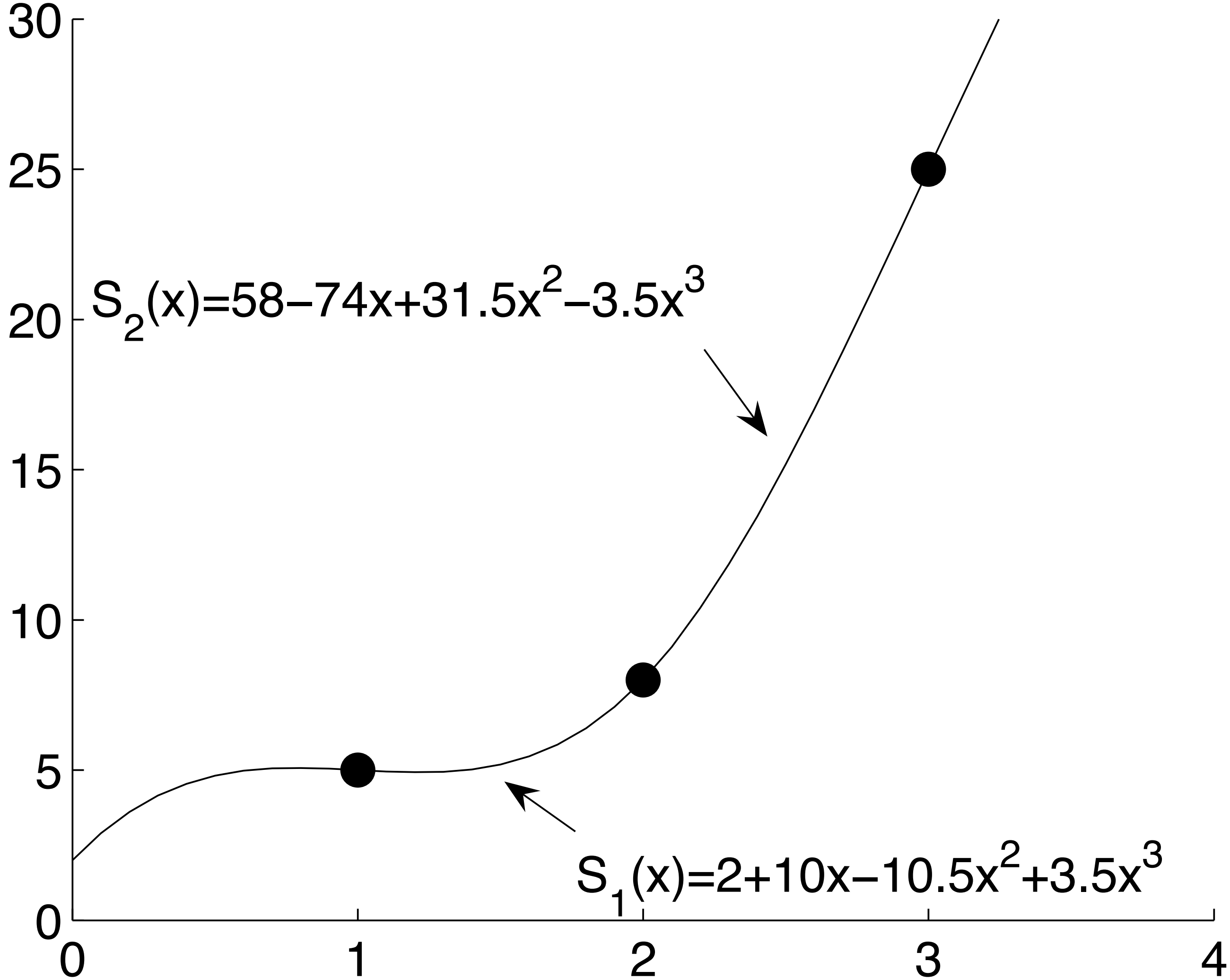
- The solution is

  c = [2; 10; -10.5; 3.5; 58; -74; 31.5; -3.5]

Interval	Spline model
$1 \leq x \leq 2$	$S_1(x) = 2 + 10x - 10.5x^2 + 3.5x^3$
$2 < x \leq 3$	$S_2(x) = 58 - 74x + 31.5x^2 - 3.5x^3$

- This solution is continuous, with continuous first and second derivatives at the interior point.

Graph of the cubic spline



Predicting data

- Let's predict our points again, using the cubic spline:

  $$y(1.67) = 2+10(1.67)-10.5(1.67)^2+3.5(1.67$$

  $$= 5.72$$

  $$y(2.33) = 58-74(2.33)+31.5(2.33)^2-3.5(2.33)^3$$

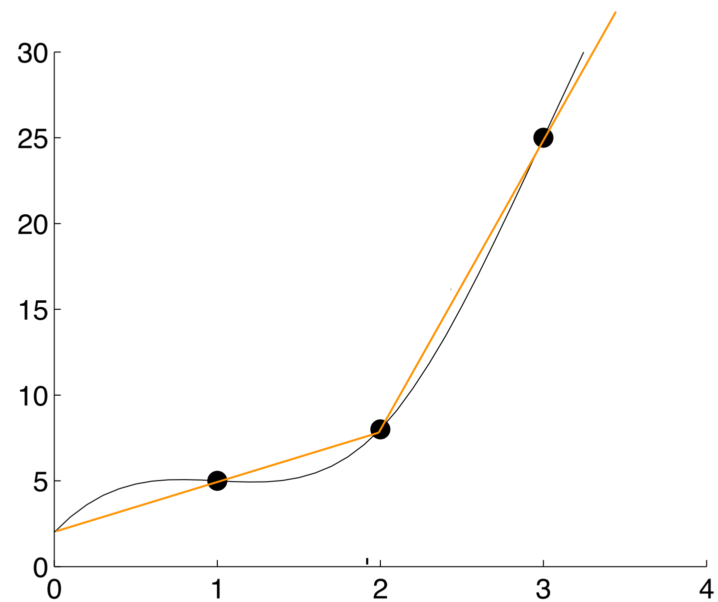
  $$= 12.32$$

- From the linear spline

  $$y(1.67) = 7.01$$

  $$y(2.33) = 13.61$$

- Which do you have more faith in?



Extending to more data points

For more data points

- Each spline is forced to pass through the endpoints of the interval over which it is defined
- We match the first and second derivatives of adjacent points
- The natural (or clamped) condition is applied at the two exterior data points.

Generalising

- Of course, we don't do that by hand
- Matlab does it quite nicely, as we saw in the previous lab
- But this is a taste of the theory behind a simple computer algorithm for fitting one type of curve to data.