Cubic Splines

Given the following list of points:

$$\begin{array}{llllllllll} x: & a & = & x_0 & < & x_1 & < & \cdots & < & x_n = b \\ y: & & & y_0 & & y_1 & & \cdots & & y_n \end{array}$$

➤ **<u>Basic idea:</u>** ***"piecewise polynomial interpolation"***

- **Use lower order polynomials that interpolate on each subinterval $[x_i, x_{i+1}]$.**
- **Force the polynomials to join up as smoothly as possible.**

➤ **Simplest example: a linear spline just "connects the dots"**

<u>Definition:</u> A cubic spline $S(x)$ is a piecewise-defined function that satisfies the following conditions:

1. **$S(x) = S_i(x)$ is a cubic polynomial on each subinterval $[x_i, x_{i+1}]$ for $i = 0, 1, \ldots, n-1$.**
2. **$S(x_i) = y_i$ for $i = 0, 1, \ldots, n$ (S interpolates all the points).**
3. **$S(x)$, $S'(x)$, and $S''(x)$ are continuous on $[a, b]$ (S is smooth).**

So we write the n cubic polynomial pieces as

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3,$$

for $i = 0, 1, \ldots, n-1$, where a_i, b_i, c_i, and d_i represent $4n$ unknown coefficients.

Derivation

Using the definition, let's determine equations relating the coefficients and keep a count of the number of equations:

		# eqns.
Interpolation and continuity:		
$S_i(x_i) = y_i$	for $i = 0, 1, \ldots, n-1$	n
$S_i(x_{i+1}) = y_{i+1}$	for $i = 0, 1, \ldots, n-1$	n
(based on both of these, $S(x)$ is continuous)		
Derivative continuity:		
$S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$	for $i = 0, 1, \ldots, n-2$	$n-1$
$S_i''(x_{i+1}) = S_{i+1}''(x_{i+1})$	for $i = 0, 1, \ldots, n-2$	$n-1$
	Total # of equations:	$4n-2$

There are still 2 equations missing! (later)

Derivation (cont'd)

We need expressions for the derivatives of S_i:

$$\begin{aligned}
S_i(x) &= a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \\
S_i'(x) &= b_i + 2c_i(x - x_i) + 3d_i(x - x_i)^2 \\
S_i''(x) &= 2c_i + 6d_i(x - x_i)
\end{aligned}$$

It is very helpful to introduce the $h_i = x_{i+1} - x_i$. Then the spline conditions can be written as follows:

- $S_i(x_i) = y_i$ **for** $i = 0, 1, \ldots, n-1$**:**

  $$\boxed{a_i = y_i}$$

- $S_i(x_{i+1}) = y_{i+1}$ **for** $i = 0, 1, \ldots, n-1$**:**

  $$\boxed{a_i + h_i b_i + h_i^2 c_i + h_i^3 d_i = y_{i+1}}$$

- $S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$ **for** $i = 0, 1, \ldots, n-2$**:**

  $$\boxed{b_i + 2h_i c_i + 3h_i^2 d_i - b_{i+1} = 0}$$

- $S_i''(x_{i+1}) = S_{i+1}''(x_{i+1})$ **for** $i = 0, 1, \ldots, n-2$**:**

  $$\boxed{2c_i + 6h_i d_i - 2c_{i+1} = 0}$$

The boxed equations above can be written as a large linear system for the $4n$ unknowns

$$[a_0, b_0, c_0, d_0, a_1, b_1, c_1, d_1, \ldots, a_{n-1}, b_{n-1}, c_{n-1}, d_{n-1}]^T$$

Alternate formulation

This linear system can be simplified considerably by defining

$$m_i = S_i''(x_i) = 2c_i \qquad \text{or} \qquad \boxed{c_i = \frac{m_i}{2}}$$

and thinking of the m_i as unknowns instead. Then:

- $S_i''(x_{i+1}) = S_{i+1}''(x_{i+1})$ **for** $i = 0, 1, \ldots, n-2$:

$$\Longrightarrow \quad 2c_i + 6h_i d_i - 2c_{i+1} = 0$$

$$\Longrightarrow \quad m_i + 6h_i d_i - m_{i+1} = 0$$

$$\Longrightarrow \quad \boxed{d_i = \frac{m_{i+1} - m_i}{6h_i}}$$

- $S_i(x_i) = y_i$ **and** $S_i(x_{i+1}) = y_{i+1}$ **for** $i = 0, 1, \ldots, n-1$:

$$\Longrightarrow \quad y_i + h_i b_i + h_i^2 c_i + h_i^3 d_i = y_{i+1}$$

Substitute c_i and d_i from above:

$$\Longrightarrow \quad \boxed{b_i = \frac{y_{i+1} - y_i}{h_i} - \frac{h_i}{2} m_i - \frac{h_i}{6}(m_{i+1} - m_i)}$$

- $S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$ **for** $i = 0, 1, \ldots, n-2$:

$$\Longrightarrow \quad b_i + 2h_i c_i + 3h_i^2 d_i = b_{i+1}$$

Substitute b_i, c_i and d_i from above and simplify:

$$\boxed{h_i m_i + 2(h_i + h_{i+1}) m_{i+1} + h_{i+1} m_{i+2} = 6\left[\frac{y_{i+2} - y_{i+1}}{h_{i+1}} - \frac{y_{i+1} - y_i}{h_i}\right]}$$

<u>Notice:</u> These are $n-1$ linear equations for $n+1$ unknowns

$$[m_0, m_1, m_2, \ldots, m_n]^T \qquad \text{where} \qquad m_i = g_i''(x_i).$$

Endpoint Conditions: Natural Spline

- **Now, we'll deal with the two missing equations ...**
- **Note: Derivative matching conditions are applied *only* at interior points, which suggests applying some constraints on the derivative(s) at x_0 and x_n.**
- **There is no unique choice, but several common ones are**

   1. *Natural Spline* (zero curvature):

$$m_0 = 0 \qquad \text{and} \qquad m_n = 0$$

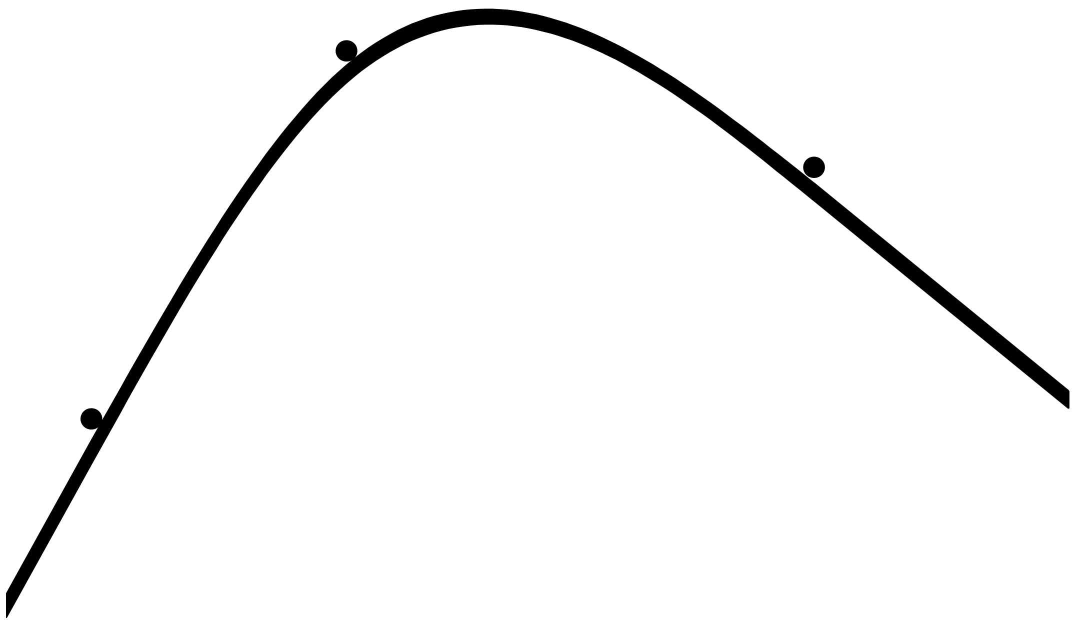
for taken together gives an $(n+1) \times (n+1)$ system:

$$\begin{bmatrix} 1 & 0 & 0 & & \cdots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & 0 & & \\ 0 & h_1 & 2(h_1+h_2) & h_2 & 0 & \\ 0 & 0 & h_2 & 2(h_2+h_3) & h_3 & \vdots \\ \vdots & & & \ddots & \ddots & \ddots \\ & & 0 & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & \cdots & & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \\ \vdots \\ m_n \end{bmatrix}$$

$$= 6 \begin{bmatrix} 0 \\ \frac{y_2-y_1}{h_1} - \frac{y_1-y_0}{h_0} \\ \frac{y_3-y_2}{h_2} - \frac{y_2-y_1}{h_1} \\ \frac{y_4-y_3}{h_3} - \frac{y_3-y_2}{h_2} \\ \vdots \\ \frac{y_n-y_{n-1}}{h_{n-1}} - \frac{y_{n-1}-y_{n-2}}{h_{n-2}} \\ 0 \end{bmatrix}$$

... or you could just drop the equations for m_0 and m_n and write it as an $(n-1) \times (n-1)$ system.

Interpretation of a Natural Spline



- The word “spline” comes from the draftsman’s spline, which is shown here constrained by 3 pegs.
- There is no force on either end to bend the spline beyond the last pegs, which results in a flat shape (zero curvature).
- This is why the $S'' = 0$ end conditions are called *natural*.

Endpoint Conditions: Clamped Spline

2. ***Clamped Spline*****, sometimes called a "complete spline" (derivative is specified):**

$$S_0'(x_0) = A \quad \Longrightarrow \quad b_0 = A$$

$$\Longrightarrow \quad A = \frac{y_1 - y_0}{h_0} - \frac{h_0}{2} m_0 - \frac{h_0}{6}(m_1 - m_0)$$

$$\Longrightarrow \quad 2h_0 m_0 + h_0 m_1 = 6\left[\frac{y_1 - y_0}{h_0} - A\right]$$

and

$$S_{n-1}'(x_n) = B \quad \Longrightarrow \quad b_{n-1} = B$$

$$\Longrightarrow \quad h_{n-1} m_{n-1} + 2h_{n-1} m_n = 6\left[B - \frac{y_n - y_{n-1}}{h_{n-1}}\right]$$

These two equations are placed in the first/last rows

$$\begin{bmatrix} 2h_0 & h_0 & 0 & \cdots & \cdots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & 0 & & \vdots \\ 0 & h_1 & 2(h_1+h_2) & h_2 & 0 & \vdots \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & \cdots & \cdots & 0 & h_{n-1} & 2h_{n-1} \end{bmatrix}$$

and the first/last RHS entries are modified accordingly.

Endpoint Conditions: Not-A-Knot

3. *Not-A-Knot Spline* (third derivative matching):

$$S_0'''(x_1) = S_1'''(x_1) \qquad \textbf{and} \qquad S_{n-2}'''(x_{n-1}) = S_{n-1}'''(x_{n-1})$$

Using $S_i'''(x) = 6d_i$ and $d_i = \frac{m_{i+1}-m_i}{6}$, these conditions become

$$m_1 - m_0 = m_2 - m_1$$

and

$$m_{n-1} - m_{n-2} = m_n - m_{n-1}$$

The matrix in this case is

$$\begin{bmatrix} -1 & 2 & -1 & \cdots & \cdots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & 0 & & \vdots \\ 0 & h_1 & 2(h_1+h_2) & h_2 & 0 & \vdots \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & \cdots & \cdots & -1 & 2 & -1 \end{bmatrix}$$

and the first/last RHS entries are zero.

<u>Note:</u> Matlab's `spline` function implements the not-a-knot condition (by default) as well as the clamped spline, but not the natural spline. Why not? (see Homework #4).

Summary of the Algorithm

Starting with a set of $n+1$ data points

$$(x_0, y_0), \quad (x_1, y_1), \quad (x_2, y_2), \ldots, (x_n, y_n)$$

1. **Calculate the values $h_i = x_{i+1} - x_i$ for $i = 0, 1, 2, \ldots, n-1$.**
2. **Set up the matrix A and right hand side vector r as on page 6 for the natural spline. If clamped or not-a-knot conditions are used, then modify A and r appropriately.**
3. **Solve the $(n+1) \times (n+1)$ linear system $Am = r$ for the second derivative values m_i.**
4. **Calculate the spline coefficients for $i = 0, 1, 2, \ldots, n-1$ using:**

$$a_i = y_i$$

$$b_i = \frac{y_{i+1} - y_i}{h_i} - \frac{h_i}{2} m_i - \frac{h_i}{6}(m_{i+1} - m_i)$$

$$c_i = \frac{m_i}{2}$$

$$d_i = \frac{m_{i+1} - m_i}{6h_i}$$

5. **On each subinterval $x_i \leq x \leq x_{i+1}$, construct the function**

$$g_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$$

Natural Spline Example

x	4.00	4.35	4.57	4.76	5.26	5.88
y	4.19	5.77	6.57	6.23	4.90	4.77
h	0.35	0.22	0.19	0.50	0.62	

Solve $Am = r$ when

$$A = \begin{bmatrix} 1 & 0 & & 0 & 0 & 0 \\ 0.35 & 1.14 & 0.22 & 0 & 0 & 0 \\ 0 & 0.22 & 0.82 & 0.19 & 0 & 0 \\ 0 & 0 & 0.19 & 1.38 & 0.50 & 0 \\ 0 & 0 & 0 & 0.50 & 2.24 & 0.62 \\ 0 & 0 & 0 & & 0 & 1 \end{bmatrix} \quad \text{and} \quad r = \begin{bmatrix} 0 \\ -5.2674 \\ -32.5548 \\ -5.2230 \\ 14.70180 \end{bmatrix}$$

$$\Longrightarrow \quad m = [0,\ 3.1762,\ -40.4021,\ -0.6531,\ 6.7092,\ 0]^T$$

Calculate the spline coefficients:

	x_i	$y_i = a_i$	b_i	c_i	d_i	Interval
$S_0(x)$	4.00	4.19	4.3290	0	1.5125	$[4.00,\ 4.35]$
$S_1(x)$	4.35	5.77	4.8848	1.5881	−33.0139	$[4.35,\ 4.57]$
$S_2(x)$	4.57	6.57	0.7900	−20.2010	34.8675	$[4.57,\ 4.76]$
$S_3(x)$	4.76	6.23	−3.1102	−0.3266	2.4541	$[4.76,\ 5.26]$
$S_4(x)$	5.26	4.90	−1.5962	3.3546	−1.8035	$[5.26,\ 5.88]$

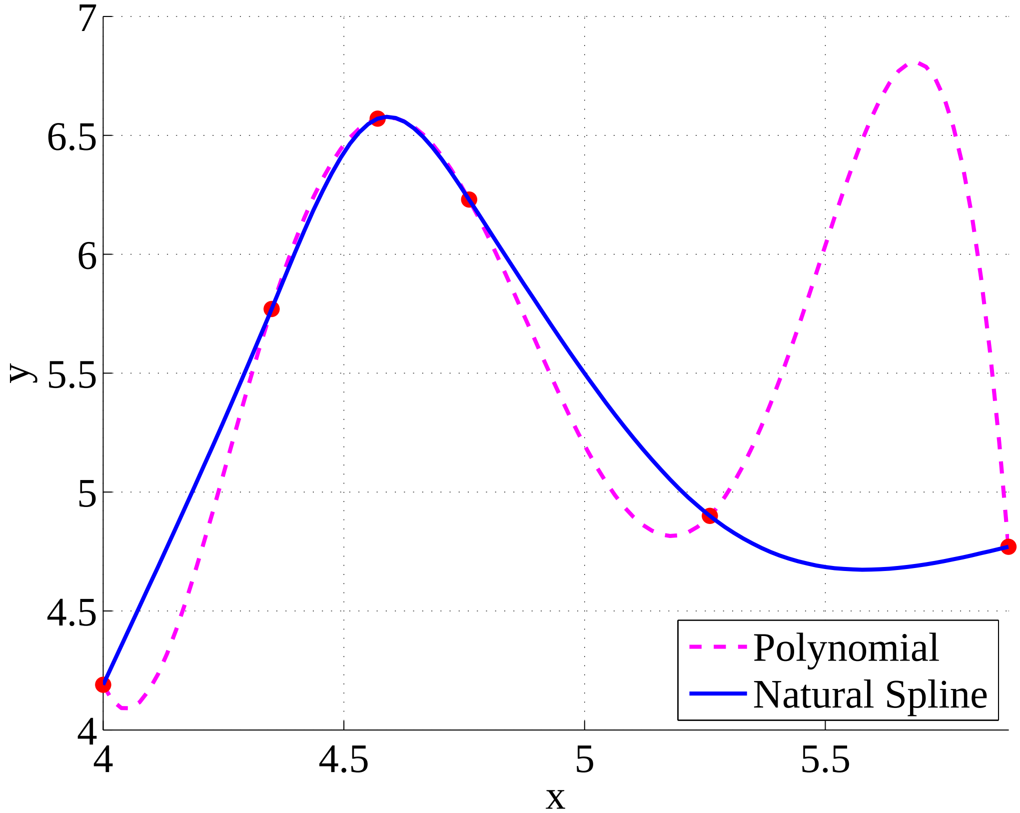
Then the spline functions are easy to read off, for example:

$$S_2(x) = 6.57 + 0.7900(x - 4.57) - 20.2010(x - 4.57)^2 + 34.8675(x - 4.57)^3$$

$$S_3(x) = 6.23 - 3.1102(x - 4.76) - 0.3266(x - 4.76)^2 + 2.4541(x - 4.76)^3$$

Exercise: **Verify that S_2 and S_3 satisfy the conditions defining a cubic spline.**

Splines vs. Interpolation



The Newton divided difference table for the data points is

x	y	a_1	a_2	a_3	a_4	a_5
4.00	4.19					
		4.5143				
4.35	5.77		-1.5402			
		3.6364		-15.3862		
4.57	6.57		-13.2337		22.6527	
		-1.7895		13.1562		-15.7077
4.76	6.23		-1.2616		-6.8778	
		-2.6600		2.6331		
5.26	4.90		2.1878			
		-0.2097				
5.88	4.77					

$$\Longrightarrow \quad P_5(x) = 4.19 + 4.5143(x-4) - 1.5402(x-4)(x-4.35) - \dots$$

Clamped Spline Example

x	4.00	4.35	4.57	4.76	5.26	5.88
y	4.19	5.77	6.57	6.23	4.90	4.77
h	0.35	0.22	0.19	0.50	0.62	

(same data)

Assume $S'(4.00) = -1.0$ and $S'(5.88) = -2.0$ (clamped).

Solve $Am = r$ where

$$A = \begin{bmatrix} 0.70 & 0.35 & 0 & 0 & 0 & 0 \\ 0.35 & 1.14 & 0.22 & 0 & 0 & 0 \\ 0 & 0.22 & 0.82 & 0.19 & 0 & 0 \\ 0 & 0 & 0.19 & 1.38 & 0.50 & 0 \\ 0 & 0 & 0 & 0.50 & 2.24 & 0.62 \\ 0 & 0 & 0 & 0 & 0.62 & 1.24 \end{bmatrix} \quad \text{and} \quad r = \begin{bmatrix} 33.0858 \\ -5.2674 \\ -32.5548 \\ -5.2230 \\ 14.7018 \\ -10.7418 \end{bmatrix}$$

$$\Longrightarrow \quad m = [54.5664,\ -14.6022,\ -35.0875,\ -3.0043,\ 11.1788,\ -14.2522]^T$$

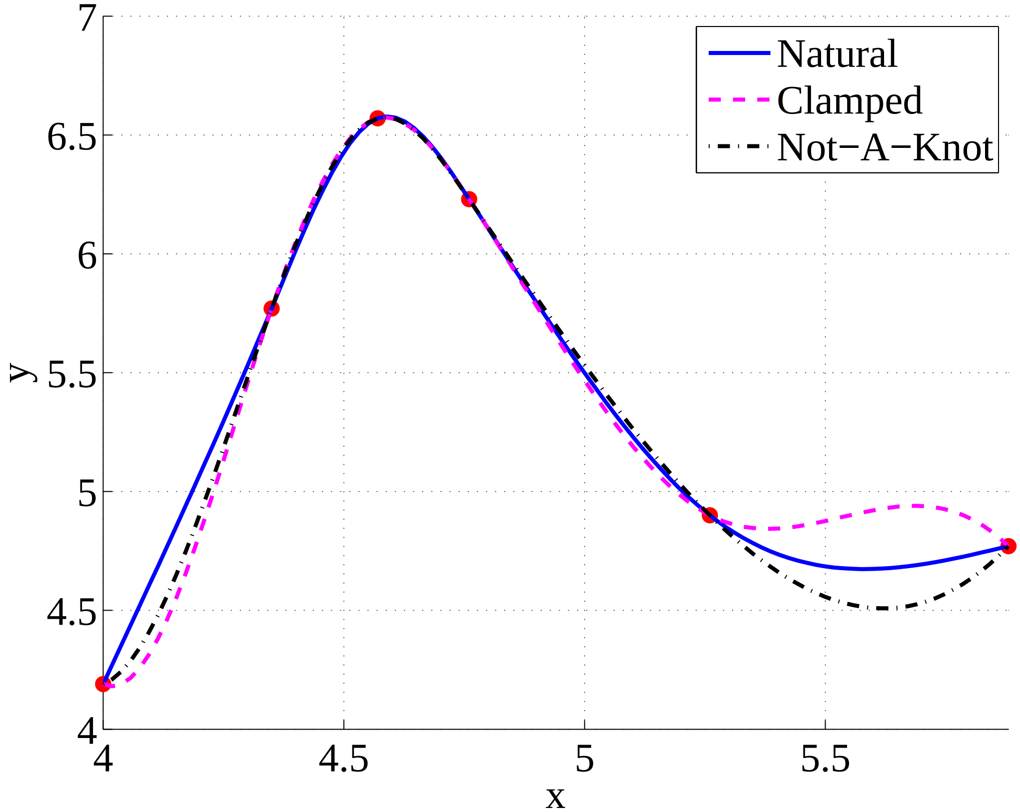
	x_i	$y_i = a_i$	b_i	c_i	d_i	Interval
$S_0(x)$	4.00	4.19	-1.0000	27.2832	-32.9375	$[4.00,\ 4.35]$
$S_1(x)$	4.35	5.77	5.9937	-7.3011	-15.5191	$[4.35,\ 4.57]$
$S_2(x)$	4.57	6.57	0.5279	-17.5437	28.1431	$[4.57,\ 4.76]$
$S_3(x)$	4.76	6.23	-3.0908	-1.5021	4.7277	$[4.76,\ 5.26]$
$S_4(x)$	5.26	4.90	-1.0472	5.5894	-6.8363	$[5.26,\ 5.88]$

Exercise: Verify that the spline satisfies the clamped end conditions:

$$S_0'(4.00) = -1.0 \quad \text{and} \quad S_4'(5.88) = -2.0.$$

Comparison

Below is a comparison of the splines with various endpoint conditions:



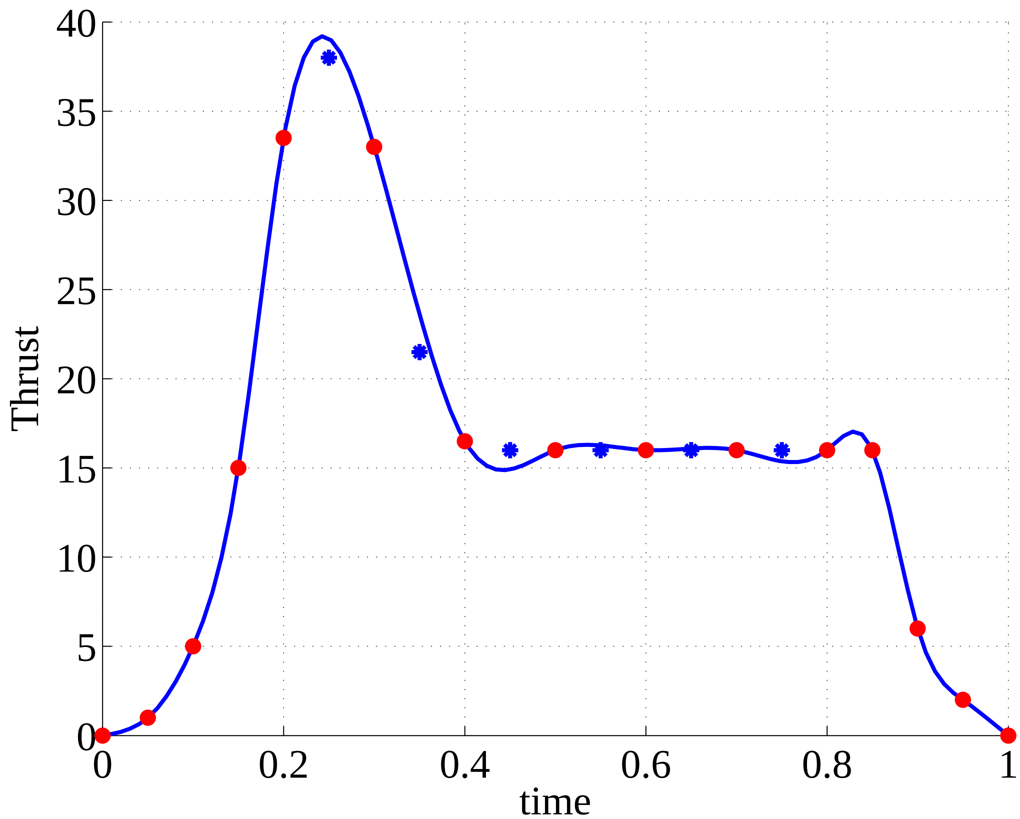
Second Example

Consider the following data representing the thrust of a model rocket versus time.

t	T
0.00	0.00
0.05	1.00
0.10	5.00
0.15	15.00
0.20	33.50
0.30	33.00
0.40	16.50
0.50	16.00

t	T
0.60	16.00
0.70	16.00
0.80	16.00
0.85	16.00
0.90	6.00
0.95	2.00
1.00	0.00

Below is a plot of the natural spline:



Notice how wiggles are introduced near the ends of the flat region – non-smooth data cause some problems for cubic splines.