Natural Splines

0. Motivation

One very common problem when having to manipulate a set of data (presumably collected from an experiment) is to approximate such data in a way that allows us to do so easily and reliably. One common requirement is that we find a function that synthesizes this data in order for us to work with the representative function rather than the data itself.

Often the data is presented in a table such as the one following.

X	Y
1.0000000000	2.0000000000
3.0000000000	5.0000000000
6.0000000000	3.0000000000
8.0000000000	4.5000000000

Figure 1. A set of data points.

A common requirement is that the synthesizing function collocates the data, i.e. we would like to find an expression $y = f(x)$ where $y_i = f(x_i) \quad \forall i$. A simple way to achieve this is to collocate the data with a polynomial. In this simple case, the polynomial would be of the form $y = a + bx + cx^2 + dx^3$. The coefficients *a, b, c, d* may be easily found from the linear system of equations:

$$\begin{aligned} y_0 &= a + bx_0 + cx_0^{\ 2} + dx_0^{\ 3} \\ y_1 &= a + bx_1 + cx_1^{\ 2} + dx_1^{\ 3} \\ y_2 &= a + bx_2 + cx_2^{\ 2} + dx_2^{\ 3} \\ y_3 &= a + bx_3 + cx_3^{\ 2} + dx_3^{\ 3} \end{aligned} \qquad (0)$$

In general, the approximation polynomial is of the form $y = \sum_{i=0}^{n} c_i x^i$ where, clearly, we require *n+1* data points and the highest degree is *n*. However, as *n* grows so does the number of equations. More importantly, the behavior of the approximant may become unstable, in the sense that interpolated values may lie far from the observed data. This is due to the fact that the high degree inherent in the approximant induces important oscillations between the observed data points because the polynomial, when collocating such points "has" to adjust itself in the realm of the unknown intervals.

To avoid this unsatisfactory state of affairs, we may abandon the idea of collocating the data with a single polynomial and try, instead, to collocate these data with several well-behaved ones. "Well-behaved", in this sense, has to be defined. We do this by requiring:

a) There are *n* elements (for *n+1* data points) in the polynomial set [which we denote by *S(x)*]. The *i-th* element of the set is denoted by $S_i(x)$.
b) The polynomial set is required to comply with $S(x_i) = y_i \quad \forall i$, i.e. it must collocate the observations.
c) We also require that the set be "smooth" between neighbouring elements. We guarantee this if $S_i'(x_i) = S_{i+1}'(x_i)$ for *i = 1, ..., n-2*.

d) We, furthermore, require that the concavity of *S(x)* does not change abruptly, i.e. $S_i''(x_i) = S_{i+1}''(x_i)$ for *i = 1, ..., n-2*.
e) Finally, we would like the curvature of *S(x)* be such that the oscillations between observed data be kept as small as possible (given that condition (a) is fulfilled).

1. Definition of a Natural Spline

We now define a set of cubic polynomials *S(x)* which we call a "spline" as

$$S(x) = \sum_{i=0}^{n-1} (a_{i,i+1} + b_{i,i+1}x + c_{i,i+1}x^2 + d_{i,i+1}x^3)\delta(x)$$

$$\delta(x) = \begin{cases} 1 & x_i \le x \le x_{i+1} \\ 0 & otherwise \end{cases}$$

One such polynomial is illustrated in figure 2.

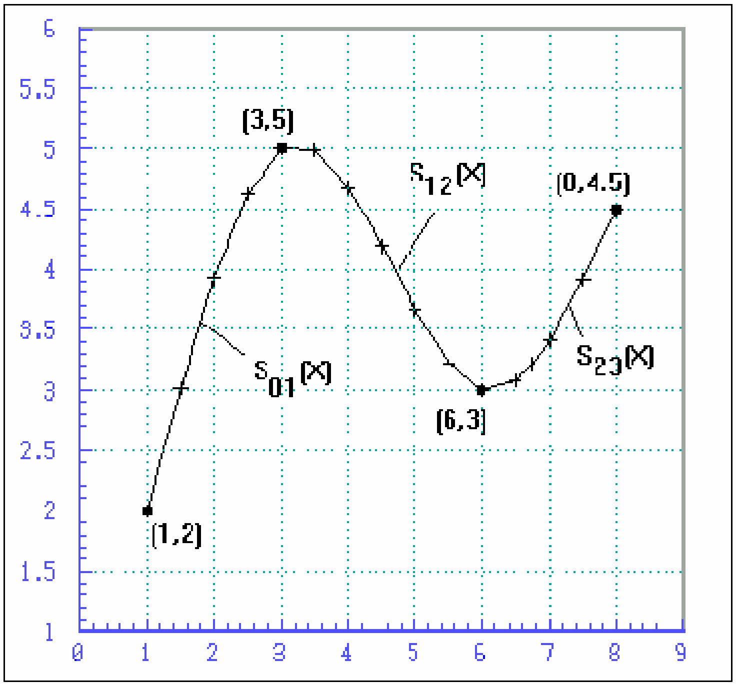


Figure 2. An Example of a Spline

The elements of the spline are of degree 3 because this is the smallest degree that allows us to comply with all of the conditions above, as will become clear from what follows. In the figure, the polynomials are given by

$$S_{01}(x) = a_{01} + b_{01}x + c_{01}x^2 + d_{01}x^3$$
$$S_{12}(x) = a_{12} + b_{12}x + c_{12}x^2 + d_{12}x^3$$
$$S_{23}(x) = a_{23} + b_{23}x + c_{23}x^2 + d_{23}x^3$$

Finding the spline means “to determine the set of coefficients” above.

From the collocating condition we may write:

$$a_{01} + b_{01}x_0 + c_{01}x_0^{\ 2} + d_{01}x_0^{\ 3} = y_0$$
$$a_{01} + b_{01}x_1 + c_{01}x_1^{\ 2} + d_{01}x_1^{\ 3} = y_1$$

which is a system with 4 unknowns and only two conditions. We may establish two further conditions from the fact that the spline is required to be continuous for its first and second derivatives. The first and second derivatives are given by:

$$b_{i,i+1} + 2c_{i,i+1}x + 3d_{i,i+1}x^2 = y'(x)$$
$$2c_{i,i+1} + 6d_{i,i+1}x = y''(x)$$

Then

$$b_{01} + 2c_{01}x_1 + 3d_{01}x_1^{\ 2} = b_{12} + 2c_{12}x_1 + 3d_{12}x_1^{\ 2}$$

and

$$2c_{01} + 6d_{01}x_1 = 2c_{12} + 6d_{12}x_1$$

For the second cubic we also know that

$$a_{12} + b_{12}x_1 + c_{12}x_1^{\ 2} + d_{12}x_1^{\ 3} = y_1$$
$$a_{12} + b_{12}x_2 + c_{12}x_2^{\ 2} + d_{12}x_2^{\ 3} = y_2$$

Now we have 6 conditions but 8 unknows. It is easy to see that, as we continue in this fashion, there will always be *n* conditions and *n+2* unknows. Therefore, to be able to completely define the spline, we must stipulate two more conditions. These may be defined in several ways. Here we only consider (for reasons that will be explained in what follows) that

$$S''(x_0) = 0 \text{ and } S''(x_n) = 0$$

and we may solve the resulting system of linear equations by any of the known methods. This system consists of *4(n+1)* linear equations whose solutions yields *4(n+1)* defining coefficients.

It would seem that the spline is an uneconomical way to handle a collocation problem, since a single polynomial requires only *n+1* equations and the corresponding *n+1* coefficients. In what follows we develop a method which deems it unnecessary to solve the set of simultaneous linear equations and, furthermore, allows us to find the interpolated points without the need for an explicit calculation of the coefficients of the equations. Rather, we find the values of the second derivatives at the data points. Although not needed, should we decide to do so, we may find the *4(n+1)* coefficients from the *n+1* second derivatives.

We denote the i-th cubic polynomial with $S_i(x)$.

These polynomials join the "knots" of the spline in such a way that:

a) The spline collocates the original data. Hence,

$$S(x_i) = f(x_i), \quad i = 0,1,...,n$$

b) The spline is continuous on its first derivative. That is,

$$S_i'(x_i) = S_{i+1}'(x_i); \quad i = 1,2,...,n-2$$

c) The spline is also continuous on its second derivative:

$$S_i''(x_i) = S_{i+1}''(x_i); \quad i = 1,2,...,n-2$$

d) The second derivative of the spline at its end points is zero:

$$S''(x_0) = S''(x_n) = 0$$

Condition (d) gives rise to the so-called "natural" spline.

For convenience, let

$$h_i = x_{i+1} - x_i$$
$$s_i = S''(x_i)$$

From the stipulation that *S(x)* be twice continuously differentiable we may write:

$$S_i''(x) = s_i \frac{x_{i+1} - x}{h_i} + s_{i+1} \frac{x - x_i}{h_i} \qquad (1)$$

To see why this is so, we note that

$$\begin{aligned} S_i''(x) &= \frac{s_i x_{i+1} - s_i x + s_{i+1} x - s_{i+1} x_i}{x_{i+1} - x_i} \\ &= \frac{(s_{i+1} - s_i)x + (s_i x_{i+1} - s_{i+1} x_i)}{x_{i+1} - x_i} \end{aligned}$$

$$\boxed{S_i''(x) = \frac{s_{i+1} - s_i}{x_{i+1} - x_i} x + \frac{s_i x_{i+1} - s_{i+1} x_i}{x_{i+1} - x_i}}$$

We also note that, from the equation of a straight line joining points (x_1, y_1) and (x_2, y_2) that

$$
\begin{aligned}
&y = mx + b \\
&y_1 = mx_1 + b \\
&y_2 = mx_2 + b \\
&b = y_1 - mx_1 \\
&y_2 = mx_2 + y_1 - mx_1 \\
&y_2 = m(x_2 - x_1) + y_1
\end{aligned}
$$

hence

$$\boxed{m = \frac{y_2 - y_1}{x_2 - x_1}}$$

and also

$$
\begin{aligned}
&y_1 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)x_1 + b \\
&b = y_1 - \left(\frac{y_2 - y_1}{x_2 - x_1}\right)x_1 \\
&b = \frac{y_1x_2 - y_1x_1 - y_2x_1 + y_1x_1}{x_2 - x_1}
\end{aligned}
$$

from which we see that

$$\boxed{b = \frac{y_1x_2 - y_2x_1}{x_2 - x_1}}$$

If we make

$$y_1 = s_i \; ; \; y_2 = s_{i+1} \; ; \; x_1 = x_i \; and \;\; x_2 = x_{i+1}$$

we have analogous forms for the line joining the second derivatives in consecutive knots of the spline.

Integrating (1) twice, we get

$$
\begin{aligned}
S_i(x) = \frac{s_i}{6h_i}(x_{i+1} - x)^3 + \frac{s_{i+1}}{6h_i}(x - x_i)^3 + \\
c_1(x - x_i) + c_2(x_{i+1} - x) \qquad\qquad (2)
\end{aligned}
$$

From the collocation condition we have

$$
\begin{aligned}
&S_i(x_i) = f_i \\
&S_i(x_{i+1}) = f_{i+1}
\end{aligned}
$$

from which we find c_1 and c_2 and write:

$$S_i(x) = \frac{s_i}{6h_i}(x_{i+1} - x)^3 + \frac{s_{i+1}}{6h_i}(x - x_i)^3 +$$

$$\left(\frac{f_{i+1}}{h_i} - \frac{s_{i+1}h_i}{6}\right)(x - x_i) + \left(\frac{f_i}{h_i} - \frac{s_i h_i}{6}\right)(x_{i+1} - x) \quad (3)$$

Differentiating once with respect to x:

$$S_i'(x) = -\frac{s_i}{2h_i}(x_{i+1} - x)^2 + \frac{s_{i+1}}{2h_i}(x - x_i)^2 +$$

$$\frac{f_{i+1} - f_i}{h_i} - \frac{h_i}{6}(s_{i+1} - s_i) \qquad (4)$$

Since we require that the spline be continuous for its first derivative, we know that:

$$S_{i-1}'(x) = S_i'(x)$$

Therefore,

$$s_{i+1} + 2s_i\left(\frac{h_i + h_{i-1}}{h_i}\right) + \frac{h_{i-1}}{h_i}s_{i-1} = \frac{6}{h_i}\left(\frac{f_{i+1} - f_i}{h_i} - \frac{f_i - f_{i-1}}{h_{i-1}}\right)$$

$$i = 1,2,\ldots,n-1 \qquad (5)$$

Expression (5) defines a system of n+1 unknowns ($s_0, s_1, \ldots, s_n$) with n-1 conditions. We need two additional conditions which we chose (for reasons that will be discussed later) to be:

$$S''(x_0) = S''(x_n) = 0 \quad \longrightarrow \quad s_0 = s_n = 0 \qquad (6)$$

2. Calculation of the Natural Spline

We re-write expression (5) as:

$$\frac{h_{i-1}}{h_i}s_{i-1} + 2\left(1 + \frac{h_{i-1}}{h_i}\right)s_i + s_{i+1} = d_i$$

$$i = 1,2,\ldots,n-1 \qquad (7)$$

where

$$d_i = \frac{6}{h_i}\left(\frac{f_{i+1} - f_i}{h_i} - \frac{f_i - f_{i-1}}{h_{i-1}}\right)$$

We also define

$$s_{i-1} = \sigma_i s_i + \tau_i; \quad i = 1,2,\ldots,n \qquad (8)$$

Since $s_0 = 0$ then

$$\sigma_1 = \tau_1 = 0 \qquad (9)$$

Putting (8) in (7), we have:

$$s_i = \frac{-1}{\frac{h_{i-1}}{h_i}\sigma_i + 2\left(1+\frac{h_{i-1}}{h_i}\right)} s_{i+1} + \frac{d_i - \frac{h_{i-1}}{h_i}\tau_i}{\frac{h_{i-1}}{h_i}\sigma_i + 2\left(1+\frac{h_{i-1}}{h_i}\right)} \quad (10)$$

Equation (10) has the same form as equation (8). We may, therefore, write:

$$\sigma_{i+1} = \frac{-1}{\frac{h_{i-1}}{h_i}\sigma_i + 2\left(1+\frac{h_{i-1}}{h_i}\right)}; \quad \tau_{i+1} = \frac{d_i - \frac{h_{i-1}}{h_i}\tau_i}{\frac{h_{i-1}}{h_i}\sigma_i + 2\left(1+\frac{h_{i-1}}{h_i}\right)} \quad (11)$$

From which we may solve σ_i, τ_i for i=1,2,...,n since we already know [from (9)] that $\sigma_1 = \tau_1 = 0$. From these values we may find the s_i in (8) starting with $s_n = 0$ and then working backwards for *i=n-1*, *n-2*, ..., *0*. Once the s_i are known, we may find the interpolated sought for values directly from (3):

$$S_i(x) = \frac{s_i}{6h_i}(x_{i+1}-x)^3 + \frac{s_{i+1}}{6h_i}(x-x_i)^3 + \left(\frac{f_{i+1}}{h_i} - \frac{s_{i+1}h_i}{6}\right)(x-x_i) + \left(\frac{f_i}{h_i} - \frac{s_i h_i}{6}\right)(x_{i+1}-x)$$

and we need not bother to find the coefficients for the cubics $S_i(x)$, although we may do so if desired.

3. The Obtention of Spline's Coefficients from its Second Derivatives

If we find the cubics for a spline, such a spline may be expressed, as mentioned above, as follows:

$$S(x) = \sum_{i=0}^{n-1}(a_{i,i+1} + b_{i,i+1}x + c_{i,i+1}x^2 + d_{i,i+1}x^3)\delta(x)$$

$$\delta(x) = \begin{cases} 1 & x_i \le x \le x_{i+1} \\ 0 & otherwise \end{cases}; i = 0,1,...,n-1$$

The expression for a cubic collocation spline, also mentioned before, is given by:

$$S_i(x) = \frac{s_i}{6h_i}(x_{i+1} - x)^3 + \frac{s_{i+1}}{6h_i}(x - x_i)^3 + c_1(x - x_i) + c_2(x_{i+1} - x)$$

$$h_i = x_{i+1} - x_i$$

$$c_1 = \frac{f_{i+1}}{h_i} - \frac{s_{i+1}h_i}{6}$$

$$c_2 = \frac{f_i}{h_i} - \frac{s_i h_i}{6}$$

and s_i is the second derivative at the *i-th* point.

If we make

$$\alpha = x_{i+1}$$
$$\beta = x_i$$

and

$$A = \frac{s_i}{6h_i};\ B = \frac{s_{i+1}}{6h_i}$$

we may then write:

$$\begin{aligned}
S_i(x) &= A(x_{i+1} - x)^3 + B(x - x_i)^3 + c_1(x - x_i) + c_2(x_{i+1} - x) \\
&= A(\alpha^3 - 3\alpha^2 x + 3\alpha x^2 - x^3) + B(x^3 - 3\beta x^2 + 3\beta x - \beta^3) + \\
&\quad c_1 x - c_1 x_i + c_2 x_{i+1} - c_2 x \\
S_i(x) &= (B - A)x^3 + (3A\alpha - 3B\beta)x^2 + (3B\beta - 3A\alpha^2 + c_1 - c_2)x + \\
&= (A\alpha^3 - B\beta^3 - c_1 x_i + c_2 x_{i+1})
\end{aligned}$$

from which we may find the coefficients for the *(i,i+1)-th* cubic of the spline:

$$\begin{aligned}
a_{i,i+1} &= A\alpha^3 - B\beta^3 - c_1 x_i + c_2 x_{i+1} \\
b_{i,i+1} &= 3B\beta - 3A\alpha^2 + c_1 - c_2 \\
c_{i,i+1} &= 3A\alpha - 3B\beta \\
d_{i,i+1} &= B - A
\end{aligned}$$

as a function of the second derivatives of the spline at the *n+1* data points.

4. Minimum Curvature Property of Natural Splines

The conditions in (6) seem to be arbitrary. However, these conditions guarantee that the interpolant is the twice continuously differentiable one which approximates the data points with the least possible “curvature”.

Proof.

Let S(x) be the collocation spline; let g(x) be any other collocation function. Also let f(x) be the function being approximated.

We may write:

$$\int_a^b [g''(t)-S''(t)]^2\,dt = \int_a^b [g''(t)]^2\,dt - 2\int_a^b [g''(t)-S''(t)]S''(t)dt - \int_a^b [S''(t)]^2 dt \qquad (12)$$

since $a = x_0 < x_1 < ... < x_n = b$, we may express the integral from the second term to the right of the "=" sign (which we denote by T) as:

$$T = \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} S''(t)[g''(t)-S''(t)]dt \qquad (13)$$

Integration by parts ($\int u\,dv = uv - \int v\,du$) yields:

$$T = \sum_{i=0}^{n-1} \{[g'(t)-S'(t)]S''(t)\Big|_{x_i}^{x_{i+1}} - \int_{x_i}^{x_{i+1}} [g'(t)-S'(t)]S'''(t)dt\} \qquad (14)$$

Since S'''(t) on $[x_i, x_{i+1}]$ is constant (call it α_i) we may write:

$$\int_{x_i}^{x_{i+1}} [g'(t)-S'(t)]S'''(t)dt = \alpha_i [g(t)-S(t)]\Big|_{x_i}^{x_{i+1}} \qquad (15)$$

which vanishes, since

$$g(x_i) - S(x_i) = f(x_i) - f(x_i) = 0$$

Hence,

$$T = \sum_{i=0}^{n-1} \{S''(x_{i+1})[g'(x_{i+1}) - S'(x_{i+1})] - S''(x_i)[g'(x_i) - S'(x_i)]\} \qquad (16)$$

This sum reduces to the extreme values:

$$T = S''(x_n)[g'(x_n - S'(x_n)] - S''(x_0)[g'(x_0) - S'(x_0)] \qquad (17)$$

If we choose (as suggested) $S''(x_n) = S''(x_0) = 0$, then these last two terms of (17) vanish as well, and we have that

$$T = 0 \qquad (18)$$

We may then write (12) as:

$$\int_a^b [g''(t) - S''(t)]^2 dt = \int_a^b [g''(t)]^2 dt - \int_a^b [S''(t)]^2 dt \qquad (19)$$

and

$$\int_a^b [g''(t)]^2 dt = \int_a^b [S''(t)]^2 dt + \int_a^b [g''(t) - S''(t)]^2 dt \qquad (20)$$

From which finally we may conclude that

$$\int_a^b [g''(t)]^2 dt \geq \int_a^b [S''(t)]^2 dt \qquad (21)$$

Thus, from all twice continuously differentiable collocating functions, the spline with $s_0 = s_n = 0$ has the least value for the integral in (21).Intuitively, what we have found means that while a straight line would give its minimum value to the integral, *S(x)* is as close to a straight line as allowed by collocation.

4.1. Some Remarks on Curvature

The integral $\int_a^b [f''(x)]^2 dt$ is not the curvature of *f(x)* but, rather, an approximation. The true curvature is defined as:

$$K = \frac{f''(x)}{\{1 + [f'(x)]^2\}^{3/2}}$$

or, equivalently,

$$K = \frac{f''(x)}{\sqrt{\{1 + [f'(x)]^2\}^3}}$$

5. Implementation

In this section we include the pseudocode for three needed software routines:

a) Sort
b) Spline Coefficients
c) Spline Interpolation

5.1 Sort

Throughout the text we have assumed that $x_1 < x_{i+1} \quad \forall i$. To ensure this, we must first sort the original data. The following code implements a simple sorting algorithm.

procedure SORT
on input:

- n is the number of data points
- vector x holds the independent variable's values
- vector y holds the dependent variable's values
- x and y are arrays of size n (x_i, y_i)

on output:

- array x is sorted from smallest to largest
- array y has been modified such that vectors (x_i, y_i) remain unaltered

```
for i←1 to n-1
      tMIN ← xi
      iMIN ← i
```

Store the least value for the subset

```
      for j←i+1 to n
         if tMIN > xj
               tMIN←xj
               iMIN←j
         endif
      endfor
```

Put the value of the minimum element of the subset at the pivot, if needed

```
      if iMIN ≠i
            xIMIN←xi
            xi←tMIN
            tMIN←yIMIN
            yIMIN←yi
            yi←tMIN
      endif
endfor
end SORT
```

procedure SPCOEF

This subroutine calculates the values of the coefficients for the spline.

on input:

- n is the number of data points
- x holds the independent variable'spoints to be interpolated
- y holds the dependent variable's points to be interpolated

on output:

- s is the vector where the coefficients are stored

- σ and τ are auxiliary stores of size *n*

```
σ_2 ← 0
τ_2 ← 0
for i ← 2 to n-1
       h_{i-1} ← x_i - x_{i-1}
       h_i ← x_{i+1} - x_i
       temp ← (h_{i-1}/h_i) (σ_i+2)+2
       σ_{i+1} ← -1/ temp
       d=6 [(y_{i+1}-y_i)/h_i - (y_i-y_{i-1})/h_{i-1}] / h_i
       τ_{i+1} ← (d-h_{i-1}τ_i/h_i) / temp
endfor
s_1 ← 0
s_n ← 0
for i ← 1 to n-2
       i_B ← n-i
       s_{iB} ← σ_{iB+1}s_{iB+1}+τ_{iB+1}
endfor
end SPCOEF
```

procedure SPLINE
on input:

- n is the number of data points
- *x* holds the independent variable's points
- *y* holds the dependent variable's points
- *s* holds the spline's coefficients
- α is the free value to interpolate

on output:

- β is the interpolated value [$\beta \leftarrow S(\alpha)$]

```
for i ← 2 to n
       if α ≤ x_i
              exit "for loop"
       endif
endfor
i ← i-1
a ← x_{i+1}-α
b ← α -x_i
h_i ← x_{i+1}-x_i
β ← as_i (a^2/h_i-h_i)/6+bs_{i+1} (b^2/h_i-h_i)/6+(ay_i+by_{i+1})/h_i
end SPLINE
```

6. Conclusions

The preceding discussion allows us to obtain a collocating expression which is:

a) Simple to calculate
b) Economical [in that a spline needs only *n* values (the s_i) to be completely defined].
c) Stable (in that, as opposed to a collocating polynomial, it does not yield large fluctuations in the interpolated points).

In fact, we may say that a natural spline is the best way to interpolate from a set of given data when nothing more is known about such data.

6.1 A Numerical Example

We finish this discussion by presenting the s_i for the data in figure 1:

Second Derivatives for Spline

#	S
1	0.0000000000
2	-1.7087912088
3	1.3626373626
4	0.0000000000

Figure 3. Numerical values of s_i for data in figure 1.

Notice that only four values are needed. In fact, since we know the extreme ones to be 0, only 2 (in general *n-2*) values are required.

Now we use the calculated spline to interpolate selected data. Data are of course, collocated; that is $S(x_i) = y_i$ as required. These interpolated points are the ones we plotted in figure 2.

Interpolated Data

X	Y
1.0000000000	2.0000000000
1.5000000000	3.0169986264
2.0000000000	3.9271978022
2.5000000000	4.6237980769
3.0000000000	5.0000000000
3.5000000000	4.9881333944
4.0000000000	4.6770451770
4.5000000000	4.1947115385
5.0000000000	3.6691086691
5.5000000000	3.2282127595
6.0000000000	3.0000000000
6.5000000000	3.0769230769
6.7500000000	3.2165178571
7.0000000000	3.4093406593
7.5000000000	3.9120879121
8.0000000000	4.5000000000

Figure 4. Interpolated data.

Finally, we explicitly calculate the coefficients for equation (0). These are shown in figure 5.

Cubic's Coefficients

Cubic #	Segment	k	X	X^2	X^3
1	[01,02]	0.0728021978	1.6423992674	0.4271978022	-0.1423992674
2	[02,03]	-8.3791208791	8.7316849817	-2.3901098901	0.1706349206
3	[03,04]	53.0054945055	-20.5979853480	2.7252747253	-0.1135531136

Figure 5. Coefficients for the elements of *S(x)*.

6. Homogenizing Data

Our main interest in splines, from the point of view of a Neural Network, is that, sometimes would like to make data homogeneous, in the sense that, if gaps are found in these data, we would like them to vanish. In figure 6, we show interpolated data and the original points. This illustrates the way a spline collocates the original information and interpolates for the unknown values.

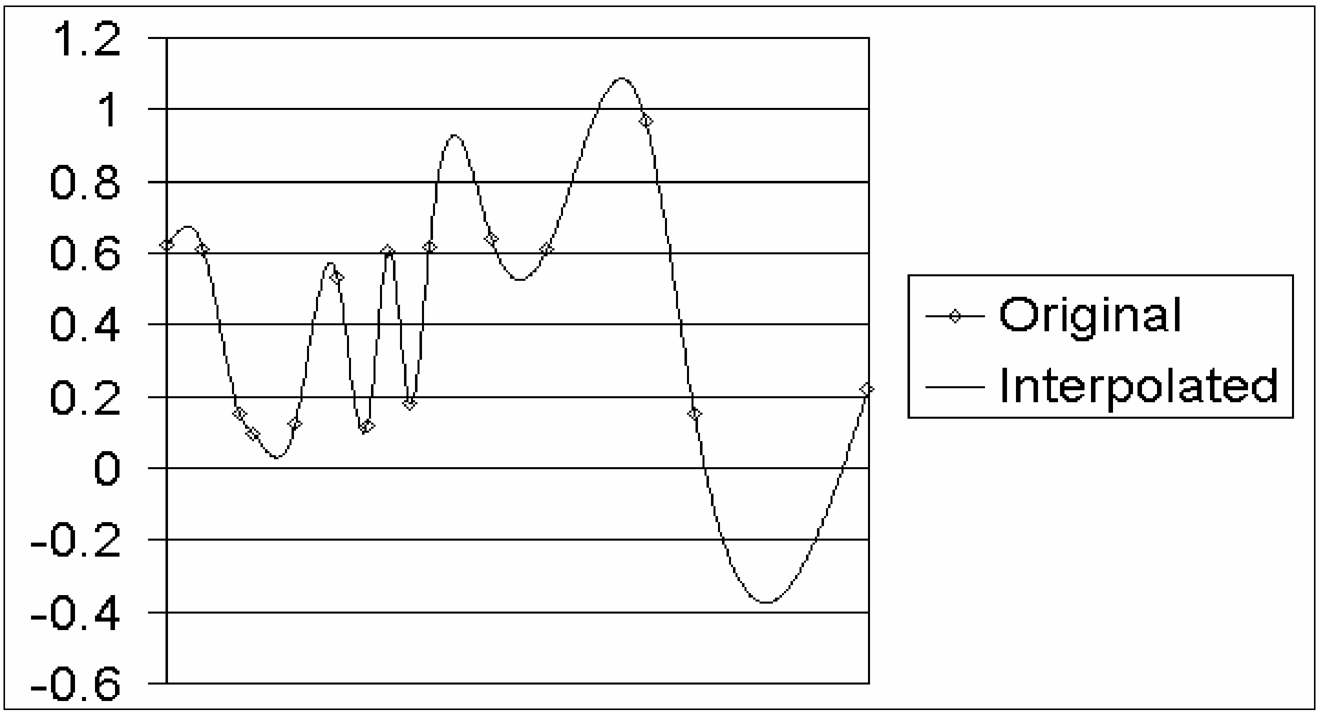
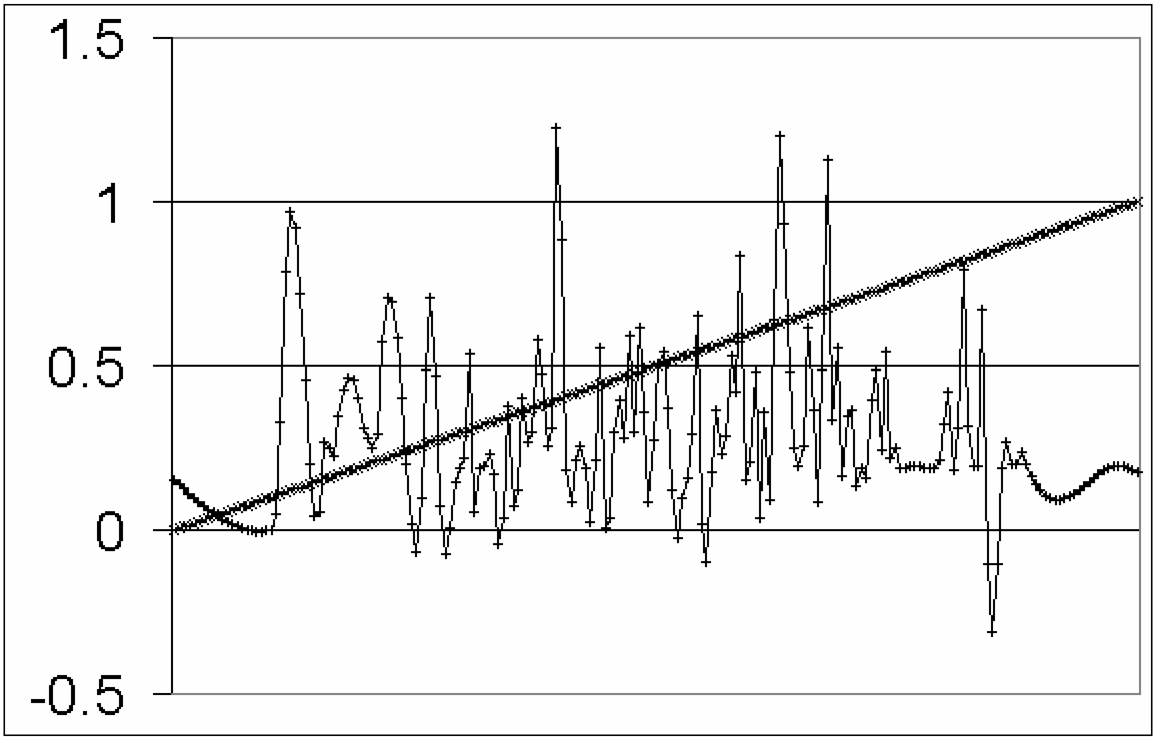
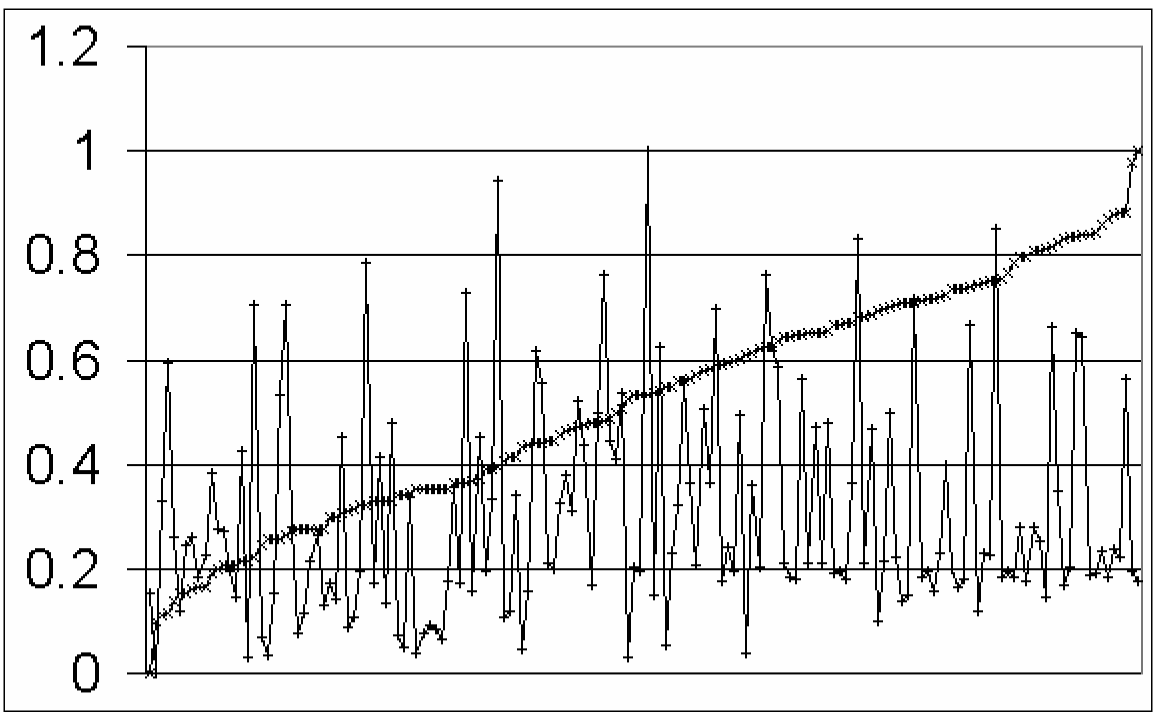


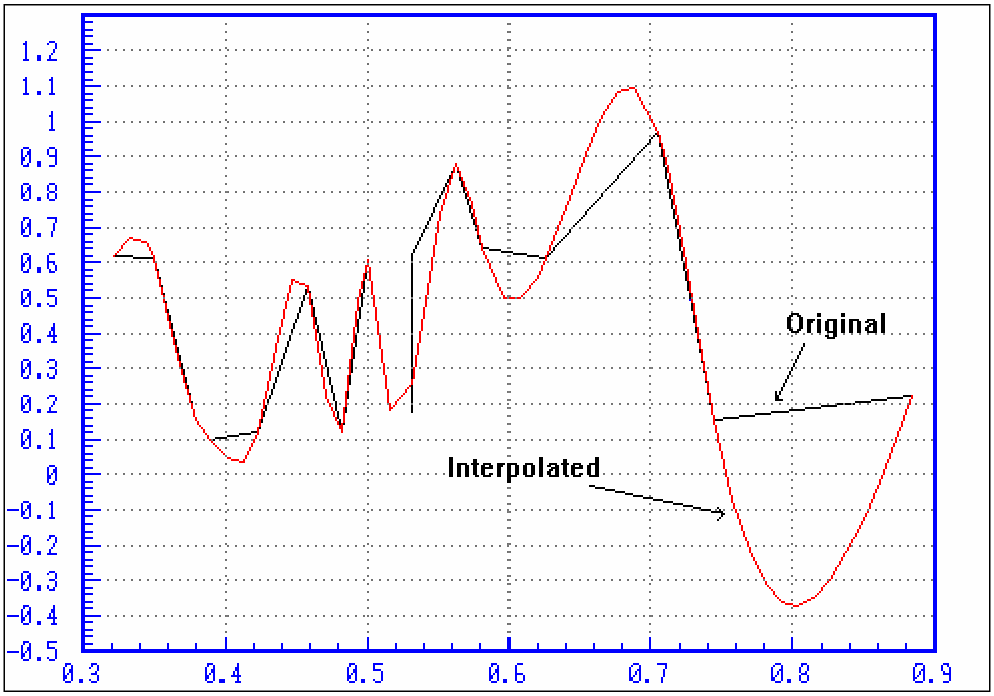
Figure 6. An Example of a Natural Spline.



Interpolated Data for X and Y



Original Data for X and Y



Interpolated and Original Data.
Interpolated points: 50
Original points: 18