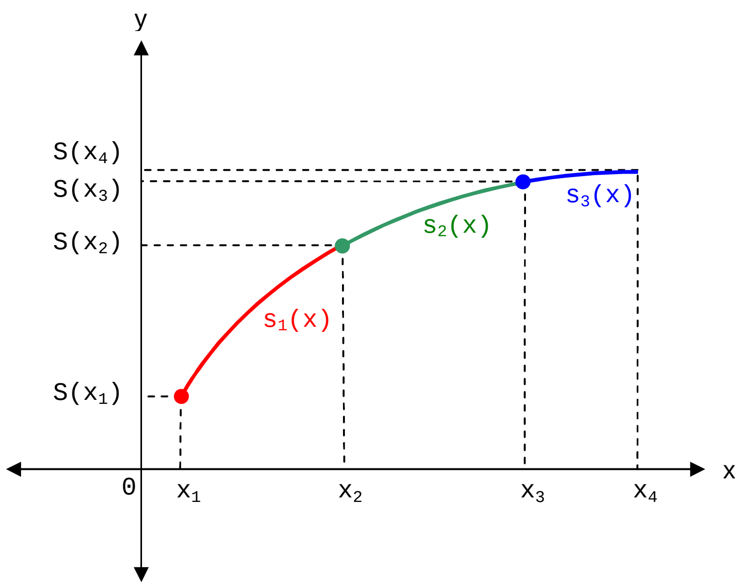
Yield Curve Construction Using Cubic Spline Interpolation

Introduction

Cubic splines are a popular choice for fitting curves to observed data, such as when constructing a yield curve. Once fit, form a smooth curve which aids interpolation. It is important to realise that observable market data, eg. Bootstrapped zero coupon rates, remain a necessary input to yield curve construction.



Consider the piecewise-defined function $S(x)$:

$$S(x)=\begin{cases} s_1(x) & \text{if} \quad x_1 \le x \le x_2 \\ s_2(x) & \text{if} \quad x_2 \le x \le x_3 \\ & \vdots \\ s_{n-1}(x) & \text{if} \quad x_{n-1} \le x \le x_n \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

Where each $s_i(x)$ is a third-order polynomial function of the form:

$$s_i(x)=a_i(x-x_i)^3+b_i(x-x_i)^2+c_i(x-x_i)+d_i \quad \ldots\ldots\ldots\ldots\ldots\ldots (2)$$

where $i=1,2,\ldots,n-1$. Note that each spline is relative to the 'base' observed data point, x_i. Also not there are n data points but $n-1$ splines.

Cubic spline interpolation is concerned with determining the unknown coefficients a_i, b_i, c_i and d_i.

Conditions

$S(x)$ is a continuous, smooth curve partly because the splines themselves are smooth and continuous functions but, more importantly, because the conditions we impose on the determination of the splines ensure $S(x)$ is smooth and continuous. These conditions are:

1. Each spline must pass through each of the observed data points.

In mathematical terms this means that for all x_i, $i = 1,\ldots,n$:

$$S(x_i) = y_i \quad \ldots\ldots\ldots\ldots \quad (3)$$

From equation (1) we can see that $S(x_i) = s_i(x_i)$ for $i = 1,\ldots,n-1$. This, combined with equations (2) and (3), gives the result:

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

However, since $(x_i - x_i) = 0$ this collapses to:

$$y_i = d_i \quad \ldots\ldots\ldots\ldots \quad (4)$$

While this holds for $i = 1,\ldots,n-1$, it is easy to see this can be extended to include the $i = n$ case.

2. Each spline must be continuous across the observed data.

At each of the interior data points (ie. all data points excluding the beginning and end points, x_1 and x_n) the value of the adjoining splines must be equal, ie.:

$$s_i(x_i) = s_{i-1}(x_i) \quad \ldots\ldots\ldots\ldots \quad (5)$$

for all x_i, $i = 2,\ldots,n-1$.

We know that $s_i(x_i) = d_i$ from above, so equation (5) implies:

$$d_i = s_{i-1}(x_i) \quad \ldots\ldots\ldots\ldots \quad (6)$$

From equation (2) we see that:

$$s_{i-1}(x_i) = a_{i-1}(x_i - x_{i-1})^3 + b_{i-1}(x_i - x_{i-1})^2 + c_{i-1}(x_i - x_{i-1}) + d_{i-1}$$

Therefore, we can show equation (6) is:

$$d_i = a_{i-1}(x_i - x_{i-1})^3 + b_{i-1}(x_i - x_{i-1})^2 + c_{i-1}(x_i - x_{i-1}) + d_{i-1}$$

Simplifying the algebra by letting $h_{i-1} = (x_i - x_{i-1})$ gives:

$$d_i = a_{i-1}h_{i-1}^3 + b_{i-1}h_{i-1}^2 + c_{i-1}h_{i-1} + d_{i-1}$$

Substituting in equation (4) gives:

$$y_i = a_{i-1}h_{i-1}^3 + b_{i-1}h_{i-1}^2 + c_{i-1}h_{i-1} + y_{i-1}$$

ie. $$c_{i-1}h_{i-1} = y_i - y_{i-1} - a_{i-1}h_{i-1}^3 - b_{i-1}h_{i-1}^2$$

ie. $$c_{i-1} = \frac{y_i - y_{i-1}}{h_{i-1}} - a_{i-1}h_{i-1}^2 - b_{i-1}h_{i-1} \quad \text{(7)}$$

for all x_i, $i = 2,\ldots,n-1$. This is a result which we will use later.

3. ***The splines must be smooth across the observed data.***

This requires the first and second derivatives of a spline at a data point, to equal the first and second derivatives of the adjacent spline at that same data point, ie.:

$$s_i^{'}(x_i) = s_{i-1}^{'}(x_i) \quad \text{(8)}$$

$$s_i^{''}(x_i) = s_{i-1}^{''}(x_i) \quad \text{(9)}$$

for all x_i, $i = 2,\ldots,n-1$.

Dealing with the second derivatives first, we can see from equation (2) that:

$$s_i^{''}(x) = 6a_i(x - x_i) + 2b_i \quad \text{(10)}$$

$$s_{i-1}^{''}(x) = 6a_{i-1}(x - x_{i-1}) + 2b_{i-1} \quad \text{(11)}$$

At the point $x = x_i$ equation (10) reduces to:

$$s_i^{''}(x_i) = 2b_i \quad \text{(12)}$$

which is an interesting result for it tells us that b_i is masquerading as the second derivative of $S(x)$ evaluated at $x = x_i$. Equations (9), (10) and (12) combine to give:

$$2b_i = 6a_{i-1}(x_i - x_{i-1}) + 2b_{i-1}$$

ie. $$b_i = 3a_{i-1}(x_i - x_{i-1}) + b_{i-1}$$

ie. $$\frac{b_i - b_{i-1}}{3(x_i - x_{i-1})} = a_{i-1}$$

As done previously, we let $h_{i-1} = (x_i - x_{i-1})$ to give:

$$a_{i-1} = \frac{b_i - b_{i-1}}{3h_{i-1}} \quad \text{(13)}$$

for all x_i, $i = 2,\ldots,n-1$.

Which is another result we will come back to.

Moving our attention to the first derivatives:

$$s_i^{'}(x) = 3a_i(x - x_i)^2 + 2b_i(x - x_i) + c_i \quad \text{(14)}$$

$$s'_{i-1}(x) = 3a_{i-1}(x - x_{i-1})^2 + 2b_{i-1}(x - x_{i-1}) + c_{i-1} \quad \dots\dots\dots\dots\dots\dots \text{(15)}$$

Again, at $x = x_i$ we can see that equation (14) collapses, this time to:

$$s'_i(x_i) = c_i$$

and this combined with equations (8) and (15) gives:

$$c_i = 3a_{i-1}(x_i - x_{i-1})^2 + 2b_{i-1}(x_i - x_{i-1}) + c_{i-1}$$

Letting $h_{i-1} = (x_i - x_{i-1})$:

$$c_i = 3a_{i-1}h_{i-1}^2 + 2b_{i-1}h_{i-1} + c_{i-1} \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \text{(16)}$$

Substituting equation (7) into (16):

$$c_i = 3a_{i-1}h_{i-1}^2 + 2b_{i-1}h_{i-1} + \frac{y_i - y_{i-1}}{h_{i-1}} - a_{i-1}h_{i-1}^2 - b_{i-1}h_{i-1}$$

ie. $$c_i = 2a_{i-1}h_{i-1}^2 + b_{i-1}h_{i-1} + \frac{y_i - y_{i-1}}{h_{i-1}}$$

Substituting in equation (13) gives:

$$c_i = 2\left(\frac{b_i - b_{i-1}}{3h_{i-1}}\right)h_{i-1}^2 + b_{i-1}h_{i-1} + \frac{y_i - y_{i-1}}{h_{i-1}}$$

ie. $$c_i = \frac{2(b_i - b_{i-1})h_{i-1}}{3} + \frac{3b_{i-1}h_{i-1}}{3} + \frac{y_i - y_{i-1}}{h_{i-1}}$$

ie. $$c_i = \frac{2b_ih_{i-1} - 2b_{i-1}h_{i-1} + 3b_{i-1}h_{i-1}}{3} + \frac{y_i - y_{i-1}}{h_{i-1}}$$

ie. $$c_i = \frac{(2b_i + b_{i-1})h_{i-1}}{3} + \frac{y_i - y_{i-1}}{h_{i-1}} \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots \text{(17)}$$

but from equation (7) we can see that:

$$c_i = \frac{y_{i+1} - y_i}{h_i} - a_ih_i^2 - b_ih_i$$

Therefore, equation (17) becomes:

$$\frac{y_{i+1} - y_i}{h_i} - a_ih_i^2 - b_ih_i = \frac{(2b_i + b_{i-1})h_{i-1}}{3} + \frac{y_i - y_{i-1}}{h_{i-1}}$$

ie. $$\frac{y_{i+1} - y_i}{h_i} - \frac{y_i - y_{i-1}}{h_{i-1}} = \frac{(2b_i + b_{i-1})h_{i-1}}{3} + a_ih_i^2 + b_ih_i$$

Using equation (13):

ie. $$\frac{y_{i+1}-y_i}{h_i}-\frac{y_i-y_{i-1}}{h_{i-1}}=\frac{(2b_i+b_{i-1})h_{i-1}}{3}+\left(\frac{b_{i+1}-b_i}{3h_i}\right)h_i^2+\frac{3b_ih_i}{3}$$

ie. $$\frac{y_{i+1}-y_i}{h_i}-\frac{y_i-y_{i-1}}{h_{i-1}}=\frac{2b_ih_{i-1}+b_{i-1}h_{i-1}}{3}+\frac{b_{i+1}h_i-b_ih_i}{3}+\frac{3b_ih_i}{3}$$

ie. $$\frac{y_{i+1}-y_i}{h_i}-\frac{y_i-y_{i-1}}{h_{i-1}}=\frac{b_{i-1}h_{i-1}+2b_ih_{i-1}-b_ih_i+3b_ih_i+b_{i+1}h_i}{3}$$

ie. $$\frac{y_{i+1}-y_i}{h_i}-\frac{y_i-y_{i-1}}{h_{i-1}}=\frac{b_{i-1}h_{i-1}}{3}+\frac{2b_ih_{i-1}+2b_ih_i}{3}+\frac{b_{i+1}h_i}{3}$$

ie. $$\frac{y_{i+1}-y_i}{h_i}-\frac{y_i-y_{i-1}}{h_{i-1}}=\frac{h_{i-1}}{3}b_{i-1}+\frac{2(h_{i-1}+h_i)}{3}b_i+\frac{h_i}{3}b_{i+1}$$

ie. $$\frac{h_{i-1}}{3}b_{i-1}+\frac{2(h_{i-1}+h_i)}{3}b_i+\frac{h_i}{3}b_{i+1}=\frac{y_{i+1}-y_i}{h_i}-\frac{y_i-y_{i-1}}{h_{i-1}}$$

ie. $$h_{i-1}b_{i-1}+2(h_{i-1}+h_i)b_i+h_ib_{i+1}=3\left(\frac{y_{i+1}-y_i}{h_i}-\frac{y_i-y_{i-1}}{h_{i-1}}\right)$$

ie. $$h_{i-1}2b_{i-1}+2(h_{i-1}+h_i)2b_i+h_i2b_{i+1}=6\left(\frac{y_{i+1}-y_i}{h_i}-\frac{y_i-y_{i-1}}{h_{i-1}}\right)$$

where $h_{i-1}=(x_i-x_{i-1})$.

which forms the linear system like so:

$$h_1 2b_1+2(h_1+h_2)2b_2+h_2 2b_3+0+0+0=6\left(\frac{y_3-y_2}{h_2}-\frac{y_2-y_1}{h_1}\right)$$

$$0+h_2 2b_2+2(h_2+h_3)2b_3+h_3 2b_4+0+0=6\left(\frac{y_4-y_3}{h_3}-\frac{y_3-y_2}{h_2}\right)$$

$$0+0+h_3 2b_3+2(h_3+h_4)2b_4+h_4 2b_5+0=6\left(\frac{y_5-y_4}{h_4}-\frac{y_4-y_3}{h_3}\right)$$

$$\vdots$$

ie. the system **Ax** = **b** where:

A is an $n-2$ x n matrix of h terms;
x is the n x 1 column vector of $2b$ coefficients: recall the relationship between $2b$ and $s_i''(x_i)$ in equation (12); and
b is an $n-2$ x 1 column vector of y and h terms.

We can already see that we have a problem: **A** is not square, therefore, it can't be inverted to solve **x** = **A**$^{-1}$**b**. Since other unknown spline coefficients (a and c) are functions of b, this means we cannot find the splines without imposing a further condition(s) on the system.

4. The first and final splines must obey the specified boundary conditions.

While the previous three conditions allowed us to solve for d_i and find a_i and c_i in terms of b_i, we were not required to make any value judgements about the splines. Unfortunately, as we have seen, these conditions were not enough to give us a workable solution. We must now make two important decisions:

$$s_i''(x_i) = ? \text{ for } i = 1$$

and

$$s_{i-1}''(x_i) = ? \text{ for } i = n$$

ie. what values for the second derivatives of the 'end points'?

It is usual to set both derivatives equal to zero, leading to a solution called 'natural' splines. This requires us to insert two rows into **A**: one at the top which consists of all zeros – save for the first column's entry which is a 1 – and one row at the bottom which consists of all zeros save for the last column's entry which is a 1. This gives **A** dimensions of n x n. We also need to add a '0' as the top entry in the **y** column vector and also as the bottom entry in the same vector, thereby making its dimensions n x 1. In effect, we have added the lines:

$$2b_1 = 0$$

and

$$2b_n = 0$$

to our system of linear equations. We can then proceed to invert **A** to get the solution to **x** and use this to solve for the a_i and c_i coefficients.

Choice of Boundary Conditions

The problem with the natural splines approach is that it may result in the wrong splines being determined. Consider the case where we fit splines to the (true) curve $y=x^3$ along the interval x = [0, 1]. The second derivative at x=0 is 6(0) = 0, so no problems there. However, at x=1 the second derivative has the value 6(1) = 6 and not 0. Accordingly, our splines will be incorrectly specified and will not match one-for-one the true function, ie. any interpolated results will not lie on the curve $y=x^3$.

For a yield curve, better boundary conditions may be set by enforcing a slope (ie. set the first derivative) at either end.

Using equation (14) recall that the first derivative at $x = x_1$ of the first spline is:

$$s_1'(x_1) = c_1$$

From equation (7) recall also that:

$$c_1 = \frac{y_2 - y_1}{h_1} - a_1 h_1^2 - b_1 h_1$$

Therefore:

$$s_1^{'}(x_1) = \frac{y_2 - y_1}{h_1} - a_1 h_1^2 - b_1 h_1$$

ie. $$s_1^{'}(x_1) = \frac{y_2 - y_1}{h_1} - \frac{b_2 - b_1}{3h_1} h_1^2 - b_1 h_1$$

ie. $$s_1^{'}(x_1) = \frac{y_2 - y_1}{h_1} - \frac{b_2 h_1 - b_1 h_1 + 3b_1 h_1}{3}$$

ie. $$\frac{b_2 h_1 + 2b_1 h_1}{3} = \frac{y_2 - y_1}{h_1} - s_1^{'}(x_1)$$

ie. $$\frac{h_1 \, 2b_2 + 2h_1 \, 2b_1}{6} = \frac{y_2 - y_1}{h_1} - s_1^{'}(x_1)$$

ie. $$h_1 \, 2b_2 + 2h_1 \, 2b_1 = 6\left(\frac{y_2 - y_1}{h_1} - s_1^{'}(x_1)\right)$$

ie. $$2h_1 \, 2b_1 + h_1 \, 2b_2 = 6\left(\frac{y_2 - y_1}{h_1} - s_1^{'}(x_1)\right)$$

which becomes the first equation in our linear system.

Turning our attention to the end point, $x = x_n$, we know from equation (15) that its first derivative is:

$$s_{n-1}^{'}(x_n) = 3a_{n-1}(x_n - x_{n-1})^2 + 2b_{n-1}(x_n - x_{n-1}) + c_{n-1}$$

ie. $$s_{n-1}^{'}(x_n) = 3a_{n-1} h_{n-1}^2 + 2b_{n-1} h_{n-1} + c_{n-1}$$

but we know from equation (7) that:

$$c_{n-1} = \frac{y_n - y_{n-1}}{h_{n-1}} - a_{n-1} h_{n-1}^2 - b_{n-1} h_{n-1}$$

Therefore:

$$s_{n-1}^{'}(x_n) = 3a_{n-1} h_{n-1}^2 + 2b_{n-1} h_{n-1} + \frac{y_n - y_{n-1}}{h_{n-1}} - a_{n-1} h_{n-1}^2 - b_{n-1} h_{n-1}$$

ie. $$s_{n-1}^{'}(x_n) - \frac{y_n - y_{n-1}}{h_{n-1}} = 2a_{n-1} h_{n-1}^2 + b_{n-1} h_{n-1}$$

$$= 2\left(\frac{b_n - b_{n-1}}{3h_{n-1}}\right)h_{n-1}^2 + b_{n-1}h_{n-1}$$

$$= \frac{2(b_n - b_{n-1})h_{n-1} + 3b_{n-1}h_{n-1}}{3}$$

$$= \frac{2b_n h_{n-1} - 2b_{n-1}h_{n-1} + 3b_{n-1}h_{n-1}}{3}$$

$$= \frac{2}{2}\left(\frac{2b_n h_{n-1} + b_{n-1}h_{n-1}}{3}\right)$$

$$= \frac{2b_n\, 2h_{n-1} + 2b_{n-1}h_{n-1}}{6}$$

ie. $$6\left(s_{n-1}^{'}(x_n) - \frac{y_n - y_{n-1}}{h_{n-1}}\right) = 2h_{n-1}\, 2b_n + h_{n-1}\, 2b_{n-1}$$

ie. $$h_{n-1}\, 2b_{n-1} + 2h_{n-1}\, 2b_n = 6\left(s_{n-1}^{'}(x_n) - \frac{y_n - y_{n-1}}{h_{n-1}}\right)$$

which is the last equation in our linear system.

Hence, by choosing $s_1^{'}(x_1)$ and $s_{n-1}^{'}(x_n)$ we can solve the linear system. For an upward sloping yield curve we may decide $s_{n-1}^{'}(x_n) = 0$ and $s_1^{'}(x_1)$ will depend on what short rates look like eg. it may be that $s_1^{'}(x_1) = 0$ is also a good choice. Either way, both choices will have an impact on all splines and, therefore, all interpolated data.

```vba
Private Function Spline( _
        years As Range, _
        rates As Range _
    ) As Variant

Dim n As Integer, _
    i As Integer, _
    j As Integer, _
    x As Variant, _
    y As Variant, _
    A As Variant, _
    b As Variant, _
    c As Variant, _
    M As Variant

n = years.Count

ReDim x(n, 1), y(n, 1)
ReDim A(n, n), b(n, 1), c(n, 1), M(n, 4)

x = years.Value
y = rates.Value

'// Initialise arrays
For i = 1 To n
    For j = 1 To n
        A(i, j) = 0
    Next j
    b(i, 1) = 0
    c(i, 1) = 0
Next i

'// Find the n-2 x n matrix 'A' and n-2 x 1 vector 'c'
For i = 2 To n - 1      '// move down the matrix/vector, row by row
    '// matrix 'A'
    For j = 1 To n      '// move across the row, element by element
        If j + 1 = i Then
            A(i, j) = x(i, 1) - x(i - 1, 1)
        ElseIf j = i Then
            A(i, j) = 2 * (x(i + 1, 1) - x(i - 1, 1))
        ElseIf j - 1 = i Then
            A(i, j) = x(i + 1, 1) - x(i, 1)
        Else
            A(i, j) = 0
        End If
    Next j
    '// vector 'c'
    c(i, 1) = 3 * ( _
              (y(i + 1, 1) - y(i, 1)) / (x(i + 1, 1) - x(i, 1)) _
              - _
              (y(i, 1) - y(i - 1, 1)) / (x(i, 1) - x(i - 1, 1)) _
              )
Next i

'// Set the Boundary Conditions
For i = 1 To n Step n - 1
    A(i, i) = 1   '// Use this for ‘normal’ splines
    c(i, 1) = 0   '// Use this for ‘normal’ splines
Next i

'// Solve for b = Inv(A) * c
b = WorksheetFunction.MMult(WorksheetFunction.MInverse(A), c)

'// Find a, b, c and d for each spline
For i = 1 To n - 1  '// Remember: there are n-1 splines
    '// a(i) =
    M(i, 1) = (b(i + 1, 1) - b(i, 1)) / (3 * (x(i + 1, 1) - x(i, 1)))
    '// b(i) =
    M(i, 2) = b(i, 1)
    '// c(i) =
    M(i, 3) = (y(i + 1, 1) - y(i, 1)) / (x(i + 1, 1) - x(i, 1)) _
                  - M(i, 1) * (x(i + 1, 1) - x(i, 1)) ^ 2 _
                  - M(i, 2) * (x(i + 1, 1) - x(i, 1))
    '// d(i)=
    M(i, 4) = y(i, 1)
Next i

Spline = M  '// Return n-1 x 4 matrix of a, b, c and d coefficients

End Function
```