**HW #4 Interpolation**

**Name: Jian Wang Fsuid: Jw09r**

1. **Executive Summary**

Generate the function of yield rate via the given eight points under the natural cubic spline algorithm and plot the yield curve and forward rate curve for twenty years.

Besides, using the existed spline curve to get the yield, discount factor and forward rate for seven years.

**The result is that the yield curve I obtained from the cubic spline algorithm is continuous and the yield rate at 7th year is 0.0898962, forward rate at the 7th year is 0.103784,discount rate at the 7th year is 0.532979**

**I. Statement of Problem**

Using the natural cubic spline algorithm to generate the yield curve and the forward rate curve under the eight given yield points. Our main tasks are as follows:

1. Implement the cubic spline algorithm. Test your implementation and show that it is working correctly.
2. Use a natural cubic spline to plot the yield curve for twenty years.
3. Use the natural cubic spline to compute and plot the forward rate curve over twenty years
4. What is the yield, discount factor and forward rate for seven years?

**II. Description of the Mathematics**

Natural cubic spline requires following condition:

On each interval [], [],…, [] , S(x) is given by a different cubic polynomial, and we have the same first derivative and second derivative at knot point from two sides.

Since is a cubic polynomial on , is a linear function satisfying and and therefore is given by the straight line between and :

Where , if this is integrated twice, then the result is

++(

Where:



The linear system for with and is symmetric, tridiagonal, diagonally dominant, and of the form

Where

Calculated using the algorithm:



We can get 

Then we can get the as follows:

Where

After we get the yield curve, from the formula given by the problem, we can get the value of forward rate f and discount factor D as follows:



III. Description of the Algorithm

The algorithm is as the follows:

/

End do

Zn=0

For i= n-1 to 1 step -1 do

End do

Output (

For i= 0 to n-1 do

End do

IV. Results

From the formula , we can know that

This is used to generate the rate of the forward rate.

**The Answer for Task One:**

With the help of software VBA, I got the result as follows:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| T(i) | Z(i) | Y(i) | A(i) | B(i) | C(i) |
| 0.5 | 0 | 0.0552 | -0.00077 | 0 | 0.009792 |
| 1.0 | -0.0023 | 0.06 | 0.000132 | -0.001149147 | 0.009217 |
| 2.0 | -0.00151 | 0.0682 | 3.5E-05 | -0.000752558 | 0.007315 |
| 4.0 | -0.00109 | 0.0801 | 1.81E-05 | -0.000542753 | 0.004725 |
| 5.0 | -0.00098 | 0.0843 | 2.03E-05 | -0.000488367 | 0.003694 |
| 10.0 | -0.00037 | 0.0931 | 8.08E-06 | -0.000183369 | 0.000335 |
| 15.0 | -0.00012 | 0.0912 | 4.14E-06 | -6.21578E-05 | -0.00089 |
| 20.0 | 0 | 0.0857 |  |  |  |

So the

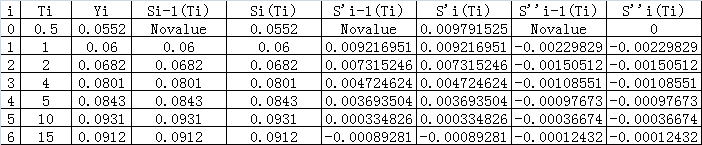
**To test if the cubic spline algorithm work correctly, I used two methods:**

**Firstly, I checked if the had the same value with the true data at each knot, I got the results as follows:**

|  |  |  |  |
| --- | --- | --- | --- |
| i | Ti | Yi | Si(Ti) |
| 0 | 0.5 | 0.0552 | 0.0552 |
| 1 | 1 | 0.06 | 0.06 |
| 2 | 2 | 0.0682 | 0.0682 |
| 3 | 4 | 0.0801 | 0.0801 |
| 4 | 5 | 0.0843 | 0.0843 |
| 5 | 10 | 0.0931 | 0.0931 |
| 6 | 15 | 0.0912 | 0.0912 |

The results showed that those has no error at each knot.

**Secondly, I tested if the following equations satisfy:**

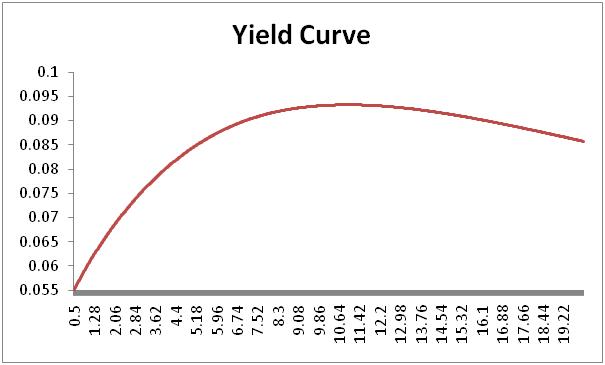
The results are as follows: 

Which showed that the algorithm is continuous on each knot.

Therefore my results to generate is correct and the method of natural cubic alpine algorithm works.

**The answer of task two:**

The figure of the yield curve is

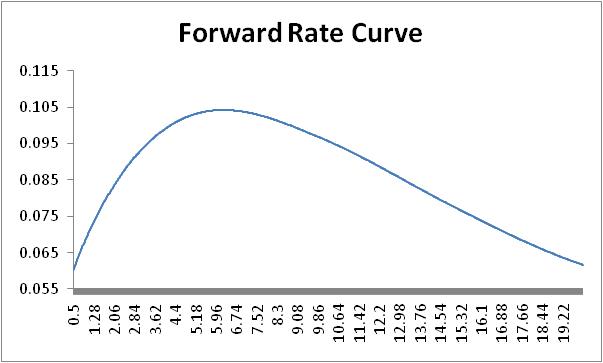


**The answer of Task three:**

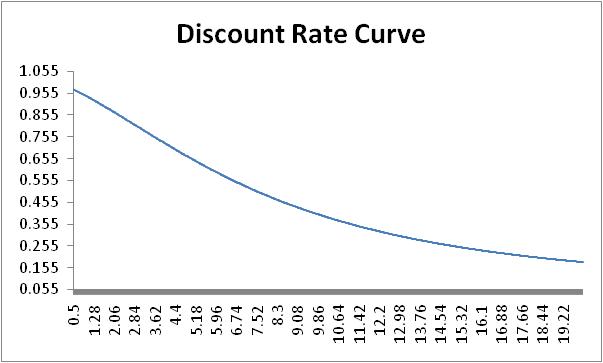
The results over twenty years are as follows, I used the distance of half a year.

|  |  |  |  |
| --- | --- | --- | --- |
| T(i) | Yield Rate | Forward rate | Discount rate |
| 0.5 | 0.0552 | 0.060095762 | 0.9727774 |
| 1 | 0.06 | 0.069216951 | 0.941764534 |
| 1.5 | 0.064337713 | 0.076588139 | 0.908003932 |
| 2 | 0.0682 | 0.082830491 | 0.872493565 |
| 2.5 | 0.071673854 | 0.088146138 | 0.835951539 |
| 3 | 0.074797655 | 0.092542753 | 0.799001092 |
| 3.5 | 0.077597629 | 0.096125238 | 0.762165389 |
| 4 | 0.0801 | 0.098998497 | 0.725858636 |
| 4.5 | 0.08232889 | 0.101208495 | 0.690402854 |
| 5 | 0.0843 | 0.102767522 | 0.656061988 |
| 5.5 | 0.086027202 | 0.103739333 | 0.623036851 |
| 6 | 0.087525471 | 0.104192092 | 0.591464967 |
| 6.5 | 0.088810056 | 0.104186799 | 0.561431582 |
| 7 | 0.089896207 | 0.103784453 | 0.532978897 |
| 7.5 | 0.090799174 | 0.103046055 | 0.506113763 |
| 8 | 0.091534208 | 0.102032604 | 0.480814537 |
| 8.5 | 0.092116557 | 0.100805098 | 0.457037089 |
| 9 | 0.092561472 | 0.099424539 | 0.434719949 |
| 9.5 | 0.092884203 | 0.097951926 | 0.413788655 |
| 10 | 0.0931 | 0.096448257 | 0.394159354 |
| 10.5 | 0.093222581 | 0.094876514 | 0.375747803 |
| 11 | 0.093259538 | 0.093175171 | 0.358489856 |
| 11.5 | 0.093216931 | 0.09136847 | 0.342324997 |
| 12 | 0.093100822 | 0.089480653 | 0.327191431 |
| 12.5 | 0.092917271 | 0.087535963 | 0.313026983 |
| 13 | 0.092672338 | 0.085558642 | 0.299769813 |
| 13.5 | 0.092372084 | 0.083572932 | 0.287358956 |
| 14 | 0.092022569 | 0.081603075 | 0.27573474 |
| 14.5 | 0.091629854 | 0.079673314 | 0.264839079 |
| 15 | 0.0912 | 0.07780789 | 0.254615682 |
| 15.5 | 0.090738575 | 0.075984787 | 0.245012044 |
| 16 | 0.090249179 | 0.074174117 | 0.235985042 |
| 16.5 | 0.089734919 | 0.072388309 | 0.227495193 |
| 17 | 0.089198905 | 0.070639796 | 0.219504755 |
| 17.5 | 0.088644243 | 0.068941009 | 0.211977688 |
| 18 | 0.088074042 | 0.06730438 | 0.204879608 |
| 18.5 | 0.087491409 | 0.065742341 | 0.198177722 |
| 19 | 0.086899452 | 0.064267322 | 0.191840766 |
| 19.5 | 0.08630128 | 0.062891756 | 0.185838927 |
| 20 | 0.062325371 | 0.235228526 | 0.287507186 |

The Forward rate curve is



The discount rate curve is



**The Answer of task Four:**

The results for seven years are as follows:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Maturity(years) | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Yield for different maturities | 0.06 | 0.0682 | 0.0747977 | 0.0801 | 0.0843 | 0.0875255 | 0.0898962 |
| Forward rate | 0.069217 | 0.0828305 | 0.0925428 | 0.0989985 | 0.102768 | 0.104192 | 0.103784 |
| Discount factor | 0.941765 | 0.872494 | 0.799001 | 0.725859 | 0.656062 | 0.591465 | 0.532979 |

**V. Conclusions**

Firstly, the natural cubic spline algorithm gave us a good approximation of the Yield.

Secondly, the yield curve we generated is continuous and had the same value of the given data at each knot.

Thirdly, the result at the 7th year are as follows:

|  |  |
| --- | --- |
| Maturity(years) | 7 |
| Yield for different maturities | 0.0898962 |
| Forward rate | 0.103784 |
| Discount factor | 0.532979 |

**VI Code Listing**

Sub interpolation()

Dim rate(8), b(8), t(8), h(8), y(8), u(8), v(8), z(8), aa(8), bb(8), cc(8), x(1000), s(2000), s1(2000), s2(2000) As Double

Dim maturity(8) As Double

Dim i As Integer

t(0) = 0.5

t(1) = 1

t(2) = 2

t(3) = 4

t(4) = 5

t(5) = 10

t(6) = 15

t(7) = 20

y(0) = 0.0552

y(1) = 0.06

y(2) = 0.0682

y(3) = 0.0801

y(4) = 0.0843

y(5) = 0.0931

y(6) = 0.0912

y(7) = 0.0857

For i = 0 To 6

h(i) = t(i + 1) - t(i)

b(i) = 6 \* (y(i + 1) - y(i)) / h(i)

Next i

u(1) = 2 \* (h(0) + h(1))

v(1) = b(1) - b(0)

For i = 2 To 6

u(i) = 2 \* (h(i) + h(i - 1)) - h(i - 1) ^ 2 / u(i - 1)

v(i) = b(i) - b(i - 1) - h(i - 1) \* v(i - 1) / u(i - 1)

Next i

z(7) = 0

For i = 6 To 1 Step -1

z(i) = (v(i) - h(i) \* z(i + 1)) / u(i)

Next i

z(0) = 0

For i = 0 To 7

Worksheets("sheet1").Select

Worksheets("sheet1").Cells(i + 2, "a").Value = z(i)

Next i

For i = 0 To 6

aa(i) = 1 / (6 \* h(i)) \* (z(i + 1) - z(i))

bb(i) = z(i) / 2

cc(i) = (-h(i)) / 6 \* z(i + 1) - h(i) / 3 \* z(i) + 1 / h(i) \* (y(i + 1) - y(i))

Worksheets("sheet1").Cells(i + 2, "b").Value = aa(i)

Worksheets("sheet1").Cells(i + 2, "c").Value = bb(i)

Worksheets("sheet1").Cells(i + 2, "d").Value = cc(i)

Next i

For i = 0 To 999

If i = 0 Then

x(i) = 0.5

Else

x(i) = x(i - 1) + 0.5

End If

Worksheets("sheet1").Cells(i + 2, "e").Value = x(i)

For j = 0 To 6

If x(i) >= t(j) And x(i) < t(j + 1) Then

s(i) = y(j) + (x(i) - t(j)) \* (cc(j) + (x(i) - t(j)) \* (bb(j) + (x(i) - t(j)) \* aa(j)))

s1(i) = cc(j) + 2 \* bb(j) \* (x(i) - t(j)) + 3 \* aa(j) \* (x(i) - t(j)) ^ 2

s2(i) = 2 \* bb(j) + 6 \* aa(j) \* (x(i) - t(j))

Worksheets("sheet1").Cells(i + 2, "f").Value = s(i)

Worksheets("sheet1").Cells(i + 2, "g").Value = s1(i)

Worksheets("sheet1").Cells(i + 2, "h").Value = s2(i)

End If

Next j

Next i

End Sub