Spline Interpolation: Cubic Spline Interpolation (CSI)

Lecture-2

Objectives

- Introduce MATLAB Spline Interpolation Functions
- Solve problems by using Cubic Spline Interpolation
- Draw curve by applying MATLAB Spline Interpolation Functions

Conditions for Cubic Spline

Let us consider the cubic spline

$$f(x) = \begin{cases} f_1(x) & x_1 < x < x_2 \\ f_2(x) & x_2 < x < x_3 \\ f_3(x) & x_3 < x < x_4 \end{cases}$$

At the common point $x = x_2$,

$$f_1(x_2) = f_2(x_2)$$
$$f_1'(x_2) = f_2'(x_2)$$
$$f_1''(x_2) = f_2''(x_2)$$

At the common point $x = x_3$,

$$f_2(x_3) = f_3(x_3)$$
$$f_2'(x_3) = f_3'(x_3)$$
$$f_2''(x_3) = f_3''(x_3)$$

Boundary conditions for Cubic Spline

Let us consider the cubic spline

$$f(x) = \begin{cases} f_1(x) & x_1 < x < x_2 \\ f_2(x) & x_2 < x < x_3 \\ f_3(x) & x_3 < x < x_4 \end{cases}$$

Normally we use three types of conditions

- Natural Cubic Spline
- Clamped Cubic Spline
- Not-a-Knot Cubic Spline

<u>Natural Cubic Spline</u>

Second derivatives at the end points are zero.

$$f_1''(x_1) = 0,$$
$$f_3(x_4) = 0$$

<u>Clamped Cubic Spline</u>

First derivatives at the end points are known.

$$f_1'(x_1) = d_1,$$
$$f_3'(x_4) = d_2$$

<u>Not-a-knot Cubic Spline</u>

Automatically adjusted boundary conditions known as **not-a-knot** cubic spline.

$$f_1'''(x_2) = f_2'''(x_2)$$
$$f_2'''(x_3) = f_3'''(x_3)$$

MATLAB Spline Interpolation Functions

- MATLAB function **spline**

  yy=spline(x, Y, xx)　　　　　for **not-a-knot** cubic spline

  x, Y are inputs and xx is expolant.

  yy=spline(x, [dY0, Y, dYn], xx)　　　for clamped cubic spline

$$dY0 = Y'(x0) \text{and} \quad dYn = Y'(xn)$$

- **csape** spline interpolation with various end conditions

Syntax: sp=csape(X, Y, conds)

some of the conditions are

1. 'second' adjusted second derivatives if not mentioned it uses $[0, 0]$
2. 'clamped' adjusted first derivatives
3. 'not-a-knot' uses not-a-knot condition

Solve Problems

Example #1: A natural cubic spline is defined by

$$f(x) = \begin{cases} A(x+1)^3 + B(x+1)^2 + C(x+1) + 1, & -1 \le x < 1 \\ D(x-1)^3 + 6(x-1)^2 + E(x-1) - 1, & 1 \le x \le 2 \end{cases}$$

I. Use continuity and boundary conditions to estimate A, B, C, D and E.

II. Find the value of $f(1.4)$ from the spline curve'

III. Use MATLB function "**sp=csape(x, y, 'conditions')**" to construct natural cubic spline for the data set $(-1, 1), (1, -1)$ and $(2, 10)$.

Find $f(1.4)$ using "**fnval(**sp,**x)**", and Plot the spline curve using "**fnplt(sp)**" along with the data points.

Solution

Let $f_1(x) = A(x+1)^3 + B(x+1)^2 + C(x+1) + 1, \quad -1 \le x < 1$

and $f_2(x) = D(x-1)^3 + 6(x-1)^2 + E(x-1) - 1, \quad 1 \le x \le 2$

Differentiating

$$f_1'(x) = 3A(x+1)^2 + 2B(x+1) + C$$
$$f_2'(x) = 3D(x-1)^2 + 12(x-1) + E$$

and

$$f_1''(x) = 6A(x+1) + 2B$$
$$f_2''(x) = 6D(x-1) + 12$$

Conditions at the common point $x = 1$

$$f_1(1) = f_2(1) \quad \Longrightarrow \quad 8A + 4B + 2C + 1 = -1 \qquad (1)$$

$$f_1'(1) = f_2'(1) \quad \Longrightarrow \quad 12\,A + 4B + C = E \qquad (2)$$

$$f_1''(1) = f_2''(1) \quad \Longrightarrow \quad 12A + 2B = 12 \qquad (3)$$

At the common point $x = 1$, $f(x), f'(x) and\ f''(x)$ are continuous.

For **natural cubic spline** the boundary conditions give

$$f_1''(-1) = 0 \qquad 2B = 0 \ \rightarrow B = 0$$

$$f_2''(2) = 0 \qquad 6D + 12 = 0 \ \rightarrow D = -2$$

From (3), $\quad 12A + 2B = 12 \quad \rightarrow \qquad A = 1$

From (1), $\quad 8A + 4B + 2C + 1 = -1 \quad \rightarrow \qquad C = -5$

From (2), $\quad 12\,A + 4B + C = E \quad \rightarrow$ $\qquad$ E=7

The natural cubic spline function is

$$f(x) = \begin{cases} (x+1)^3 - 5(x+1) + 1\,, & -1 \le x < 1 \\ -2(x-1)^3 + 6(x-1)^2 + 7(x-1) - 1, & 1 \le x \le 2 \end{cases}$$

(ii) $f(x) = -2(x-1)^3 + 6(x-1)^2 + 7(x-1) - 1, \quad 1 \le x \le 2$

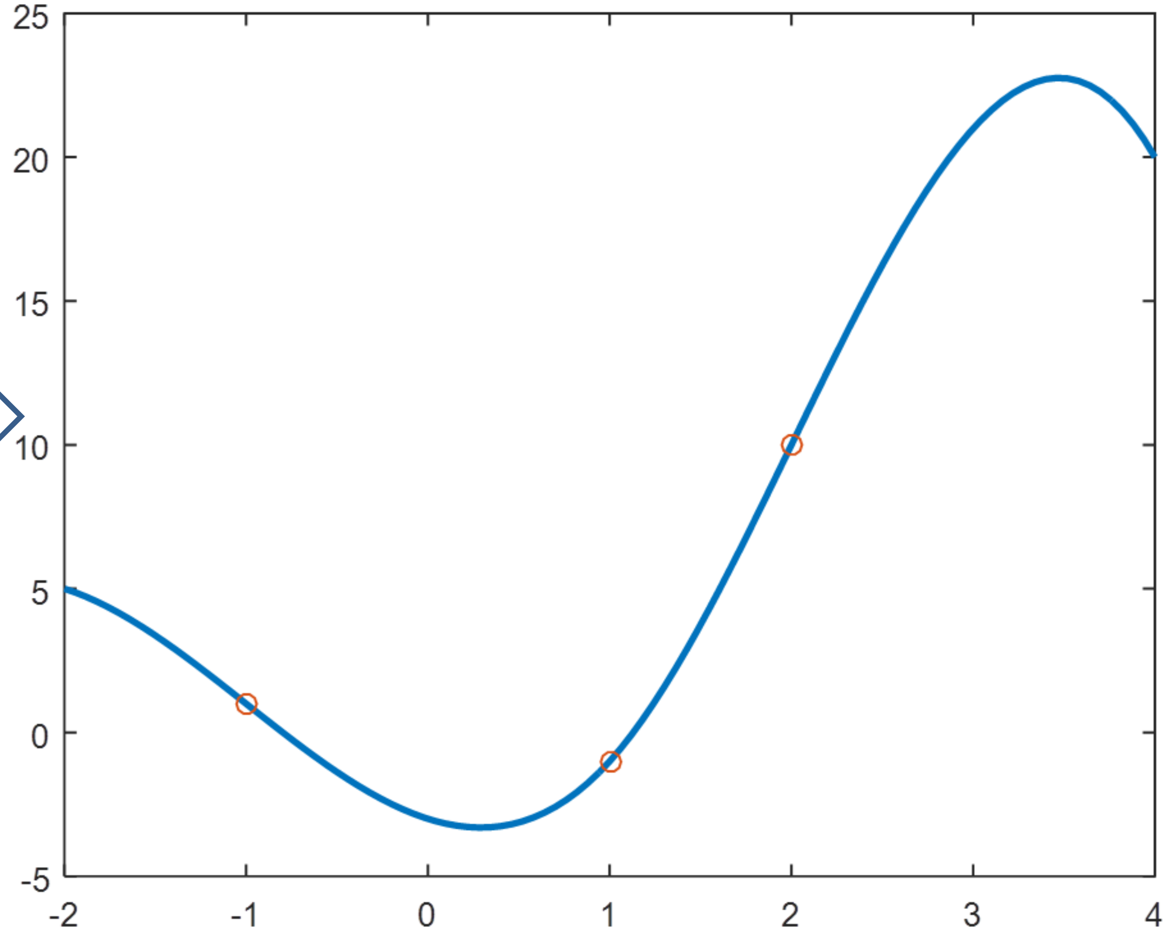
$$f(1.4) = f_2(1.4) = \quad -2(0.4)^3 + 6(0.4)^2 + 7(0.4) = 2.632.$$

iii.

```
>> clear
>> x=[-1 1 2];
>> y=[1 -1 10];
>> sp=csape(x,y,'second');
>> y1=fnval(sp,1.4)

y1 =
   2.6320
```

```
>> fnplt(sp, [-2, 4])
>> hold on                          % used to plot in the same figure
>> plot(x, y,'O')
>> hold off
```

Cubic spline curve



Example #2: A cubic spline $f(x)$ which interpolates the data $(1,-10),(2,-6),(4,2),(5,18)$ is defined by

$$f(x)=\begin{cases}(x-1)^3+b_1(x-1)^2+c_1(x-1)-10, & 1\le x<2\\ a_2(x-2)^3-(x-2)^2+c_2(x-2)-6, & 2\le x<4\\ a_3(x-4)^3+b_3(x-4)^2+10(x-4)+2, & 4\le x\le 5\end{cases}$$

i. If the spline satisfies the not-a-knot boundary conditions, find a_2 and a_3.

ii. Use MATLAB function “**sp=spline(x, y)**” to construct the spline curve and find coefficients by **“sp.coefs**”.

iii. Write down MATLAB codes using “**fnval(sp,x)**” to estimate the values of $f(x)$ for $x = 1.4, 2.5$ and 4.8 from the spline curve.

iv. Write down MATLAB codes using “**fnplt(sp)**” to plot the spline curve and the given data points.

Solutions:

Let the cubic spline be $f(x) = \begin{cases} f_1(x), & 1 \le x < 2 \\ f_2(x), & 2 \le x < 4 \\ f_3(x), & 4 \le x \le 5 \end{cases}$

where

$$f_1(x) = (x-1)^3 + b_1(x-1)^2 + c_1(x-1) - 10,$$
$$f_2(x) = a_2(x-2)^3 - (x-2)^2 + c_2(x-2) - 6,$$
$$f_3(x) = a_3(x-4)^3 + b_3(x-4)^2 + 10(x-4) + 2.$$

Then

$$f_1{}'(x) = 3(x-1)^2 + 2b_1(x-1) + c_1,$$
$$f_2{}'(x) = 3a_2(x-2)^2 - 2(x-2) + c_2,$$
$$f_3{}'(x) = 3a_3(x-4)^2 + 2b_3(x-4) + 10.$$

And

$$f_1''(x) = 6(x-1) + 2b_1,$$
$$f_2{}''(x) = 6a_2(x-2) - 2,$$
$$f_3''(x) = 6a_3(x-4) + 2b_3.$$

Also

$$f_1'''(x) = 6, \quad f_2'''(x) = 6a_2, \quad f_3'''(x) = 6a_3.$$

i. Spline curve satisfies not-a-knot boundary conditions. Thus

$$f_1'''(2) = f_2'''(2) \quad \Rightarrow \quad 6 = 6a_2 \quad \text{or} \;\; a_2 = 1.$$

$$f_3'''(4) = f_2'''(4) \quad or \quad 6a_3 = 6a_2 = 6 \quad or \quad a_3 = 1.$$

ii.

$$f(x) = \begin{cases} f_1(x), & 1 \le x < 2 \\ f_2(x), & 2 \le x < 4 \\ f_3(x), & 4 \le x \le 5 \end{cases}$$

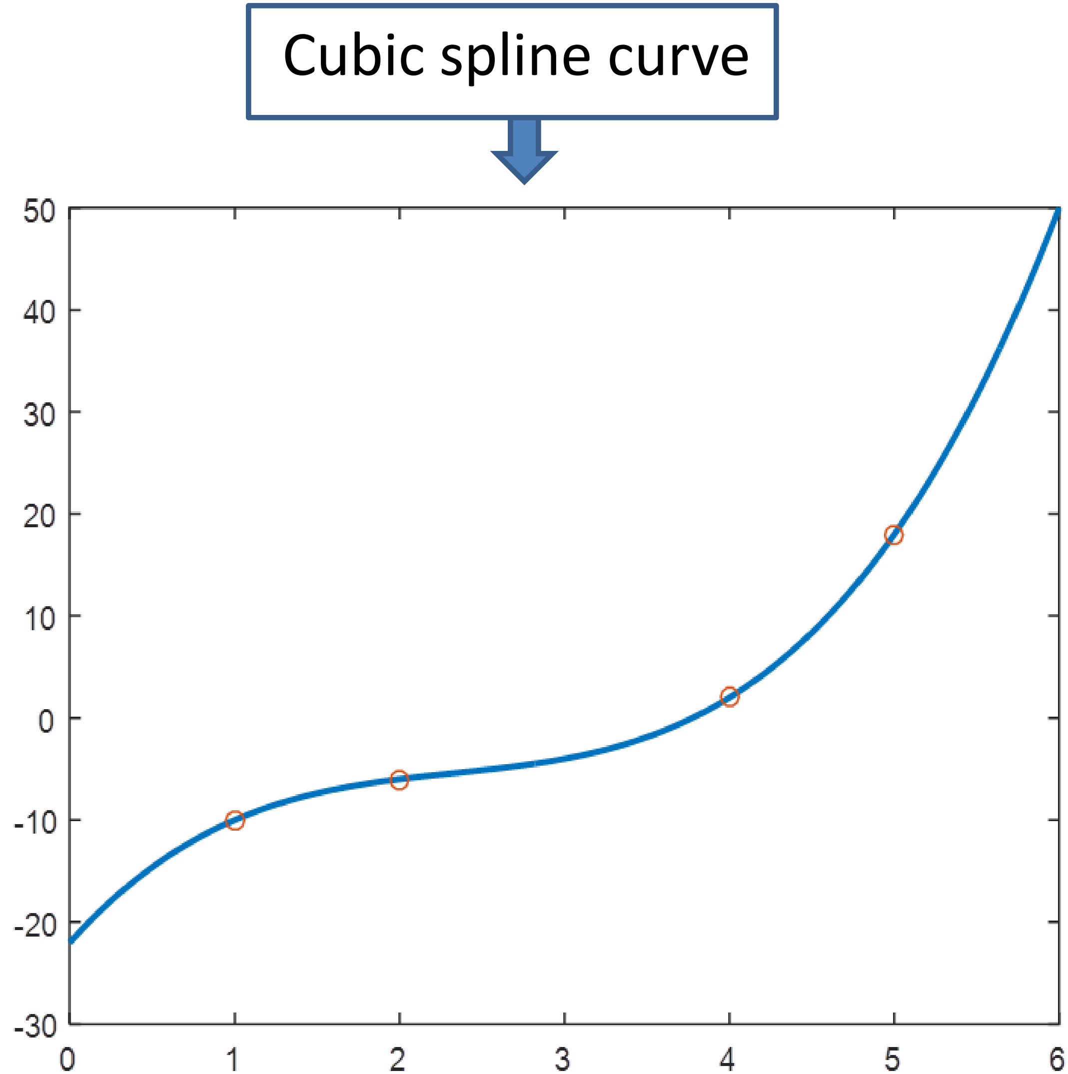
```
>> x=[1 2 4 5];
>> y=[-10 -6 2 18];
>> sp=spline(x,y);
>> format short g
>> Coefficients = sp.coefs

Coefficients =
        1      -4       7     -10
        1      -1       2      -6
        1       5      10       2
```

iii.
```
>> x1=[1.4, 2.5, 4.8];
>> y1=fnval(sp,x1);
>> xy_value =[x1', y1']
```

xy_value =

1.4	-7.776
2.5	-5.125
4.8	13.712

iv.
```
>> fnplt(sp, [0, 6])
>> hold on
>> plot(x,y,'o')
```



Outcomes

- **Functions can be derived by using Cubic Spline Interpolation (CSI) from given data sets.**
- **Functions can be plotted by using built in function in MATLAB for Cubic Spline Interpolation (CSI).**

Multiple questions:

S.No.	Questions
1	Which command can be used to find coefficients after constructing spline curve in MATHLAB by? (a) sp.coefs, (b) esape(x, y, ‘conditions’), (c) None of them
2	By using Cubic Spline Interpolation (CSI) from available data sets- (a) Functions can be derived, (b) Functions can not be derived, (c) None of them, (d) Both of them
3	By applying built in function in MATLAB for Cubic Spline Interpolation - (a) Functions can be plotted, (b) Functions can not be plotted , (c) Both of them
4	Which command can be used for to construct natural cubic spline for the given data set? (a) csape(x, y, ‘conditions’), (b) esape(x, y, ‘conditions’), (c) None of them

Try to do yourself

Exercise 1: A natural cubic spline $f(x)$ which interpolates the data points $(-1, 5), (2, 2), (4, 60)$ is defined by

$$f(x) = \begin{cases} A(x+1)^3 + B(x+1)^2 + C(x+1) + 5, & -1 \le x < 2 \\ D(x-1)^3 + E(x-1)^2 + 17(x-1) + 2, & 2 \le x \le 4 \end{cases}$$

i. Use continuity and boundary conditions to find the values of *A, B, C, D* and *E*.

ii. Estimate $f(1)$ and $f(3)$ from .the spline curve.

Exercise 2: A clamped cubic spline function through (-1,1), (2,-2) (4,30) is defined by

$$f(x) = \begin{cases} A(x+1)^3 + 3(x+1)^2 + B(x+1) + 1, & -1 \le x < 2 \\ C(x-2)^3 + D(x-2)^2 + E(x-2) - 2, & 2 \le x \le 4 \end{cases}$$

i. Given that $f'(-1) = 1$ and $f'(4) = 30$, find the values of *A, B, C,D* and *E*.

ii. Estimate the value of $f(2.5)$

iii. Write a MATLAB code using function “**spline(x, y)**” to construct the spline curve and “**fnval(function,x)**” to estimate the values of $f(x)$ for $x = -0.5, 2.5 \text{ and } 3.8$

References

[1] Applied Numerical Methods With Matlab for Engineers and Scientists (Steven C.Chapra).

[2] Applied Numerical Analysis – C.F.Gerald & P.O.Wheatley, 7th Edition, 2003, Pearson Education Limited, USA.

[3] Numerical Analysis & Computing – W. Cheney & D. Kincaid, 6th Edition, 2007, Cengage Learning, Inc, USA.

[4] Numerical Analysis – J. Douglas Faires , Annette Burden , Richard Burden, 10th Edition, 2015, Cengage Learning, Inc, USA.