15/15

Cristian B. Jetomo

MATH 174 B2L

Exercise 3.2: Cubic Splines

In this problem, we shall derive a cubic spline $s(x)$ that approximates the given function

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \tag{1}$$

on the interval $[-3,3]$ such that $s(x)$ has the minimum curvature. You may still put it here.

By Theorem 4.2, for $s(x)$ to have the minimum curvature, then $s(x)$ must be the clamped cubic spline which, as defined, must be satisfy Definition 4.2 and the additional imposed conditions

$$s'(-3) = f'(-3) \tag{2}$$

and

$$s'(3) = f'(3) \tag{3}$$

Moreover, given the function f in (1), we obtain using an online calculator that

$$f^{(4)}(x) = -\frac{-x^4 e^{-\frac{x^2}{2}} + 6x^2 e^{-\frac{x^2}{2}} - 3e^{-\frac{x^2}{2}}}{\sqrt{2\pi}} \tag{4}$$

which is continuous over the interval $[-3,3]$. Hence, by Theorem 4.3, it follows that the L^∞ norm of the clamped cubic spline s is given by

$$\begin{aligned} \|s(x)\|_\infty &= \max_{x\in[-3,3]} |f(x) - s(x)| \\ &\le \frac{5}{384} h^4 \max_{x\in[-3,3]} \left| f^{(4)}(x) \right| \end{aligned} \tag{5}$$

where $h = x_{i+1} - x_i$ for $i = 1,2,\ldots,n$ (uniform increment) and where $n+1$ is the number of abscissas to which s is relative to.

avoid ending sentences w/ prepositions

Now, using equation (5), we want to find h so that the L^∞ norm of s does not exceed 0.0156. That is,

(1pt)

$$\begin{aligned} \|s(x)\|_\infty &\le 0.0156 \\ \frac{5}{384}h^4 \max_{x\in[-3,3]} \left|f^{(4)}(x)\right| &\le 0.0156 \quad ✓ \\ h^4 &\le \frac{(384)(0.0156)}{(5)\max_{x\in[-3,3]}\left|f^{(4)}(x)\right|} \\ h &\le \left|\sqrt[4]{\frac{(384)(0.0156)}{(5)\max_{x\in[-3,3]}\left|f^{(4)}(x)\right|}}\right| \end{aligned} \tag{6}$$

Plotting the absolute value of the function in equation (4), we obtain the illustration below.

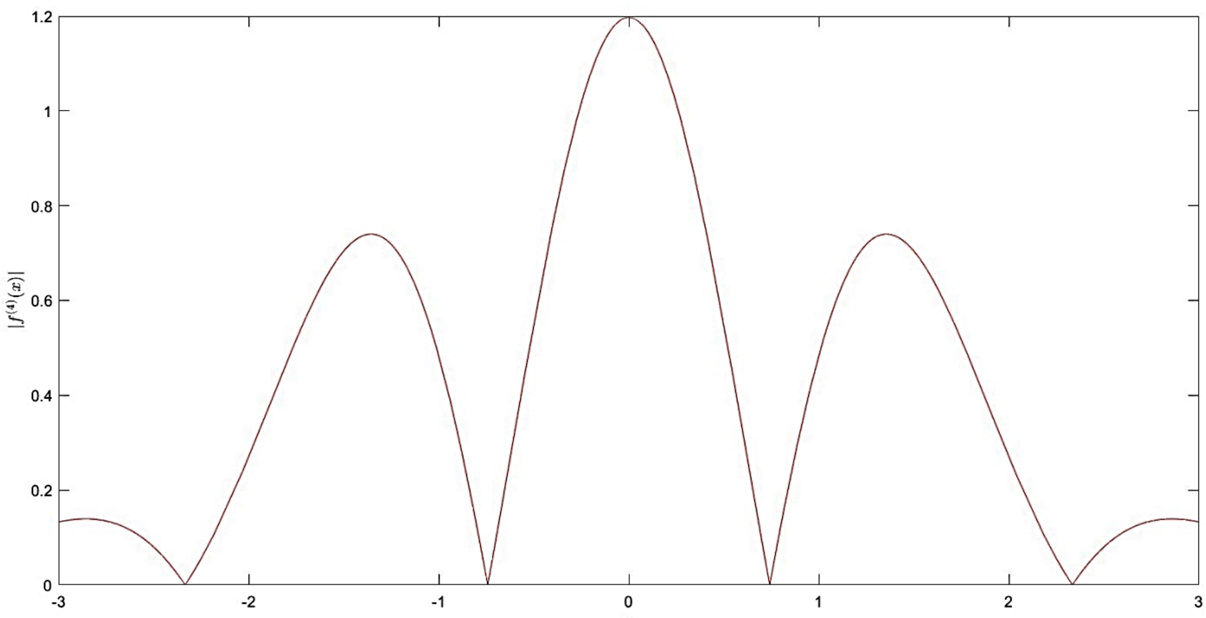

Figure 3.2.1. Graph of the function $\left|f^{(4)}(x)\right|$ over the interval $[-3,3]$.

As observed, the graph is symmetric about the y-axis and is maximum at $x = 0$. Hence, from the final line in equation (6), we obtain that the uniform increment h is given by

$$h \le \left|\sqrt[4]{\frac{(384)(0.0156)}{(5)\left|f^{(4)}(0)\right|}}\right| \tag{7}$$

Solving for h using equation (7) in MATLAB, we obtain its value to be

(5pts)

```
h =
   1.000261664218997
```

Rounding this value down to two decimal places, we obtain that $h = 1.00$.

This implies that if we let $h = 1.00$ as the step-size of the abscissas to create the clamped cubic spline $s(x)$, then it guarantees that the L^∞ norm of s will not exceed 0.0156. Consequently, the interpolatory abscissas for the interpolant s are provided in the following table:

Table 3.2.1 Interpolatory data points (x, y) for the clamped cubic spline $s(x)$.

x	-3.00	-2.00	-1.00	0.00	1.00	2.00	3.00
y	4.43$\times 10^{-3}$	5.40$\times 10^{-2}$	2.42$\times 10^{-1}$	3.99$\times 10^{-3}$	2.42$\times 10^{-1}$	5.40$\times 10^{-2}$	4.43$\times 10^{-3}$

From the table, we obtain that the number of abscissas is $n + 1 = 7$ and thus, $n = 6$.

Now, we shall construct $s(x)$ using the formula

$$s_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \tag{8}$$

For $i = 1, 2, \ldots, n$ and where $x \in [-3, 3]$. Also, by definition of cubic splines, we have the equations

$$a_i = y_i \tag{9}$$

$$b_i = \frac{a_{i+1} - a_i}{h} - \frac{2c_i + c_{i+1}}{3}h \tag{10}$$

and

$$d_i = \frac{c_{i+1} - c_i}{3h} \tag{11}$$

Note that we used h instead of the conventional h_i in equations (9), (10), and (11) since we set h to be the uniform increment of the abscissas which implies that $h_i = h$ for $i = 1, 2, \ldots, n$.

Now, also as defined, the values of c_i are obtained by solving for x in the matrix equation $Ax = b$ given by

$$\begin{bmatrix} 2h & h & 0 & \cdots & 0 & \cdots & 0 \\ h & 2(h+h) & h & 0 & 0 & \cdots & 0 \\ 0 & h & 2(h+h) & h & 0 & \cdots & 0 \\ & & & \vdots & & & \\ 0 & 0 & 0 & \cdots & h & 2(h+h) & h \\ 0 & 0 & 0 & \cdots & 0 & h & 2h \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{n-1} \\ c_n \\ c_{n+1} \end{bmatrix} = \begin{bmatrix} \frac{3(a_2 - a_1)}{h} - 3f'(-3) \\ \frac{3(a_3 - a_2)}{h} - \frac{3(a_2 - a_1)}{h} \\ \frac{3(a_4 - a_3)}{h} - \frac{3(a_3 - a_2)}{h} \\ \vdots \\ \frac{3(a_{n+1} - a_n)}{h} - \frac{3(a_n - a_{n-1})}{h} \\ 3f'(3) - \frac{3(a_{n+1} - a_n)}{h} \end{bmatrix} \tag{12}$$

Note that the matrices A and b, respectively, in equation (12) can be simplified as follows:

$$\begin{bmatrix} 2h & h & 0 & \cdots & 0 & \cdots & 0 \\ h & 4h & h & 0 & 0 & \cdots & 0 \\ 0 & h & 4h & h & 0 & \cdots & 0 \\ & & & \vdots & & & \\ 0 & 0 & 0 & \cdots & h & 4h & h \\ 0 & 0 & 0 & \cdots & 0 & h & 2h \end{bmatrix}$$

and

(2pts)

$$\begin{bmatrix} \frac{3(a_2-a_1)}{h} - 3f'(-3) \\ \frac{3(a_3-2a_2+a_1)}{h} \\ \frac{3(a_4-2a_3+a_2)}{h} \\ \vdots \\ \frac{3(a_{n+1}-2a_n+a_{n-1})}{h} \\ 3f'(3) - \frac{3(a_{n+1}-a_n)}{h} \end{bmatrix}$$

Using MATLAB, we solve for the constraints a_i's which are exactly the y-row of Table 3.2.1. Afterwards, we can solve for the constants $c_i's$ by solving for x in the matrix equation (12). Lastly, $b_i's$ and $d_i's$ can be obtained using equations (10) and (11), respectively. The summary of the obtained values of the constants are tabulated below.

Table 3.2.2. Values of the constants a_i, b_i, c_i and d_i (rounded to 4 decimal places) required by the formula in equation (8) for $s_i(x)$ for $i = 1,2,..,n$.

i	a_i	b_i	c_i	d_i
1	4.43×10^{-3}	1.33×10^{-2}	4.59×10^{-3}	3.17×10^{-2}
2	5.40×10^{-2}	1.17×10^{-1}	9.96×10^{-2}	-2.91×10^{-2}
3	2.42×10^{-1}	2.29×10^{-1}	1.22×10^{-2}	-8.46×10^{-2}
4	3.99×10^{-3}	0.00	-2.42×10^{-1}	8.46×10^{-2}
5	2.42×10^{-1}	-2.29×10^{-1}	1.22×10^{-2}	2.91×10^{-2}
6	5.40×10^{-2}	-1.17×10^{-1}	9.96×10^{-2}	-3.17×10^{-2}

show the actual A and b matrices.

Using the constants from Table 3.2.2 and ~~applying~~ using them ~~to~~ in equation (8), we obtain the clamped cubic spline $s(x)$ to be

(3pts)
$$s(x) = \begin{cases} 3.167 \times 10^{-2}x^3 + 2.896 \times 10^{-1}x^2 + 8.960 \times 10^{-1}x + 9.408 \times 10^{-1} & -3.00 \leq x \leq -2.00 \\ -2.912 \times 10^{-2}x^3 - 7.514 \times 10^{-2}x^2 + 1.664 \times 10^{-1}x + 4.544 \times 10^{-1} & -2.00 \leq x \leq -1.00 \\ -8.460 \times 10^{-2}x^3 - 2.416 \times 10^{-1}x^2 - 1.041 \times 10^{-17}x + 3.989 \times 10^{-1} & -1.00 \leq x \leq 0.00 \\ 8.460 \times 10^{-2}x^3 - 2.416 \times 10^{-1}x^2 + 3.989 \times 10^{-1} & 0.00 \leq x \leq 1.00 \\ 2.912 \times 10^{-2}x^3 - 7.514 \times 10^{-2}x^2 - 1.664 \times 10^{-1}x + 4.544 \times 10^{-1} & 1.00 \leq x \leq 2.00 \\ -3.167 \times 10^{-2}x^3 + 2.896 \times 10^{-1}x^2 - 8.960 \times 10^{-1}x + 9.408 \times 10^{-1} & 2.00 \leq x \leq 3.00 \end{cases}$$

Furthermore, using the equation above, we obtain its plot, together with the graph of $f(x)$ to be as follows:

(3pts)

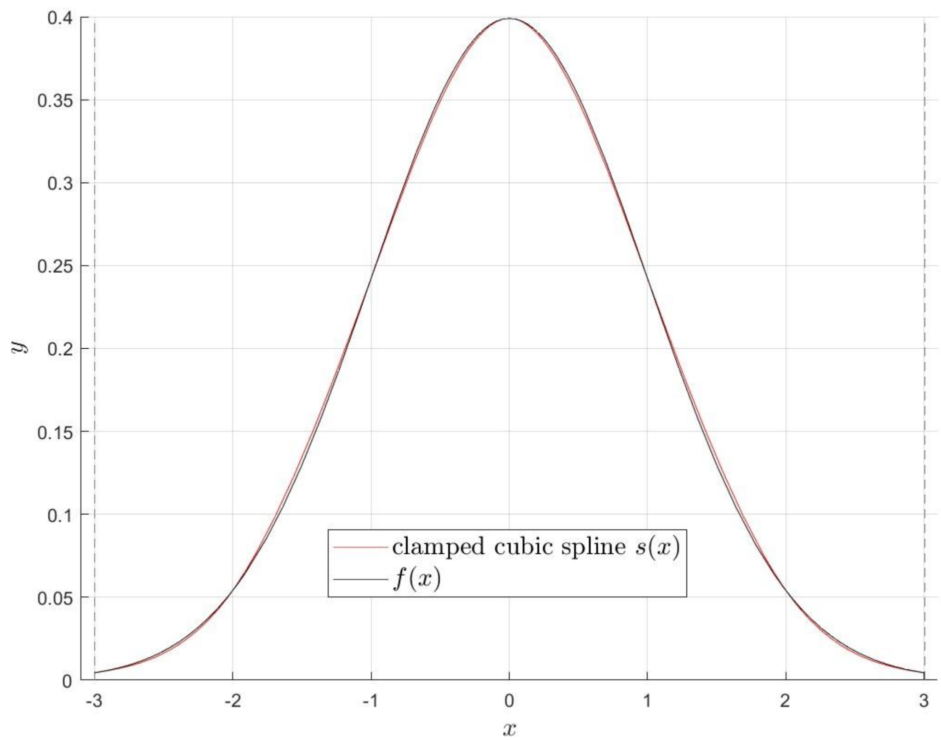

Figure 3.2.2. Graph of the function $f(x)$ together with the graph of the generated clamped cubic spline $s(x)$ approximating f over the interval $[-3, 3]$.

Observe from the figure that the graph of s is relatively similar to the graph of f which implies that our obtained clamped cubic spline effectively approximates f. Notice also that there are visible slight discrepancies in the two graphs over the interval $[-3, 3]$ but is minimized. Note that we constructed a cubic spline with a minimum curvature by only having to impose additional conditions which required values of the first derivative of the function at the endpoints we concern in our interpolation.

(1pt)

Annex A

Program used in MATLAB for Exercise 3.2

```
clc
clear
close all

syms x real
syms S
%% Function and Necessary Constants
f = 1/(sqrt(2*pi))*exp(-(x^2/2));
interval =[-3, 3];

%% Finding increment h
ub = 0.0156;  % upper bound L^infty norm
f4Prime = diff(f, 4);
c = 5/384;

% plot of f4Prime
figure
fplot(abs(f4Prime), interval, 'r')
xlabel("$x$", 'Interpreter', 'Latex')
ylabel("$|f^{(4)}(x)|$", 'Interpreter', 'Latex')

% maximum value of abs(f4Prime) occurs at x=0
maxf4Prime = double(abs(subs(f4Prime, 0)));

% calculation of h (uniform increment)
h = abs(nthroot((ub)/(c*maxf4Prime), 4));
disp('h = ')
disp(h)

h = 1.00;  % after rounding down to 2 decimal places
disp('h = ')
disp(h)

%% Abscissas
X = min(interval):h:max(interval);
Y = 1/(sqrt(2*pi))*exp(-(X.^2/2));
numPts = length(X);  % n+1 (no. of abscissas)
disp('x =')
disp(X)
disp('y = ')
disp(Y')

% computation for constant a
a = Y;
%% Solving for constant c
A = zeros(numPts, numPts);  % matrix A of eqn (12)
B = zeros(numPts, 1);  % matrix b of eqn (12)
for k = 2:numPts-1  % iterates over 2 to n
    % supplying middle rows of A
    A(k, k-1) = h;
    A(k, k) = 4*h;
```

```
    A(k, k+1) = h;

    % supplying middle rows of b
    B(k) = (3/h) * (a(k+1) - 2*a(k) + a(k-1));
end

% imposing free conditions (clamped)
fPrime = diff(f, 1);  % first derivative of f
fPrimea = double(subs(fPrime, -3));  %fPrime(-3)
fPrimeb = double(subs(fPrime, 3));  %fPrime(3)
A(1, 1) = 2*h;
A(1, 2) = h;
A(numPts, numPts-1) = h;
A(numPts, numPts) = 2*h;
B(1) = (3/h)*(a(2)-a(1)) - (3*fPrimea);
B(numPts) = (3*fPrimeb) - (3/h) * (a(numPts)-a(numPts-1));

% computes for c
c = A\B;
%% Solving for constants b and d
b = zeros(numPts-1, 1);
d = zeros(numPts-1, 1);
for k = 1:numPts-1
    % constant d
    d(k) = (c(k+1)-c(k))/(3*h);

    % constant b
    l = (a(k+1)-a(k))/h;
    r = (2*c(k)+c(k+1))/3 * h;
    b(k) = l-r;
end
disp('b =')
disp(b)
disp('d =')
disp(d)

%% Constructing piecewise cubic splines
for i = 1:numPts-1
    S(i) = a(i) + b(i)*(x-X(i)) + c(i)*(x-X(i))^2 + d(i)*(x-X(i))^3;
end

disp(vpa(expand(S'), 4))

%% Plotting S with f
s = piecewise( ...
    (X(1)<=x)&(x<=X(2)), S(1), ...
    (X(2)<=x)&(x<=X(3)), S(2), ...
    (X(3)<=x)&(x<=X(4)), S(3), ...
    (X(4)<=x)&(x<=X(5)), S(4), ...
    (X(5)<=x)&(x<=X(6)), S(5), ...
    (X(6)<=x)&(x<=X(7)), S(6));

figure
hold on
fplot(s, 'r')
```

```
fplot(f, interval, 'k')
% scatter(X, Y, 'filled', 'k')  % scatter plot of data points (optional)
grid
legend('clamped cubic spline $s(x)$', '$f(x)$', ...
    'Interpreter', 'Latex', 'Fontsize', 14)
xlabel('$x$', ...
    'Interpreter', 'Latex', 'Fontsize', 14)
ylabel('$y$', ...
    'Interpreter', 'Latex', 'Fontsize', 14)
xlim([-3.1, 3.1])
```