HW12

Q1 [CORRECT]

Given the data

x	1	2	2.5	3	4	5
f(x)	1	5	7	8	2	1

Fit these data with

(a) a cubic spline with natural end conditions

(b) a cubic spline with not-a-knot end conditions

Rounding a decimal number to four decimal places.

Compute the value of the splines when x = 4.5

A cubic spline with natural end conditions : $f_{predict}$ (4.5) =

A cubic spline with not-a-knot end conditions : $f_{predict}$ (4.5) =

```
x = [1 2 2.5 3 4 5];
y = [1 5 7 8 2 1];
xtest = 4.5;
natspl = natspline(x, y, xtest)
spl = spline(x, y, xtest)
```

Q2 [CORRECT]

Bessel functions often arise in advanced engineering and scientific analyses such as the study of electric fields.

These functions are usually not amenable to straightforward evaluation and, therefore, are often compiled in standard mathematical tables.

For example,

x	1.8	2	2.2	2.4	2.6
J(x)	0.5815	0.5767	0.556	0.5202	0.4708

Rounding a decimal number to four decimal places.

Estimate J(2.1) using

(a) an interpolating polynomial

J(2.1) =

(b) cubic splines (based on the default MATLAB spline function)

J(2.1) =

```
x =    [1.8 2 2.2 2.4 2.6];
y =    [0.5815 0.5767 0.556 0.5202 0.4708];
xtest = 2.1;
interp = interp1(x, y, xtest, "cubic")
spl = spline(x, y, xtest)
```

Q3 [CORRECT]

Fit these 4 points (3.0, 2.5), (4.5, 1.0), (7.0, 2.5), (9.0, 0.5) with quadratic splines

$s_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2$

Determine coefficients of each spline segments.

Rounding a decimal number to four decimal places.

For $s_1(x)$, $a_1 =$ 2.5000 , $b_1 =$ -1.0000 , $c_1 =$ 0.0000

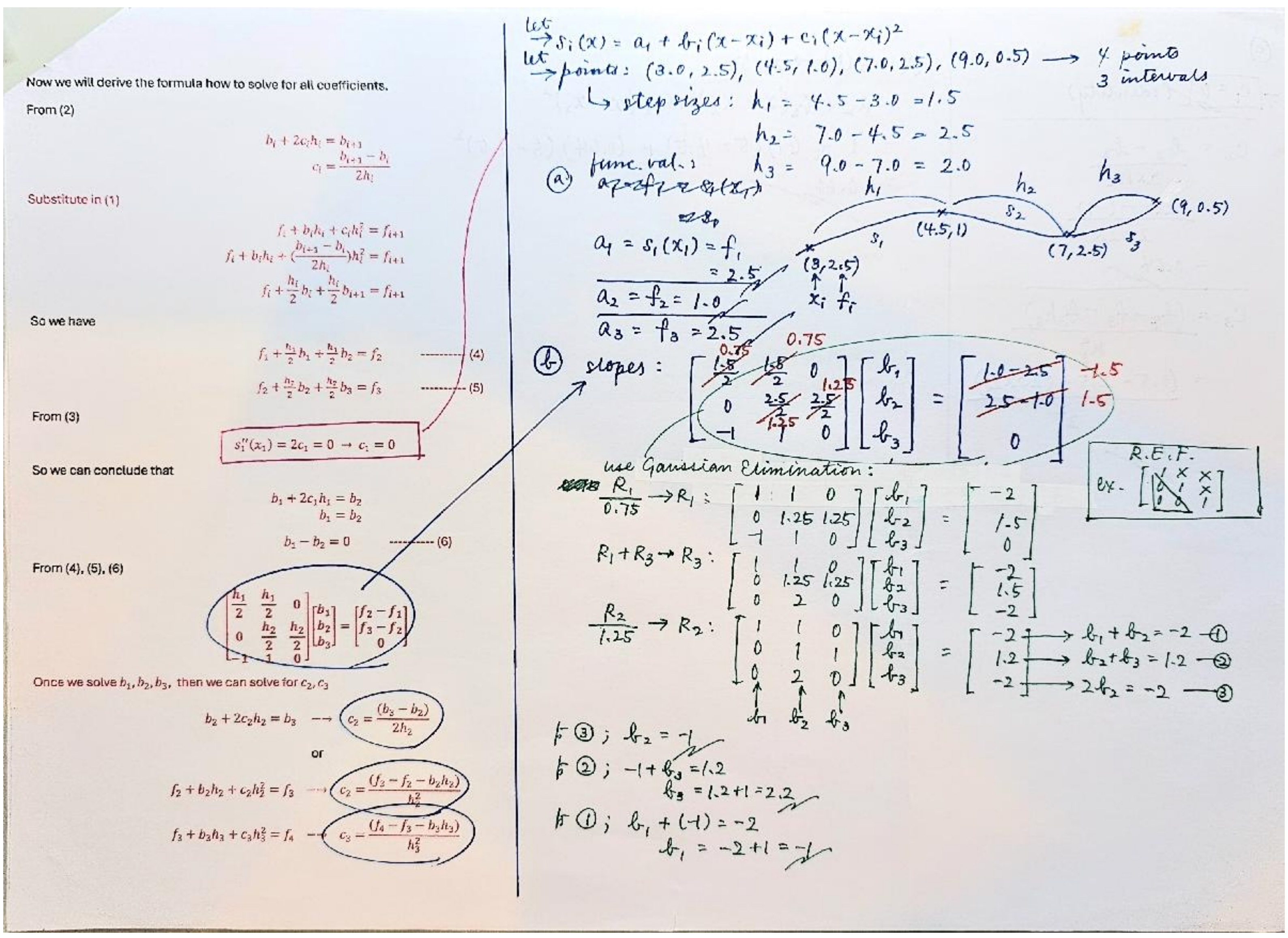
For $s_2(x)$, $a_2 =$ 1.0000 , $b_2 =$ -1.0000 , $c_2 =$ 0.6400

For $s_3(x)$, $a_3 =$ 2.5000 , $b_3 =$ 2.2000 , $c_3 =$ -1.6000

And also compute the value of the spline when x = 5

$s_2(5) =$ 0.6600

Hand Calculation



Now we will derive the formula how to solve for all coefficients.

From (2)

$$b_i + 2c_i h_i = b_{i+1}$$
$$c_i = \frac{b_{i+1} - b_i}{2h_i}$$

Substitute in (1)

$$f_i + b_i h_i + c_i h_i^2 = f_{i+1}$$
$$f_i + b_i h_i + \left(\frac{b_{i+1} - b_i}{2h_i}\right)h_i^2 = f_{i+1}$$
$$f_i + \frac{h_i}{2} b_i + \frac{h_i}{2} b_{i+1} = f_{i+1}$$

So we have

$$f_1 + \frac{h_1}{2} b_1 + \frac{h_1}{2} b_2 = f_2 \quad \text{------ (4)}$$
$$f_2 + \frac{h_2}{2} b_2 + \frac{h_2}{2} b_3 = f_3 \quad \text{------ (5)}$$

From (3)

$$\boxed{s_1''(x_1) = 2c_1 = 0 \rightarrow c_1 = 0}$$

So we can conclude that

$$b_1 + 2c_1 h_1 = b_2$$
$$b_1 = b_2$$
$$b_1 - b_2 = 0 \quad \text{------ (6)}$$

From (4), (5), (6)

$$\begin{bmatrix} \frac{h_1}{2} & \frac{h_1}{2} & 0 \\ 0 & \frac{h_2}{2} & \frac{h_2}{2} \\ -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} f_2 - f_1 \\ f_3 - f_2 \\ 0 \end{bmatrix}$$

Once we solve b_1, b_2, b_3, then we can solve for c_2, c_3

$$b_2 + 2c_2 h_2 = b_3 \dashrightarrow c_2 = \frac{(b_3 - b_2)}{2h_2}$$

or

$$f_2 + b_2 h_2 + c_2 h_2^2 = f_3 \dashrightarrow c_2 = \frac{(f_3 - f_2 - b_2 h_2)}{h_2^2}$$
$$f_3 + b_3 h_3 + c_3 h_3^2 = f_4 \dashrightarrow c_3 = \frac{(f_4 - f_3 - b_3 h_3)}{h_3^2}$$

let $\rightarrow s_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2$

let $\rightarrow$ points: (3.0, 2.5), (4.5, 1.0), (7.0, 2.5), (9.0, 0.5) $\longrightarrow$ 4 points, 3 intervals

↳ step sizes: $h_1 = 4.5 - 3.0 = 1.5$

$h_2 = 7.0 - 4.5 = 2.5$

$h_3 = 9.0 - 7.0 = 2.0$

(a) func. val.:

$a_1 = s_1(x_1) = f_1 = 2.5$

$a_2 = f_2 = 1.0$

$a_3 = f_3 = 2.5$

h_1, h_2, h_3, s_1, s_2, s_3, (3, 2.5), (4.5, 1), (7, 2.5), (9, 0.5), x_i f_i

(b) slopes:

$$\begin{bmatrix} \cancel{\frac{1.5}{2}}\ 0.75 & \cancel{\frac{1.5}{2}}\ 0.75 & 0 \\ 0 & \cancel{\frac{2.5}{2}}\ 1.25 & \cancel{\frac{2.5}{2}}\ 1.25 \\ -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} \cancel{1.0 - 2.5}\ -1.5 \\ \cancel{2.5 - 1.0}\ 1.5 \\ 0 \end{bmatrix}$$

use Gaussian Elimination:

R.E.F. ex. $\begin{bmatrix} 1 & x & x \\ 0 & 1 & x \\ 0 & 0 & 1 \end{bmatrix}$

$\frac{R_1}{0.75} \rightarrow R_1$: $\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1.25 & 1.25 \\ -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 1.5 \\ 0 \end{bmatrix}$

$R_1 + R_3 \rightarrow R_3$: $\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1.25 & 1.25 \\ 0 & 2 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 1.5 \\ -2 \end{bmatrix}$

$\frac{R_2}{1.25} \rightarrow R_2$: $\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 2 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 1.2 \\ -2 \end{bmatrix}$ $\begin{matrix} \rightarrow b_1 + b_2 = -2 \text{ — ①} \\ \rightarrow b_2 + b_3 = 1.2 \text{ — ②} \\ \rightarrow 2b_2 = -2 \text{ — ③} \end{matrix}$

b_1 b_2 b_3

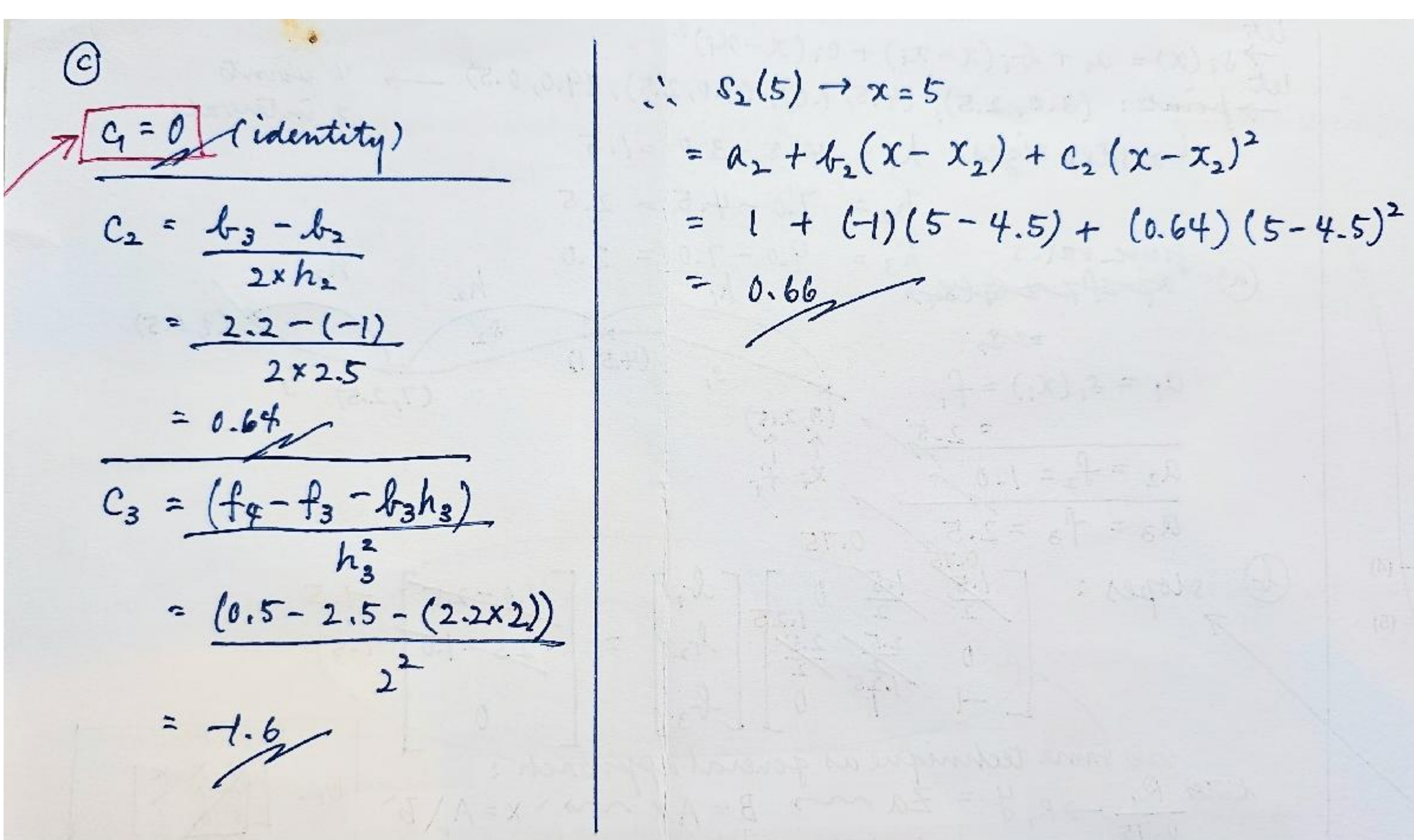
f ③; $b_2 = -1$

f ②; $-1 + b_3 = 1.2$

$b_3 = 1.2 + 1 = 2.2$

f ①; $b_1 + (-1) = -2$

$b_1 = -2 + 1 = -1$



(c)

$\boxed{c_1 = 0}$ (identity)

$$c_2 = \frac{b_3 - b_2}{2 \times h_2} = \frac{2.2 - (-1)}{2 \times 2.5} = 0.64$$

$$c_3 = \frac{(f_4 - f_3 - b_3 h_3)}{h_3^2} = \frac{(0.5 - 2.5 - (2.2 \times 2))}{2^2} = -1.6$$

$\therefore s_2(5) \rightarrow x = 5$

$= a_2 + b_2(x - x_2) + c_2(x - x_2)^2$

$= 1 + (-1)(5 - 4.5) + (0.64)(5 - 4.5)^2$

$= 0.66$

Q4 [CORRECT]

The following data define the sea-level concentration of dissolved oxygen for fresh water as a function of temperature:

T	0	8	16	24	32	40
O	14.621	11.843	9.870	8.418	7.305	6.413

Fit the data with

(a) piecewise linear interpolation

(b) a fifth-order polynomial

(c) cubic splines (the MATLAB spline function)

Estimate O(27).

Rounding a decimal number to four decimal places.

(a) piecewise linear interpolation O(27) =

(b) a fifth-order polynomial O(27) =

(c) cubic splines O(27) =

```
x    = [0     8     16     24     32     40];
y    = [14.621     11.843     9.870     8.418     7.305     6.413];
xtest = 27;
pplin = interp1(x, y, xtest, "linear")
pp = polyfit(x, y, 5);
p5 = polyval(pp, xtest) % NOT `ppval`
spl = spline(x, y, xtest)
```

Q5 [CORRECT]

Fit these 4 points (3.0, 2.5), (4.5, 1.0), (7.0, 2.5), (9.0, 0.5) with cubic splines with natural end conditions

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

Determine coefficients of each spline segments.

Rounding a decimal number to four decimal places.

For $s_1(x)$,

$a_1 =$ 2.5000, $b_1 =$ -1.4198,

$c_1 =$ 0, $d_1 =$ 0.1866

For $s_2(x)$,

$a_2 =$ 1, $b_2 =$ -0.1605,

$c_2 =$ 0.8395, $d_2 =$ 0.1866

For $s_3(x)$,

$a_3 =$ 2.5000, $b_3 =$ 0.0221,

$c_3 =$ -0.7665, $d_3 =$ 0.1278

And also compute the value of the spline when x=5

$s_2(5) =$ 1.1029

```
x = [3.0 4.5 7.0 9.0];
y = [2.5 1.0 2.5 0.5];
xx = linspace(3, 9, 4)
xtest = 5;
natspl = natspline(x, y, xx);
natspltest = natspline(x, y, xtest)
```

Q6 [CORRECT]

Given the data

x	0	100	200	400	600	800	1000
f(x)	0	0.8244	1.0000	0.7358	0.4060	0.1991	0.0916

Fit these data with

(a) a cubic spline with natural end conditions

(b) a cubic spline with not-a-knot end conditions

Rounding a decimal number to four decimal places.

Compute the value of the splines when x = 500

A cubic spline with natural end conditions : $f_{predict}(500)$ =

A cubic spline with not-a-knot end conditions : $f_{predict}(500)$ =

```
x = [0      100     200     400     600     800     1000];
y =     [0      0.8244      1.0000    0.7358     0.4060     0.1991     0.0916];
xtest = 500;
natspl = natspline(x, y, xtest)
spl = spline(x, y, xtest)
```

Q7 [CORRECT]

Fit a cubic spline (based on the default MATLAB spline function) to the following data to determine f(x) at x = 1.5:

x	0	2	4	7	10	12
f(x)	20	20	12	7	6	6

Rounding a decimal number to four decimal places.

f(1.5) =

```
x =     [0      2      4      7      10      12];
y = [20      20      12      7      6      6];
xtest = 1.5;
spl = spline(x, y, xtest)
```

```
spl =
21.3344
```