NM hw3 report

P1.

1.

(a)

$h = 0.12$, $s = \frac{0.231 - 0.12}{0.12} \fallingdotseq 0.925$

$f(x) \approx f(x_0) + s\Delta f(x_0) + \frac{s(s-1)}{2}\Delta^2 f(x_0)$

$\Rightarrow f(0.231) \approx 0.79168 + (0.925)(-0.01874) + \frac{(0.925)(0.925-1)}{2}(-0.01129)$

$\approx \underline{0.774182}$ #

(b)

$f(x) \approx f(x_0) + s\Delta f(x_0) + \frac{s(s-1)}{2!}\Delta^2 f(x_0) + \frac{s(s-1)(s-2)}{3!}\Delta^3 f(x_0)$

$\Rightarrow \frac{s(s-1)(s-2)}{3!}\Delta^3 f(x_0) = \frac{(0.925)(0.925-1)(0.925-2)}{6} \cdot 0.00134 \approx 0.0000167$

$\Rightarrow f(x) \approx 0.774182 + 0.0000167 = \underline{0.7742}$ #

(from (a))

(c)

① 誤差相 $\frac{s(s-1)(s-2)}{3!}\Delta^3 f(x_0) \Rightarrow \underline{0.0000167 \text{ (from (b))}}$ #

② 誤差相 $\frac{s(s-1)(s-2)(s-3)}{4!}\Delta^3 f(x_0)$

$\Rightarrow \frac{0.925(0.925-1)(0.925-2)(0.925-3)}{24} \cdot 0.00038 \approx \underline{-0.0000245}$ #

(d)

誤差相 $\frac{s(s-1)(s-2)}{3!}\Delta^3 f(x_0)$

$\Rightarrow \Delta^3 f(0.24) = 0.00172$

$\Delta^3 f(0.36) = 0.00200$

$\Rightarrow \left|\frac{s(s-1)(s-2)}{3!}\Delta^3 f(0.24)\right| < \left|\frac{s(s-1)(s-2)}{3!}\Delta^3 f(0.36)\right|$

$\Rightarrow$ 從 0.24 開始較好 #

p2.

a.

2.

① 5 points evenly in $[-1, 1]$ $\Rightarrow h = 0.5$, $x_0 = -1$

②

x	$f(x)$
-1	0
-0.5	0
0	1
0.5	0
1	0

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

$S_0(x): [-1, -0.5]$

$S_1(x): [-0.5, 0]$

$S_2(x): [0, 0.5]$

$S_3(x): [0.5, 1]$

③ 1. $S_i(x_i) = \underline{f(x_i) = a_i}$

2. $S_i(x_{i+1}) = S_{i+1}(x_{i+1}) = f(x_{i+1}) = a_{i+1}$

$\Rightarrow h = x_{i+1} - x_i = 0.5$

$\Rightarrow a_i + b_i h + c_i h^2 + d_i h^3 = a_{i+1} \Rightarrow \underline{a_{i+1} = \frac{1}{8} d_i + \frac{1}{4} c_i + \frac{1}{2} b_i + a_i}$

3. $S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$, $S'(x) = 3d_i(x-x_i)^2 + 2c_i(x-x_i) + b_i$

$\Rightarrow 3d_i h^2 + 2c_i h + b_i = b_{i+1} \Rightarrow \underline{b_{i+1} = \frac{3}{4} d_i + c_i + b_i}$

4. $S_i''(x_{i+1}) = S_{i+1}''(x_{i+1})$, $S''(x) = 6d_i(x-x_i) + 2c_i$

$\Rightarrow 6d_i h + 2c_i = 2c_{i+1} \Rightarrow \underline{c_{i+1} = \frac{3}{2} d_i + c_i}$

④ $a_0 = 0$, $a_1 = 0$, $a_2 = 1$, $a_3 = 0$, $b_0 = 0$, $c_0 = 0$, $d_0 = 0$

$$\begin{cases} a_1 = \frac{1}{8} d_0 + \frac{1}{4} c_0 + \frac{1}{2} b_0 + a_0 \\ b_1 = \frac{3}{4} d_0 + b_0 + c_0 \\ c_1 = \frac{3}{2} d_0 + c_0 \\ a_2 = \frac{1}{8} d_1 + \frac{1}{4} c_1 + \frac{1}{2} b_1 + a_1 \\ b_2 = \frac{3}{4} d_1 + b_1 + c_1 \\ c_2 = \frac{3}{2} d_1 + c_1 \\ a_3 = \frac{1}{8} d_2 + \frac{1}{4} c_2 + \frac{1}{2} b_2 + a_2 \\ b_3 = \frac{3}{4} d_2 + b_2 + c_2 \\ c_3 = \frac{3}{2} d_2 + c_2 \\ \frac{1}{8} d_3 + \frac{1}{4} c_3 + \frac{1}{2} b_3 + a_3 = 0 \\ \frac{3}{4} d_3 + c_3 + b_3 = 0 \end{cases}$$

$\Rightarrow$

$\begin{cases} a_0 = 0 \\ b_0 = 0 \\ c_0 = -3 \\ d_0 = 6 \end{cases}$: $S_0(x) = 6(x+1)^3 - 3(x+1)^2$

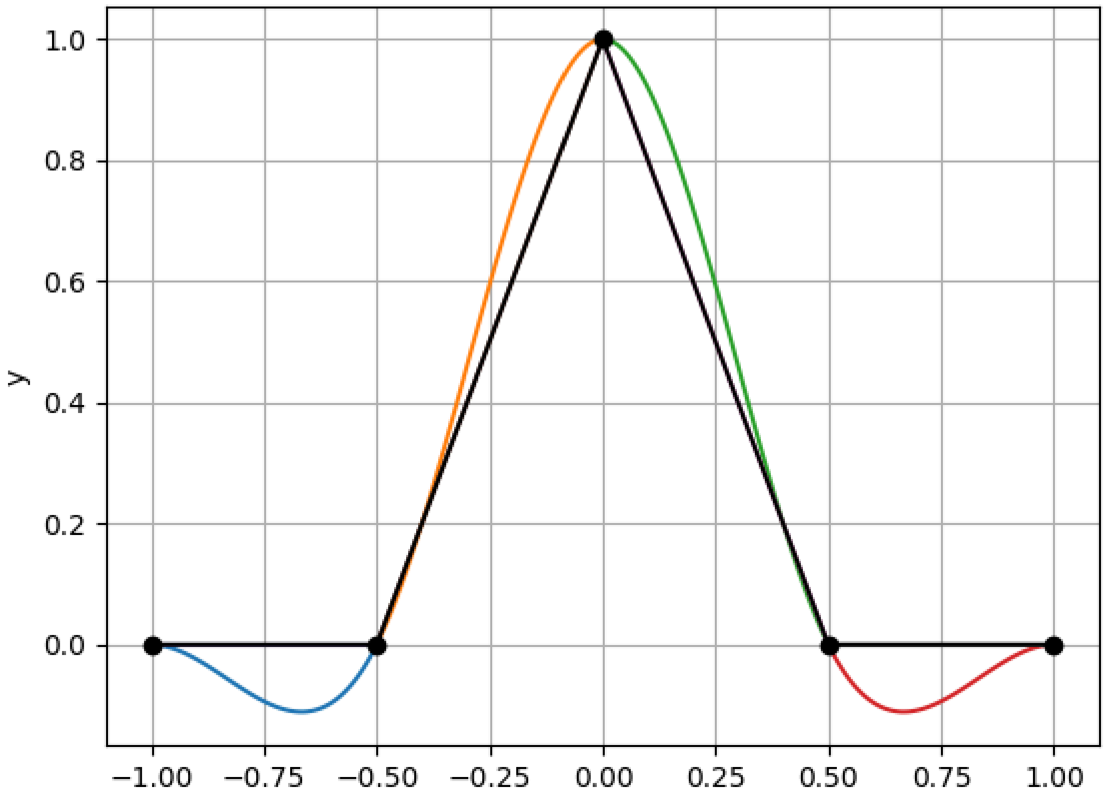
$\begin{cases} a_1 = 0 \\ b_1 = \frac{3}{2} \\ c_1 = 6 \\ d_1 = -10 \end{cases}$: $S_1(x) = -10(x+0.5)^3 + 6(x+0.5)^2 + \frac{3}{2}(x+0.5)$

$\begin{cases} a_2 = 1 \\ b_2 = 0 \\ c_2 = -9 \\ d_2 = 10 \end{cases}$: $S_2(x) = 10x^3 - 9x^2 + 1$

$\begin{cases} a_3 = 0 \\ b_3 = -\frac{3}{2} \\ c_3 = 6 \\ d_3 = -6 \end{cases}$: $S_3(x) = -6(x-0.5)^3 + 6(x-0.5)^2 - \frac{3}{2}(x-0.5)$

#

b.

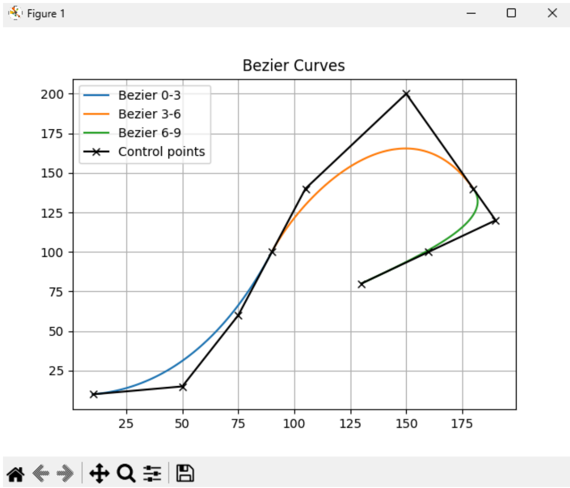


p3.

a.

先把10個控制點存在control_points，分成三段，每段套上bezier_curve公式，設定參數u 在[0,1] 上均勻取點，並分別將每段plot出來把 10 個控制點(x,y)分成三段，每段包含 4 個連續的控制點。針對每一段，套上 cubic Bezier 曲線的公式，並將每一小段分別畫出來。

```python
import numpy as np
import matplotlib.pyplot as plt

# 控制點
control_points = np.array([
    [10, 10],
    [50, 15],
    [75, 60],
    [90, 100],
    [105, 140],
    [150, 200],
    [180, 140],
    [190, 120],
    [160, 100],
    [130, 80]
])
x = control_points[:, 0]
y = control_points[:, 1]

# u from 0 to 1
u = np.linspace(0, 1, 100)

# 定義三段 Bezier 曲線公式
def bezier_curve(P, u):
    return (
        (1 - u) ** 3 * P[0] +
        3 * u * (1 - u) ** 2 * P[1] +
        3 * u**2 * (1 - u) * P[2] +
        u**3 * P[3]
    )

# 第一段 (0,1,2,3)
b1 = bezier_curve(x[0:4], u)
b1_y = bezier_curve(y[0:4], u)

# 第二段 (3,4,5,6)
b2 = bezier_curve(x[3:7], u)
b2_y = bezier_curve(y[3:7], u)

# 第三段 (6,7,8,9)
b3 = bezier_curve(x[6:10], u)
b3_y = bezier_curve(y[6:10], u)

# 畫出來
plt.plot(b1, b1_y, label='Bezier 0-3')
plt.plot(b2, b2_y, label='Bezier 3-6')
plt.plot(b3, b3_y, label='Bezier 6-9')

# 畫控制點
plt.plot(x, y, 'x-', label='Control points', color='black')

plt.legend()
plt.title('Connected Bezier Curves')
plt.grid(True)
plt.show()
```



b.

因為point 234 / 567都是colinear所以在point 3 / 6都會smoothly connect

c.

3.

(c)

$P_0 \sim P_3$: $b_0(u) = (1-u)^3 P_0 + 3u(1-u)^2 P_1 + 3u^2(1-u) P_2 + u^3 P_3$, for $u \in [0,1]$

$P_3 \sim P_6$: let $t = u-1$

$b_0(t) = (1-t)^3 P_3 + 3t(1-t)^2 P_4 + 3t^2(1-t) P_5 + t^3 P_6$, for $t \in [0,1]$

$= (2-u)^3 P_3 + 3(u-1)(2-u)^2 P_4 + 3(u-1)^2(2-u) P_5 + (u-1)^3 P_6$, for $u \in [1,2]$

$P_6 \sim P_9$: let $t = u-2$

$b_0(t) = (1-t)^3 P_6 + 3t(1-t)^2 P_7 + 3t^2(1-t) P_8 + t^3 P_9$, for $t \in [0,1]$

$= (3-u)^3 P_6 + 3(u-2)(3-u)^2 P_7 + 3(u-2)^2(3-u) P_8 + (u-2)^3 P_9$, for $u \in [1,2]$

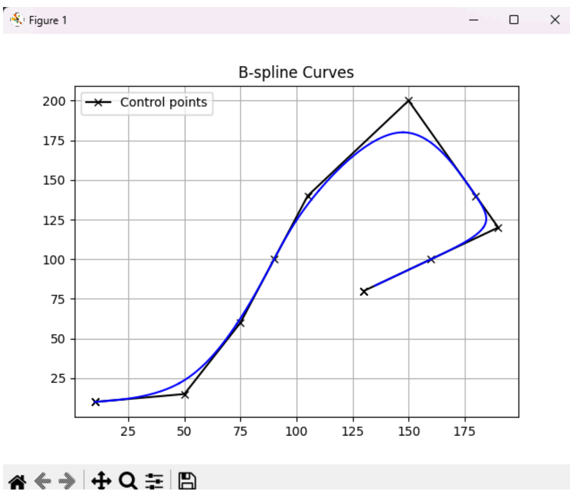
#

P4

a.

使用 uniform cubic B-spline 的四個基底函數，新增 P(-2) = P(-1) = P(0) 以及 P(n+2) = P (n+1) = P(1) ，每次取 4 個連續的控制點 (i-2, i-1, i , i+1)來組成一小段曲線，設定參數u 在[0,1] 上均勻取點，並分別plot出來。

```python
# u 在 [0,1] 之間
u = np.linspace(0, 1, 100)

# 定義 uniform cubic B-spline basis functions
def B_spline_basis(u):
    b0 = (1 - u)**3 / 6
    b1 = (3*u**3 - 6*u**2 + 4) / 6
    b2 = (-3*u**3 + 3*u**2 + 3*u + 1) / 6
    b3 = u**3 / 6
    return b0, b1, b2, b3

# 畫每一小段
for i in range(2,9):
    b0, b1, b2, b3 = B_spline_basis(u)
    g_x = b0 * x[i-2] + b1 * x[i-1] + b2 * x[i] + b3 * x[i+1]
    g_y = b0 * y[i-2] + b1 * y[i-1] + b2 * y[i] + b3 * y[i+1]
    plt.plot(g_x, g_y)
```



b.

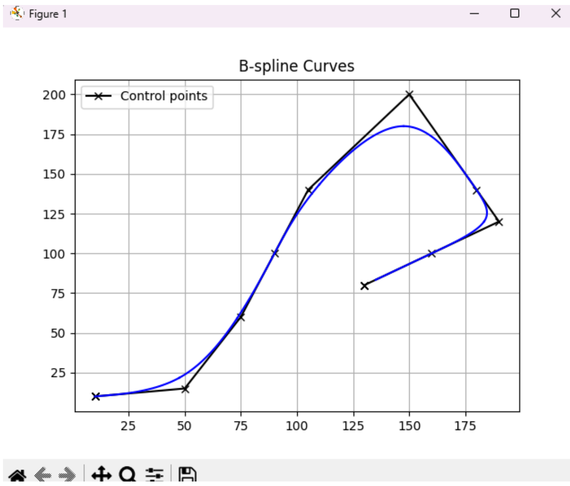
b-spline本來就是C2 continuous，所以會smoothly connected

c.

給予p0-p3, p3-p6, p6-p9分別不同範圍的u，再丟入B spline basis矩陣運算時透過不同的位移值將其位移回[0,1]的範圍內，得出曲線與(a)相同

```python
# u 在 [0,1] 之間
u = np.linspace(0, 1, 100)
u2 = np.linspace(1, 2, 100)
u3 = np.linspace(2, 3, 100)
```

```python
for i in range(2, 12):
    if i <= 3:
        b0, b1, b2, b3 = B_spline_basis(u)
        g_x = b0 * x[i-2] + b1 * x[i-1] + b2 * x[i] + b3 * x[i+1]
        g_y = b0 * y[i-2] + b1 * y[i-1] + b2 * y[i] + b3 * y[i+1]
    elif i <= 6:
        b0, b1, b2, b3 = B_spline_basis(u2 - 1)
        g_x = b0 * x[i-2] + b1 * x[i-1] + b2 * x[i] + b3 * x[i+1]
        g_y = b0 * y[i-2] + b1 * y[i-1] + b2 * y[i] + b3 * y[i+1]
    else:
        b0, b1, b2, b3 = B_spline_basis(u3 - 2)
        g_x = b0 * x[i-2] + b1 * x[i-1] + b2 * x[i] + b3 * x[i+1]
        g_y = b0 * y[i-2] + b1 * y[i-1] + b2 * y[i] + b3 * y[i+1]

    plt.plot(g_x, g_y, color='blue')
```



P5

a.

5、

$$z = ax + by + c \ , \quad A = [\, x_i \quad y_i \quad z_i \,]$$

$\Rightarrow$ use least square to solve

$$A^T A \begin{bmatrix} a \\ b \\ c \end{bmatrix} = A^T z$$

#

b.

```python
# 題目給的資料
x = np.array([0.40, 1.2, 3.4, 4.1, 5.7, 7.2, 9.3])
y = np.array([0.70, 2.1, 4.0, 4.9, 6.3, 8.1, 8.9])
z = np.array([0.031, 0.933, 3.058, 3.349, 4.870, 5.757, 8.921])

# (A^T A) a = A^T b
A = np.vstack([x, y, np.ones_like(x)]).T
b = z

# A^T A = A^T b
ATA = A.T @ A
ATb = A.T @ b

# 解x
x = np.linalg.solve(ATA, ATb)
a, b_coeff, c = x

print(f"res: z = {a:.5f} x + {b_coeff:.5f} y + {c:.5f}")
```

使用python的矩陣運算解出z為

```
res: z = 1.59609 x + -0.70238 y + 0.22067
```

c.

```python
# residuals
z_pred = A @ x
residuals = z - z_pred
SSE = np.sum(residuals**2)

print(f"sum of the squares of the deviations:  {SSE:.5f}")
```

計算每一個點和預測值的差距並平方相加

```
sum of the squares of the deviations:  0.31940
```

P6

6、

$$\cos^2(x) = 1 - x^2 + \frac{1}{3}x^4 - \frac{2}{45}x^6$$

$$= \frac{a_0 + a_1 x + a_2 x^2 + a_3 x^3}{1 + b_1 x + b_2 x^2 + b_3 x^3}$$

$$\Rightarrow \begin{array}{ll} a_0 = 1 & \frac{1}{3} - b_2 = 0 \\ a_1 = b_1 & \\ a_2 = -1 + b_2 & \frac{1}{3}b_1 - b_3 = 0 \\ a_3 = -b_1 + b_3 & \frac{-2}{45} + \frac{1}{3}b_2 = 0 \end{array} \Rightarrow \begin{array}{ll} a_0 = 1 & b_1 = 0 \\ a_1 = 0 & b_2 = \frac{1}{3} \\ a_2 = \frac{-2}{3} & b_3 = 0 \\ a_3 = 0 & \end{array}$$

$$\sin(x^4 - x) = -x + \frac{1}{6}x^3 + x^4 - \frac{1}{120}x^5$$

$$= \frac{a_0 + a_1 x + a_2 x^2 + a_3 x^3}{1 + b_1 x + b_2 x^2 + b_3 x^3}$$

$$\Rightarrow \begin{array}{ll} a_0 = 0 & \\ a_1 = -1 & 0 = -b_3 + \frac{1}{6}b_1 + 1 \\ a_2 = -b_1 & 0 = \frac{1}{6}b_2 + b_1 - \frac{1}{120} \\ a_3 = -b_2 + \frac{1}{6} & 0 = \frac{1}{6}b_3 + b_2 + \frac{1}{120}b_1 \end{array} \Rightarrow \begin{array}{ll} a_0 = 0 & b_1 = \frac{78}{2147} \\ a_1 = -1 & \\ a_2 = \frac{-78}{2147} & b_2 = \frac{-7213}{42940} \\ a_3 = \frac{43109}{128820} & b_3 = \frac{2160}{2147} \end{array}$$

$$\Rightarrow xe^x = x + x^2 + \frac{1}{2}x^3 + \frac{1}{6}x^4 + \frac{1}{24}x^5 + \frac{1}{120}x^6$$

$$x + x^2 + \frac{1}{2}x^3 + \frac{1}{6}x^4 + \frac{1}{24}x^5 + \frac{1}{120}x^6 = \frac{a_0 + a_1 x + a_2 x^2 + a_3 x^3}{1 + b_1 x + b_2 x^2 + b_3 x^3}$$

$$\Rightarrow \begin{array}{ll} a_0 = 0 & 0 = \frac{1}{6} + \frac{1}{2}b_1 + b_2 + b_3 \\ a_1 = 1 & 0 = \frac{1}{24} + \frac{1}{6}b_1 + \frac{1}{2}b_2 + b_3 \\ a_2 = 1 + b_1 & 0 = \frac{1}{120} + \frac{1}{24}b_1 + \frac{1}{6}b_2 + \frac{1}{2}b_3 \\ a_3 = \frac{1}{2} + b_1 + b_3 & \end{array}$$

$$\Rightarrow xe^x = \frac{x + \frac{3}{5}x^2 + \frac{1}{20}x^3}{1 - \frac{2}{5}x + \frac{3}{20}x^2 - \frac{1}{60}x^3}$$

\#

P7

7.

(a) $f(x) = xe^{-x},\ f'(x) = e^{-x} - xe^{-x} = (1-x)e^{-x}$

$\Rightarrow f(1) = \frac{1}{e},\ f'(1) = 0$

$f(2) = \frac{2}{e^2},\ f'(2) = \frac{-1}{e^2}$

$f(3) = \frac{3}{e^3},\ f'(3) = \frac{-2}{e^3}$

$$\Rightarrow P_1(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{e} \\ \frac{2}{e^2} \\ 0 \\ \frac{-1}{e^2} \end{bmatrix} = \left(\frac{2}{e} - \frac{5}{e^2}\right)u^3 + \left(\frac{-3}{e} + \frac{7}{e^2}\right)u^2 + \frac{1}{e}$$

$$\Rightarrow P_2(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{2}{e^2} \\ \frac{3}{e^3} \\ \frac{-1}{e^2} \\ \frac{-2}{e^3} \end{bmatrix} = \left(\frac{3}{e^2} - \frac{8}{e^3}\right)u^3 + \left(\frac{-4}{e^2} + \frac{11}{e^3}\right)u^2 + \frac{-u+2}{e^2}$$

(b) $f\left(\frac{3}{2}\right) = P_1\left(\frac{1}{2}\right)$

$$= \left(\frac{2}{e} - \frac{5}{e^2}\right)\frac{1}{8} + \left(\frac{-3}{e} + \frac{7}{e^2}\right)\frac{1}{4} + \frac{1}{e} \fallingdotseq 0.3362$$