Assignment 3

學號: 111550129　姓名: 林彥亨

---

1. (20pts)

(a) Construct the divided-difference table from these data:

x	-0.2	0.3	0.7	-0.3	0.1
$f(x)$	1.23	2.34	-1.05	6.51	-0.06

(b) Use the divided-difference table to interpolate for $f(0.4)$ with the first three points.

(c) Repeat (b) but use the best set of three points. Which points should be used?

(a)

x	$f(x)$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$	$f[x_i, x_{i+1} \dots x_{i+4}]$
-0.2	1.23	2.22	-11.8833	-103.5833	93.6111
0.3	2.34	-8.475	-1.525	-81.5	
0.7	-1.05	-7.56	14.775		
-0.3	6.51	-16.425			
0.1	-0.06				

#

$$f[x_0, x_1] = \frac{2.34-1.23}{0.3+0.2} = \frac{1.11}{0.5} = 2.22$$

$$f[x_1, x_2] = \frac{-1.05-2.34}{0.7-0.3} = \frac{-3.39}{0.4} = -8.475$$

$$f[x_2, x_3] = \frac{7.56}{-1} = -7.56$$

$$f[x_3, x_4] = \frac{-0.06-6.51}{0.4} = \frac{-6.57}{0.4} = -16.425$$

$$f[x_0, x_1, x_2] = \frac{-8.475-2.22}{0.9} = -11.8833$$

$$f[x_1, x_2, x_3] = \frac{-7.56+8.475}{-0.6} = -1.525$$

$$f[x_2, x_3, x_4] = \frac{-8.865}{-0.6} = 14.775$$

$$f[x_0, x_1, x_2, x_3] = \frac{-1.525+11.8833}{-0.1} = -103.5833$$

$$f[x_1, x_2, x_3, x_4] = \frac{14.775+1.525}{-0.2} = -81.5$$

$$f[x_0, x_1 \dots x_4] = \frac{-81.5+103.5833}{0.3} = 93.6111$$

(b)

$$f(0.4) = (1.23) + (2.22)(0.4+0.2) + (-11.8833)\underbrace{(0.4+0.2)(0.4-0.3)}_{0.06}$$

$$= 1.23 + 1.332 + (-0.712998)$$

$$= 1.849 \ \#$$

(c)

x	$f(x)$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$
0.1	-0.06	12	-34.125
0.3	2.34	-8.475	
0.7	-1.05		

0.1, 0.3, 0.7 #

$$P_2(x) = f[x_2] + f[x_2, x_3](x-x_2) + f[x_2, x_3, x_4](x-x_2)(x-x_3)$$

$$= (-0.06) + 12(0.4-0.1) + (-34.125)\underbrace{(0.4-0.1)(0.4-0.3)}_{0.3 \times 0.1}$$

$$= -0.06 + 3.6 + -1.02375$$

$$= 2.51625 \ \#$$

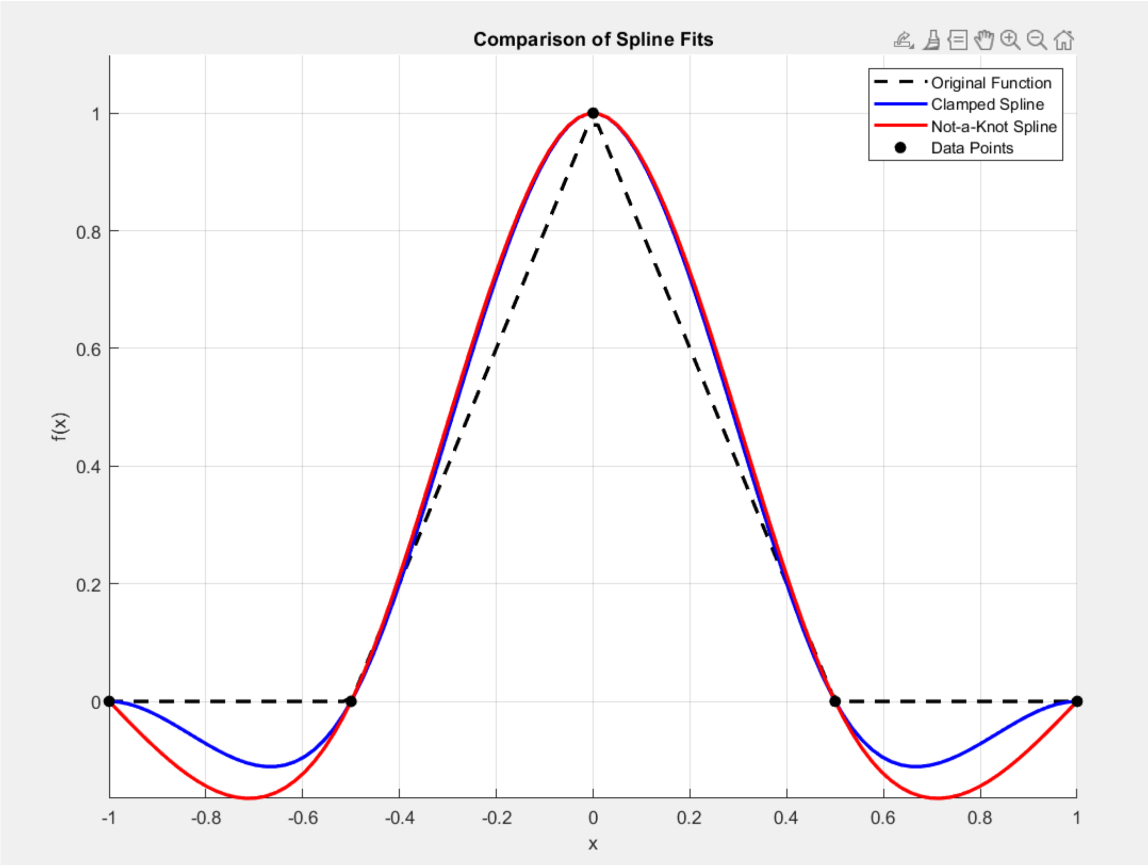
2. (20pts) Fit the function below with a natural cubic spline that matches to $f(x)$ at five evenly spaced points in [-1, 1]. Use end conditions 3 and 4 to plot the spline curve together with $f(x)$. Which end condition gives the best fit to the function?

$$f(x) = \begin{cases} 0, & -1 < x < -0.5 \\ 1 - |2x|, & -0.5 < x < 0.5 \\ 0, & 0.5 < x < 1 \end{cases}$$

- Here is the code to construct the comparison graph.

```
x = [-1, -0.5, 0, 0.5, 1];
y = [0, 0, 1, 0, 0];

% Define a fine grid for evaluation and plotting
xx = linspace(-1, 1, 100);

% Function definition for f(x)
f = @(x) (x>=-0.5 & x<=0.5).*(1 - 2*abs(x));

% Clamped Spline with zero derivatives at endpoints
pp_clamped = csape(x, [0, y, 0], 'clamped');

% Not-a-Knot Spline
pp_notaknot = csape(x, y, 'variational'); % 'variational' is equivalent to not-a-knot

% Evaluate the splines
yy_clamped = ppval(pp_clamped, xx);
yy_notaknot = ppval(pp_notaknot, xx);

% Plot the original function and the splines
figure;
hold on;
plot(xx, f(xx), 'k--', 'LineWidth', 2);
plot(xx, yy_clamped, 'b-', 'LineWidth', 2);
plot(xx, yy_notaknot, 'r-', 'LineWidth', 2);
plot(x, y, 'ko', 'MarkerFaceColor', 'k');
xlim([-1, 1]);
ylim([-0.1, 1.1]);
title('Comparison of Spline Fits');
xlabel('x');
ylabel('f(x)');
grid on;
legend('Original Function', 'Clamped Spline', 'Not-a-Knot Spline', 'Data Points');
```

- here is the result figure.



Through the figure, we can have the answer that the end condition 3 gives the better fit to the function.

---

3. (20pts) If these four points are connected in order by straight lines, a zigzag line is created:

$$(0, 0), (1, 0.3), (2, 1.7), (3, 1.5).$$

   (a) Using the two interior points as controls, find the cubic Bezier curve. Plot this together with the zigzag line

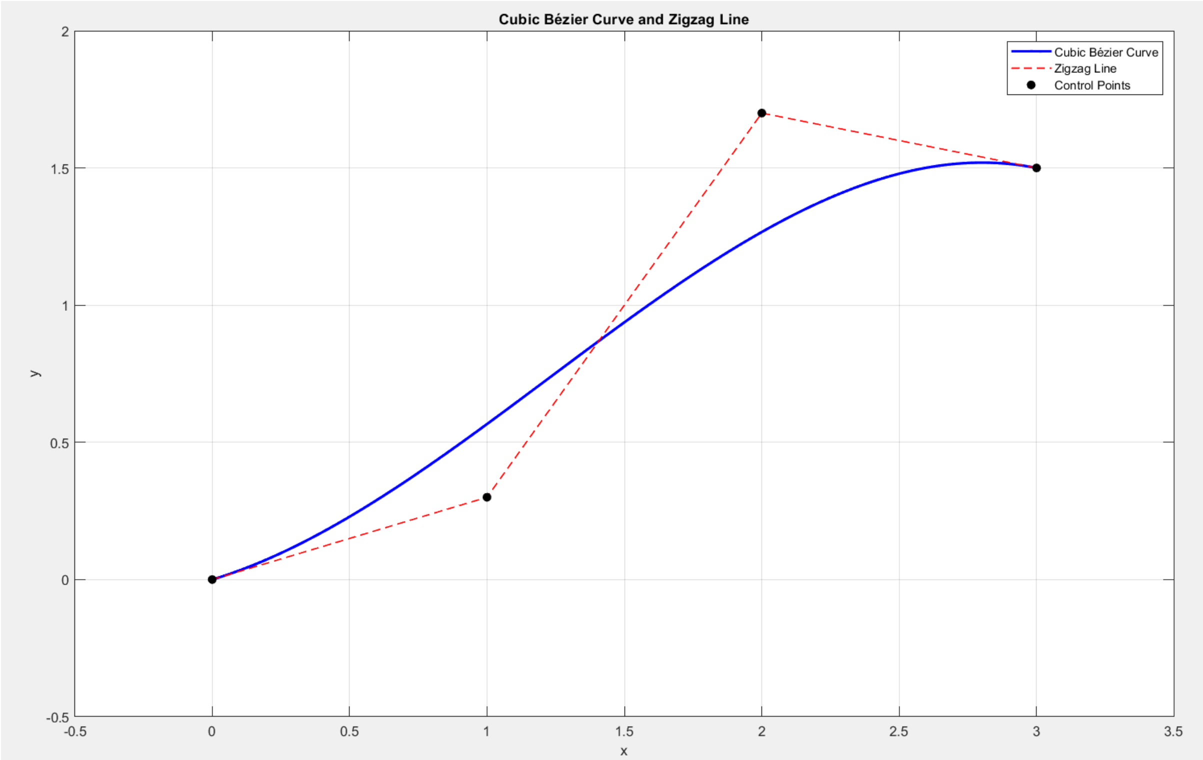
   (b) If the second and third points (the control points) are moved, the Bezier curve will change. If these are moved vertically, where should they be located so that the Bezier curve passes through all of the original four points?

Part(a)

- Here is the code.

```matlab
% Define the points
P0 = [0, 0];
P1 = [1, 0.3];
P2 = [2, 1.7];
P3 = [3, 1.5];

% Generate t values
t = linspace(0, 1, 100);

% Calculate the Bézier curve
Bx = (1-t).^3 * P0(1) + 3*(1-t).^2 .* t * P1(1) + 3*(1-t).*t.^2 * P2(1) + t.^3 * P3(1);
By = (1-t).^3 * P0(2) + 3*(1-t).^2 .* t * P1(2) + 3*(1-t).*t.^2 * P2(2) + t.^3 * P3(2);

% Plot the Bézier curve
figure;
plot(Bx, By, 'b-', 'LineWidth', 2);
hold on;

% Plot the zigzag line
zigzagX = [P0(1), P1(1), P2(1), P3(1)];
zigzagY = [P0(2), P1(2), P2(2), P3(2)];
plot(zigzagX, zigzagY, 'r--', 'LineWidth', 1);

% Add points
plot(zigzagX, zigzagY, 'ko', 'MarkerFaceColor', 'k');

% Labels and legend
title('Cubic Bézier Curve and Zigzag Line');
xlabel('x');
ylabel('y');
legend('Cubic Bézier Curve', 'Zigzag Line', 'Control Points');
grid on;

% Adjust the axis limits for better viewing
axis equal;
xlim([-0.5, 3.5]);
ylim([-0.5, 2]);
```

- Here is the figure for the part(a) that the cubic Bezier Curve and the zigzag line.



Part(b)

3. (20pts) If these four points are connected in order by straight lines, a zigzag line is created:

$$(0,0),(1,0.3),(2,1.7),(3,1.5).$$

(a) Using the two interior points as controls, find the cubic Bezier curve. Plot this together with the zigzag line

(b) If the second and third points (the control points) are moved, the Bezier curve will change. If these are moved vertically, where should they be located so that the Bezier curve passes through all of the original four points?

1

(b)

Let $(1, 0.3) \rightarrow (1, a)$

$(2, 1.7) \rightarrow (2, b)$

$$D(u) = \begin{bmatrix} (1-u)^3 \\ 3u(1-u)^2 \\ 3u^2(1-u) \\ u^3 \end{bmatrix} \begin{bmatrix} D_0 \\ D_1 \\ D_2 \\ D_3 \end{bmatrix}$$

$\rightarrow Y_0(1) = Y_1(0) = 0.3$

$Y_1(1) = Y_2(0) = 1.7$

4. (20pts)The function whose values are tabulated below is $z = x + e^y$. Construct the B-spline surface from the rectangular array of 16 points nearest to (2.8, 0.54) and find z(2.8, 0.54).

$x \backslash y$	0.2	0.4	0.5	0.7	0.9
1.3	2.521	2.792	2.949	3.314	3.760
2.5	3.721	3.992	4.149	4.514	4.960
3.1	4.321	4.592	4.749	5.114	5.560
4.7	5.921	6.192	6.349	6.714	7.160
5.5	6.721	6.992	7.149	7.514	7.960

First, we try to decide the 16 points nearest to (2.8, 0.54)

- here is the rectangualr array i chosed.

4. (20pts)The function whose values are tabulated below is $z = x + e^y$. Construct the B-spline surface from the rectangular array of 16 points nearest to (2.8, 0.54) and find z(2.8, 0.54).

$x\backslash y$	0.2	0.4	0.5	0.7	0.9
1.3	2.521	2.792	2.949	3.314	3.760
2.5	3.721	3.992	4.149	4.514	4.960
3.1	4.321	4.592	4.749	5.114	5.560
4.7	5.921	6.192	6.349	6.714	7.160
5.5	6.721	6.992	7.149	7.514	7.960

$$\begin{bmatrix} (1.3,0.2,2.521) & (1.3,0.4,2.792) & (1.3,0.5,2.949) & (1.3,0.7,3.314) \\ (2.5,0.2,3.721) & (2.5,0.4,3.992) & (2.5,0.5,4.149) & (2.5,0.7,4.514) \\ (3.1,0.2,4.321) & (3.1,0.2,4.592) & (3.1,0.5,4.749) & (3.1,0.7,5.114) \\ (4.7,0.2,5.921) & (4.7,0.4,6.192) & (4.7,0.5,6.349) & (4.7,0.7,6.714) \end{bmatrix}$$

$X_{ij}(u,v) = u^T M X_{ij} M^T v$ ，同理 $Y_{ij}(u,v)$，$Z_{ij}(u,v)$

$$M = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad u = \begin{bmatrix} u^3 \\ u^2 \\ u \\ 1 \end{bmatrix}, \quad v = \begin{bmatrix} v^3 \\ v^2 \\ v \\ 1 \end{bmatrix}$$

$X_{ij}(u,v) = 2.8$

$$= u^T \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1.3 & 1.3 & 1.3 & 1.3 \\ 2.5 & 2.5 & 2.5 & 2.5 \\ 3.1 & 3.1 & 3.1 & 3.1 \\ 4.7 & 4.7 & 4.7 & 4.7 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} u$$

$$= u^T \begin{bmatrix} 0 & 0 & 0 & 1.6 \\ 0 & 0 & 0 & -1.8 \\ 0 & 0 & 0 & 3.6 \\ 0 & 0 & 0 & 1.3 \end{bmatrix} v = 1.6u^3 - 1.8u^2 + 3.6u + 1.3$$

$\Rightarrow u = 0.4832$

$$Y_{ij}(u,v) = 0.54$$

$$= u^T \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.2 & 0.4 & 0.5 & 0.7 \\ 0.2 & 0.4 & 0.5 & 0.7 \\ 0.2 & 0.4 & 0.5 & 0.7 \\ 0.2 & 0.4 & 0.5 & 0.7 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} u$$

$$= u^T \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.2 & -0.3 & 0.6 & 0.2 \end{bmatrix} V = 0.2V^3 - 0.3V^2 + 0.6V + 0.2$$

$$\Rightarrow V = 0.6966$$

$$Z_{ij}(u,v) \Rightarrow Z_{ij}(0.4832, 0.6966)$$

$$= u^T \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 2.521 & 2.792 & 2.949 & 3.314 \\ 3.721 & 3.992 & 4.149 & 4.514 \\ 4.321 & 4.592 & 4.749 & 5.114 \\ 5.921 & 6.192 & 6.349 & 6.714 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} u$$

$$= u^T \begin{bmatrix} 0 & 0 & 0 & 1.6 \\ 0 & 0 & 0 & -1.8 \\ 0 & 0 & 0 & 3.6 \\ 0.3220 & -0.342 & 0.813 & 1.3 \end{bmatrix} \qquad V = 4.5302 \#$$

5. (20pts) The equation of a plane is z = ax + by + c. We can fit experimental data to such a plane using the least-squares technique. Here are some data for $z = f(x, y)$

x	0.40	1.2	3.4	4.1	5.7	7.2	9.3
y	0.70	2.1	4.0	4.9	6.3	8.1	8.9
z	0.031	0.933	3.058	3.349	4.870	5.757	8.921

(a) Develop the normal equations to fit the (x, y) data to a plane.

(b) Use these equations to fit $z = ax + by + c$.

(c) What is the sum of the squares of the deviations of the points from the plane?

Part(a)

(a)

$$\begin{bmatrix} z_0 \\ z_1 \\ \vdots \\ z_5 \\ z_6 \end{bmatrix} = \begin{bmatrix} x_0 & y_0 & 1 \\ x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots \\ x_6 & y_6 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

Part(b)

- here is the code.

```matlab
% Given data points
x = [0.40, 1.2, 3.4, 4.1, 5.7, 7.2, 9.3];
y = [0.70, 2.1, 4.0, 4.9, 6.3, 8.1, 8.9];
z = [0.031, 0.933, 3.058, 3.349, 4.870, 5.757, 8.921];

% Building the matrix A for the normal equations
A = [sum(x.^2), sum(x.*y), sum(x);
     sum(x.*y), sum(y.^2), sum(y);
     sum(x), sum(y), length(x)];

% Building the vector B for the normal equations
B = [sum(x.*z);
     sum(y.*z);
     sum(z)];

% Solving for the coefficients [a; b; c]
coefficients = A \ B;

% Display the coefficients
a = coefficients(1);
b = coefficients(2);
c = coefficients(3);
fprintf('Plane equation is z = %f*x + %f*y + %f\n', a, b, c);
```

- here is the result for the function z = ax + by + c

```
>> Question5_a
Plane equation is z = 1.596092*x + -0.702381*y + 0.220666
```

Part(c)

- here is the code.

```matlab
% Given data points
x = [0.40, 1.2, 3.4, 4.1, 5.7, 7.2, 9.3];
y = [0.70, 2.1, 4.0, 4.9, 6.3, 8.1, 8.9];
z = [0.031, 0.933, 3.058, 3.349, 4.870, 5.757, 8.921];

% Plane coefficients (example values, replace with your actual results)
a = 1.596092;  % Example value
b = -0.702381;  % Example value
c = 0.220666; % Example value

% Calculate predicted z-values using the plane equation
z_pred = a * x + b * y + c;

% Calculate the deviations
deviations = z - z_pred;

% Calculate the sum of the squares of the deviations
sum_of_squares = sum(deviations.^2);

% Display the sum of squares
fprintf('The sum of the squares of the deviations is: %f\n', sum_of_squares);
```

- here is the result.

```
>> Question5_c
The sum of the squares of the deviations is: 0.319395
```

---

6. (20pts) Find the first few terms of the Chebyshev series for $cos(x)$ by rewriting the Maclaurin series in terms of the $T(x)$'s and collecting terms. Convert this to a power series in x. Compare the error of both the Chebyshev series and the power series after truncating each to the fourth degree.

- here is the code

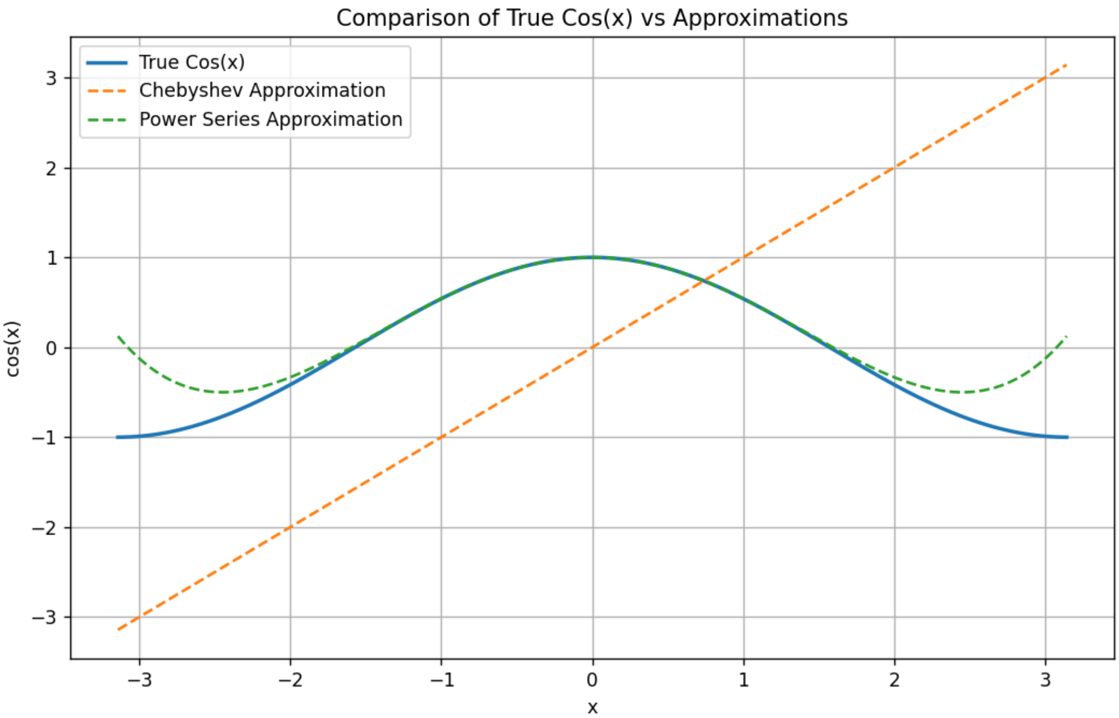
```python
import numpy as np
import matplotlib.pyplot as plt

# Define the functions
def chebyshev_series(x):
    return x  # T1(x) since cos(x) = T1(x) for x in the domain of cosine function

def power_series(x):
    return 1 - (x**2)/2 + (x**4)/24  # Up to x^4

# Values to evaluate
x_values = np.linspace(-np.pi, np.pi, 400)
true_values = np.cos(x_values)
chebyshev_values = chebyshev_series(x_values)
power_values = powe

# Plotting
plt.figure(figsize=
plt.plot(x_values,
plt.plot(x_values, chebyshev_values, '--', label='Chebyshev Approximation', linewidth=1.5)
plt.plot(x_values, power_values, '--', label='Power Series Approximation', linewidth=1.5)
plt.legend()
plt.grid(True)
plt.title('Comparison of True Cos(x) vs Approximations')
plt.xlabel('x')
plt.ylabel('cos(x)')
plt.show()

# Calculate the maximum error
error_chebyshev = np.max(np.abs(true_values - chebyshev_values))
error_power = np.max(np.abs(true_values - power_values))
print(f'Maximum error in Chebyshev Approximation: {error_chebyshev}')
print(f'Maximum error in Power Series Approximation: {error_power}')
```

"chebyshev": Unknown word. cSpell

(variable) chebyshev_values: NDArray[floating[Any]]

檢視問題 (Alt+F8) 快速修復... (Ctrl+.)

- here is the figure that comparing the error both Chebyshev and power series.
- we get the error of power series is less than error of power.



7. (20pts)Find the Fourier coefficients for $f(x) = x^2 - 1$ if it is periodic and one period extends from $x = -1$ to $x = 2$.

$a \Rightarrow x = -1$

$b \Rightarrow x = 2$ $\quad L = 3$

$a_0 = \frac{1}{L}\int_a^b f(x)\,dx = \frac{1}{3}\int_{-1}^{2}(x^2-1)\,dx = \frac{1}{3}\left[\frac{1}{3}x^3 - x\right]_{-1}^{2} = 0$ #

$a_n = \frac{2}{L}\int_a^b f(x)\cos\left(\frac{2\pi n x}{L}\right)dx$ $\quad \left(\frac{8}{3} - 2\right) - \left(\frac{-1}{3} + 1\right)$

$= \frac{2}{3}\int_{-1}^{2}(x^2-1)\cos\left(\frac{2\pi n x}{3}\right)dx$ # $\quad = \left(\frac{2}{3}\right) - \left(\frac{2}{3}\right) = 0$

$b_n = 0$ ($\because$ even function)