Table of Contents

Assignemnt

Lagrange's Interpolation

```
% Name        : Mohamed Mafaz
% Roll Number : AM25M009
% Department   : Applied Mechanics
```

Part 1 (Preprocessing / Writing Functions)

Lagrange's Interpolation Function

```
clc;
clear;

function [sum] = LI(xs, ys, number)
    % Straight forward brute force way to find l
    l = [];

    for j = 1:length(xs)
        numerator = 1;
        denominator = 1;
        for i = 1:length(xs)
            if i ~= j       % Or else it will always give 0
                numerator = numerator * (number - xs(i));  % Calculating
Numerator and Denominator differently
                denominator = denominator * (xs(j) - xs(i));
            end
        end
        l(j) = numerator / denominator;
    end

    % This calculates l0 x y0 + l1 x y1 + ....
    sum = 0;
    for i = 1: length(l)
        sum = sum + (ys(i) * l(i));
    end
end

% Lagrange_Basis
function [L] = Lagrange_Basis(xs, j, number)
    L = ones(size(number));
    n = length(xs);
    for i = 1:n
        if i ~= j
            L = L .* (number - xs(i)) / (xs(j) - xs(i));
        end
    end
end
```

Part 2 (Processing / Using the function)

```
x = [4.0, 5.0, 6.0, 7.0, 8.0];
y = [1.58740105, 1.709976, 1.81712059, 1.912931, 2.0];
sample_points = 50;

% Predicting
test_xs = linspace(min(x), max(x), sample_points);

test_ys = zeros(1, sample_points);
for i = 1:sample_points
    test_ys(i) = LI(x, y, test_xs(i));
end

% Printing Error
total_error = 0;
for i = 1:length(x)
    total_error = total_error + (abs(y(i) - LI(x, y, x(i))));
end
fprintf("Total error: %d\n", total_error);
```

```
Total error: 0
```

Part 3 (post processing or plots or results)

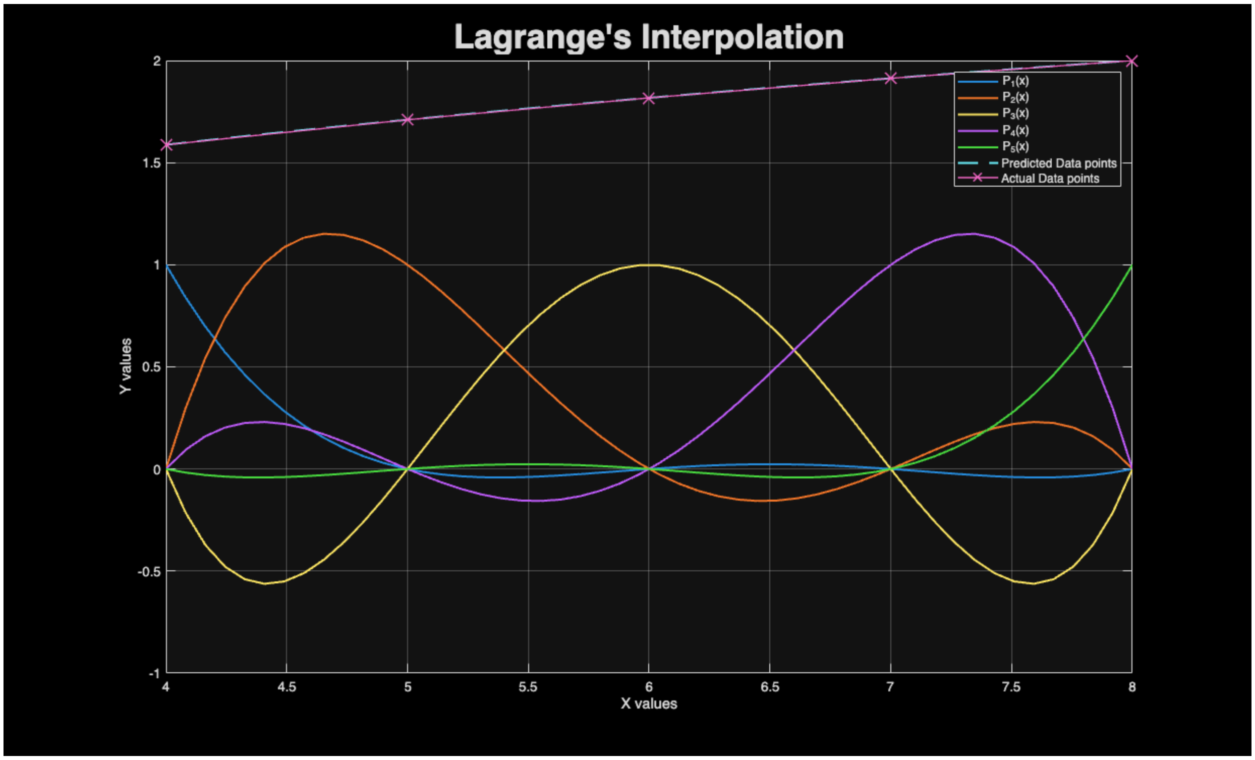
```
for j = 1: length(x)
    lb = Lagrange_Basis(x, j, test_xs);
    plot(test_xs, lb, 'LineWidth', 1.5, 'DisplayName', sprintf('P_{%d}(x)',
j));
    hold on
end

% Plotting predicted Data
plot(test_xs, test_ys, '--', 'LineWidth', 1.5, 'DisplayName', 'Predicted
Data points');
xlabel('X values');
ylabel('Y values');
title("Lagrange's Interpolation", 'FontSize', 25);


hold on

% Plotting actual Data
plot(x, y, 'LineWidth', 1, 'DisplayName', 'Actual Data points', Marker='x',
MarkerSize=12);
```

```
legend show;
grid on;
```



Published with MATLAB® R2025a