INDIAN INSTITUTE OF TECHNOLOGY GANDHINAGAR

MA 202: MATHEMATICS - IV
Semester–II, Academic Year 2022-23

Tutorial Set -3
Question - 12

By

Kush Patel
[20110131]

➔ In question 12, we are asked to estimate the value of average weight for day 12 and 16 by nth order polynomial interpolation. I have used the Lagrange Interpolation method. For nth order polynomial interpolation, we need (n+1) data points (x_1,y_1),...., (x_{n+1},y_{n+1}) We are given input data points as shown below.

Day	0	6	10	13	17	20	28
Sample 1 average weight (mg)	6.67	17.33	42.67	37.33	30.10	29.31	28.74
Sample 1 average weight (mg)	6.67	16.11	18.89	15.00	10.56	9.44	8.89

Table 1: Average weight of oak leaves (two samples)

- ➔ The idea behind the interpolation is to interpret a polynomial which takes on certain values at arbitrary points. Approximation for Lagrange Interpolation method, as we know, is given by:

Lagrange Polynomial:

$$p(x) = L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3 + \dots L_N(x)y_N$$

$$L_k(x) = \frac{(x-x_1)(x-x_2)\dots(x-x_{k-1})(x-x_{k+1})\dots(x-x_N)}{(x_k-x_1)(x_k-x_2)\dots(x_k-x_{k-1})(x_k-x_{k+1})\dots(x_k-x_N)}$$

Numerator of $L_k(x)$ is product of all $(x - x_i)$ except for $(x - x_k)$

Denominator of $L_k(x)$ is product of all $(x_k - x_i)$ except for $(x_k - x_k)$

- ➔ Here, we need to estimate both values for both samples for 1 to nth order polynomial interpolation. We can estimate the most accurate values in the 6th order because we have given 6+1 = 7 sample data. I built a function called Interpolation(N,d) that can take two inputs as previously. One is the value of n in which order of polynomial we need to estimate and the second is d(number of days where we need to estimate the value of average weight.)

- ➔ To run this function and get the output, I ran a loop by varying the value of n(order) and gave the command to find the required values and printed it.

- ➔ To find the maximum average weight for each sample, I have run the loop by dividing the points between 6 to 13 in 0.0001 scale for sample-1 and sample-2. By doing this, the function will check all values corresponding to each and every point and get the values and then by comparing it, it will return output. I made a loop in such a way that it will return the maximum value.

```
% Tutorial-3
% Question-12

% Polynomial Interpolation (Lagrange Interpolation)
for z=1:6
    a = Interpolation_1(z,12);
    disp('The value of W for Sample-1, d=12 and for order')
    disp(z)
    disp('is ')
    disp(a)
    b = Interpolation_2(z,12);
    disp('The value of W for Sample-2, d=12 and for order')
    disp(z)
    disp('is ')
    disp(b)
    c = Interpolation_1(z,16);
    disp('The value of W for Sample-1, d=16 and for order')
    disp(z)
    disp('is ')
    disp(c)
    d = Interpolation_2(z,16);
    disp('The value of W for Sample-2, d=16 and for order')
    disp(z)
    disp('is ')
    disp(d)
end
```

➔ As we can see above, this loop gets every result for 1 to nth order polynomial interpolation for both values day 12 and day 16, for both samples as well and returns the estimated average weight for that.

```
k=0;
max1=0;
% For finding maximum of Sample-1
for i =6:0.0001:13
    a = Interpolation_1(6,i);
    if a>max1
        max1 = a;
        k=i;
    else
        continue
    end
end
disp('The approximate maximum average weight of Sample-1 is = ')
    disp(max1)
    disp('for d = ')
    disp(k)


z = 0;
max2 = 0;
% For finding maximum of Sample-2
for i =6:0.0001:13
    a = Interpolation_2(6,i);
    if a>max2
        max2 = a;
        z=i;
    else
        continue
```

➔ As we can observe above, for both samples, the function itself takes values as an input from 6 to 13 because according to the given input data set points, maximum lies in between 6 to 13. Here, function will return the value and its precision is up to 3 decimal numbers. We can increase the precision by lowering the scale from 0.0001 to 0.00000001. But by doing this, the loop will take some more time because iteration will increase.

➔ In this function, it also checks the maximum value at every iteration. The maximum value keeps updated according to the condition shown above.

```matlab
function v = Interpolation_1(N,d)
format('long')
D = [0, 6, 10, 13, 17, 20, 28];
n = size(D);
n = n(2);
W = [6.67, 17.33, 42.67, 37.33, 30.10, 29.31, 28.74];
L = zeros(1,n);
p=1;
q=1;
for i=1:N
    for j=1:N
        if i == j
            continue;
        else
            p = p * (d - D(j));
            q = q * (D(i) -D(j));
        end
    end
    L(1,i) = W(i) * (p/q);
    p=1;
    q=1;
end
v=0;
for k=1:N
    v = v + L(1,k);
end
end
```

```matlab
function v = Interpolation_2(N,t)
format('long')
T = [0, 6, 10, 13, 17, 20, 28];
n = size(T);
n = n(2);
V = [6.67, 16.11, 18.89, 15.00, 10.56, 9.44, 8.89];
L = zeros(1,n);
p=1;
q=1;
for i=1:N
    for j=1:N
        if i == j
            continue;
        else
            p = p * (t - T(j));
            q = q * (T(i) -T(j));
        end
    end
    L(1,i) = V(i) * (p/q);
    p=1;
    q=1;
end
v=0;
for k=1:N
    v = v + L(1,k);
end
end
```

```
The value of W for Sample-1, d=12 and for order
     6


is
  40.279511698385647

The value of W for Sample-2, d=12 and for order
     6


is
  16.534990958621211

The value of W for Sample-1, d=16 and for order
     6


is
  30.392578463553257

The value of W for Sample-2, d=16 and for order
     6


is
  11.198997792123844
```

Simulated required output

```
The approximate maximum average weight of Sample-1 is =
  42.791748409056609

for d =
  10.326900000000000

The approximate maximum average weight of Sample-2 is =
  19.322018713952513

for d =
   8.873799999999999

>>
```

The approx maximum weight for both samples

➔ By observing above values, we can say that on the 6th order interpolation, the output values of average weight for day 12 and day 16 is more accurate and it also fits according to the given data points because the estimated value for day 12 lies in between day 10 and day 13, and the estimated value for day 16 lies in between day 13 and day 17. It means our estimation is correct.