Numerical Computation - Assignment 8

薛劭杰 1930026143

Q1.

We know that $S(x) = \begin{cases} 4 - \frac{11}{4}x + \frac{3}{4}x^3 & [0, 1] \\ 2 - \frac{1}{2}(x-1) + c(x-1)^2 - \frac{3}{4}(x-1)^3 & [1, 2] \end{cases}$

$S_0^{'}(x) = \frac{9}{4}x^2 - \frac{11}{4}$, $S_1^{'}(x) = -\frac{1}{2} + 2c(x-1) - \frac{9}{4}(x-1)^2$.

$S_0^{''}(x) = \frac{9}{2}x$, $S_1^{''}(x) = 2c - \frac{9}{2}(x-1)$

At point where $x = 1$ which is the junction of the spline:

$S_0^{''}(1) = S_1^{''}(1) \rightarrow \frac{9}{2} = 2c - \frac{9}{2}(1-1) \rightarrow c = \frac{9}{4}$

$S_0^{''}(0) = \frac{9}{2} * 0 = 0$, $S_1^{''}(2) = \frac{9}{2} - \frac{9}{2} * (2-1) = 0$

$S_0^{''}(0) = S_1^{''}(2) = 0 \rightarrow S_0^{''}(t_0) = 0 = S_{n-1}^{''}(t_n)$. In this question, $n = 2$. So, $S(x)$ is a natural cubic spline.

Q2.

Set the natural cubic spline by:

$S(x) = \begin{cases} a_0(x-x_0)^3 + b_0(x-x_0)^2 + c_0(x-x_0) + d_0 & on\ [0,2] \\ a_1(x-x_1)^3 + b_1(x-x_1)^2 + c_1(x-x_1) + d_1 & on\ [2,3] \end{cases}$

which goes through (0,1), (2,3), (3,2). So we have:

$S_0(x_0) = y_0 = d_0 = 1$

$S_0(x_1) = a_0(x_1-x_0)^3 + b_0(x_1-x_0)^2 + c_0(x_1-x_0) + d_0$

$= a_0(2-0)^3 + b_0(2-0)^2 + c_0(2-0) + d_0$

$= 8a_0 + 4b_0 + 2c_0 + 1 = y_1 = 3$

$\rightarrow 8a_0 + 4b_0 + 2c_0 = 2$

$S_1(x_1) = y_1 = d_1 = 3$

$S_0(x_2) = a_1(x_2-x_1)^3 + b_1(x_2-x_1)^2 + c_1(x_2-x_1) + d_1$

$= a_1(3-2)^3 + b_1(3-2)^2 + c_1(3-2) + d_1$

$= a_1 + b_1 + c_1 + 3 = y_2 = 2$

$\rightarrow a_1 + b_1 + c_1 = -1$

Then we have $S_0^{'}(x_1) = S_1^{'}(x_1)$:

$S_0^{'}(x_1) = 3a_0(x_1-x_0)^2 + 2b_0(x_1-x_0) + c_0 = 12a_0 + 4b_0 + c_0$

$S_1^{'}(x_1) = 3a_1(x_1-x_1)^2 + 2b_1(x_1-x_1) + c_1 = c_1$

$\rightarrow 12a_0 + 4b_0 + c_0 = c_1$

Then we have $S_0^{''}(x_1) = S_1^{''}(x_1)$:

$S_0^{''}(x_1) = 6a_0(x_1-x_0) + 2b_0 = 12a_0 + 2b_0$

$S''_1(x_1) = 6a_1(x_1 - x_1) + 2b_1 = 2b_1$
$\rightarrow 6a_0 + b_0 = b_1$

Because the spline is natural cubic spline, then it has $S''_0(x_0) = 0 = S''_{n-1}(t_n) = S''_2(x_2)$:
$S''_0(x_0) = 6a_0(x_0 - x_0) + 2b_0 = 2b_0$
$S''_1(x_2) = 6a_1(x_2 - x_1) + 2b_1 = 6a_1 + 2b_1$
$\rightarrow 3a_1 + b_1 = b_0 = 0$

To conclude, we know $b_0 = 0$, $d_0 = 1$, $d_1 = 3$. And we have five function and there only 5 unknow, so we can get the result:

$$\begin{cases} 3a_1 + b_1 = 0 \\ 6a_0 = b_1 \\ a_1 + b_1 + c_1 = -1 \\ 12a_0 + c_0 = c_1 \\ 8a_0 + 2c_0 = 2 \end{cases} \rightarrow \begin{cases} a_0 = -\frac{1}{6} \\ a_1 = \frac{1}{3} \\ b_1 = -1 \\ c_0 = \frac{5}{3} \\ c_1 = -\frac{1}{3} \end{cases}$$

The natural cubic spline is:

$$S(x) = \begin{cases} -\frac{1}{6}(x - x_0)^3 + \frac{5}{3}(x - x_0) + 1 & on\ [0,2] \\ \frac{1}{3}(x - x_1)^3 - (x - x_1)^2 - \frac{1}{3}(x - x_1) + 3 & on\ [2,3] \end{cases}$$

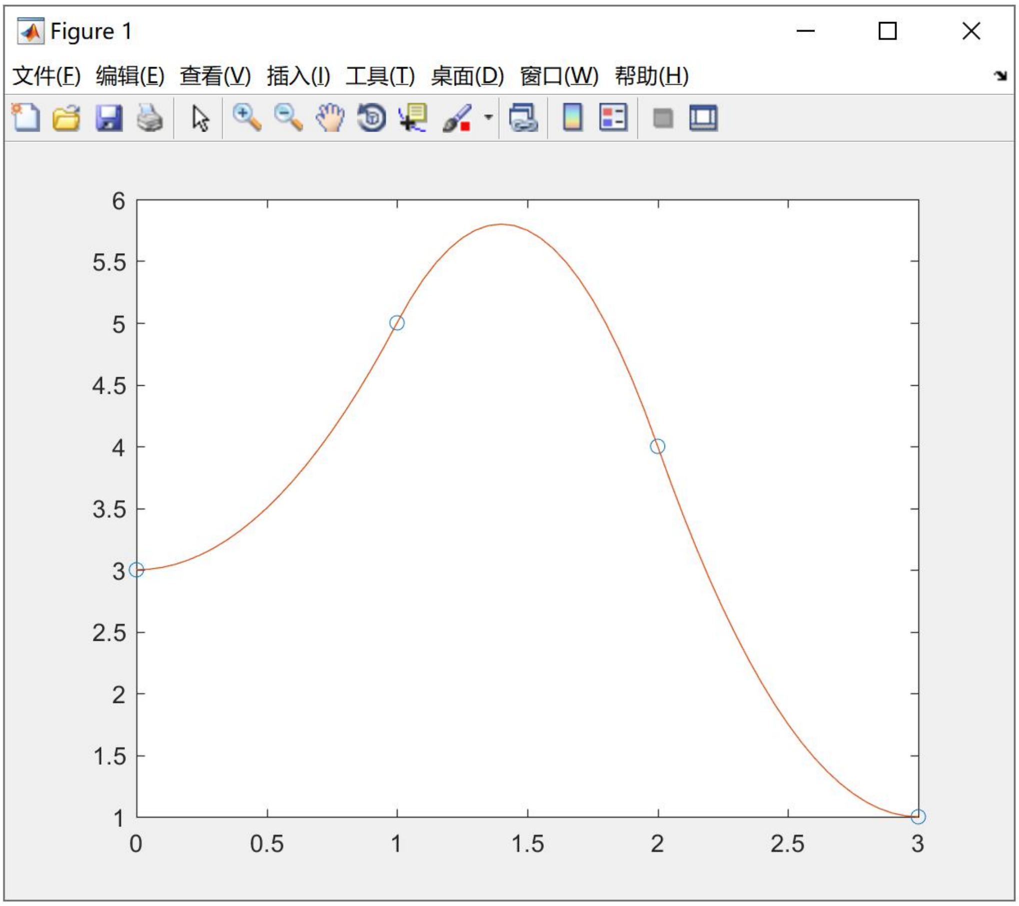
Q3.
Base condition: $b_0 = 0$:

```
function v = piecequad(x, y, z0, u);
    n = length(x);
    % The first dimension is n minus 1
    % Since n points has n-1 segment
    a= zeros(n-1, 1);
    b = zeros(n-1, 1);

    % h is also a vector and h_k = x_k+1 - x_k
    h = diff(x);
    % delta is also a vector and delta_k = y_k+1 - y_k
    delta = diff(y);

    % Natural, the first derivative of the first point is 0
    b(z0) = 0;

    % Calculate each b
    for i=1:n-1
        b(i+1) = ((2 * delta(i)) / h(i)) - b(i);
    end

    % Calculate each a by conclusion
    for i=1:n-1
        a(i) = (b(i+1) - b(i))/(2 * h(i));
    end
```

```
    % Initialize
    [r ,c] = size(u);
    splinevals = zeros(size(u));
    v = zeros(c,1);

    % Detemine which segment the u(i) in
    for i=1:c
        for j=1:n-1
            if x(j) <= u(i) && u(i) < x(j+1)
                k=j;
            end
            v(i) = a(k)*((u(i)-x(k))^2)+b(k)*((u(i)-x(k)))+y(k);
        end

    end

    for i=1:c
        splinevals(i) = v(i);
    end
    plot(x,y,'o',u,splinevals,'-');
end
```

Q4.

Because it is a natural spline, so base condition: $b_0 = 0$ (The first derivative of the first point is 0):




```
>> u=0:0.05:3;
>> piecequad([0;1;2;3],[3;5;4;1], 1, u);
```

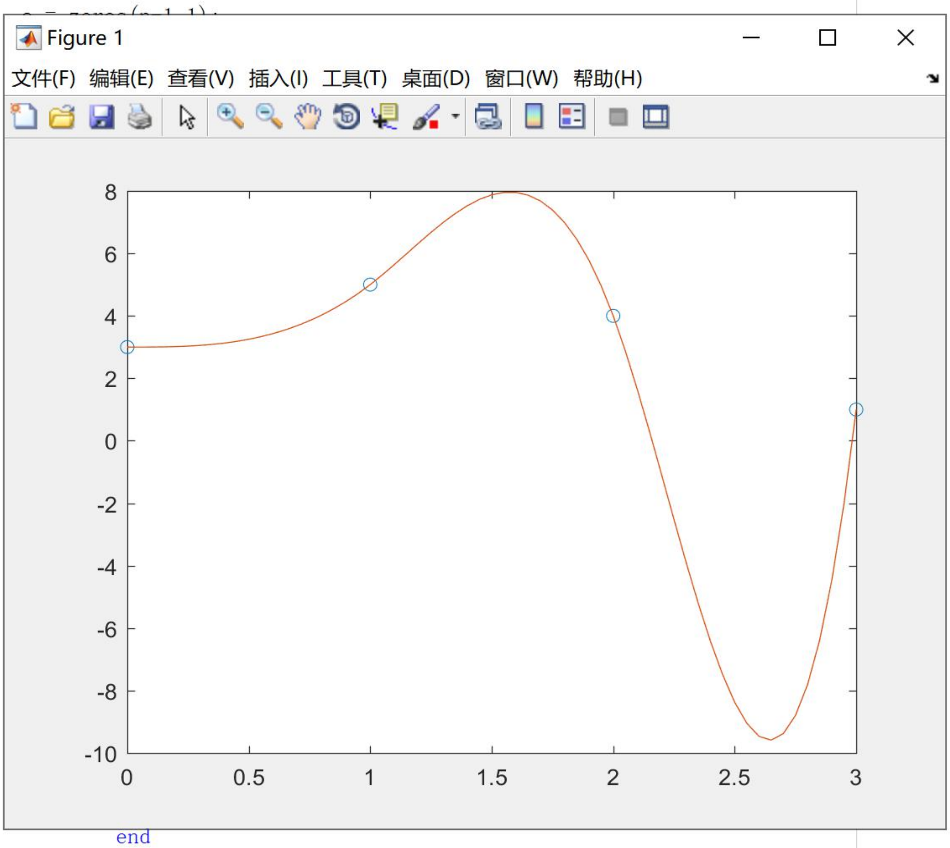
Displace the a, b and get coefficient for this spline is:

$$S(x) = \begin{cases} 2x^2 + 3 & on\ [0,1] \\ -5(x-1)^2 + 4(x-1) + 5 & on\ [1,2] \\ 3(x-2)^2 - 6(x-2) + 4 & on\ [2,3] \end{cases}$$

Then the cubic one is: (different part with quardatic)

```
a(1) = delta(1)/(h(1)^2);
b(1)=0;
c(1)=0;

for i=1:n-2
    c(i+1)=3*a(i)*h(i)^2 + 2*b(i)*h(i)+c(i);
    b(i+1)=3*a(i)*h(i)+b(i);
    a(i+1)=(delta(i+1)-c(i+1)-b(i+1)*h(i+1))/(h(i+1)^2);
end
```

```
function v = piececubic(x,y,u)
    n =length(x);
```




```
        end
    end
    v(i) = a(k)*((u(i)-x(k))^3)+b(k)*((u(i)-x(k))^2)+c(k)*(u(i)-x(k))+y(k);
```

Displace the a, b and get coefficient for this spline is:

$$Q(x) = \begin{cases} 2x^3 + 3 & on\ [0,1] \\ -13(x-1)^3 + 6(x-2)^2 + 6(x-1) + 5 & on\ [1,2] \\ 51(x-2)^3 - 33(x-2)^2 - 21(z-2) + 4 & on\ [2,3] \end{cases}$$

Q5.

The vector of the coefficients for the best quadratic fit the data:

The first row for A is $(1,1,\ldots,1)^T$

The second row of A is $(x_1, x_2, \ldots, x_n)^T$

The third row of A is $\left(x_1^2, x_2^2, \ldots, x_n^2\right)^T$

```
function RMSE = coequad(x,y)
    plot(x,y,'.');

    % Hold on and we can show more data in the plot
    hold on;
    A = zeros(length(x),3);
    A(:,1) = ones(length(x),1);
    A(:,2) = x;
    A(:,3) = x.^2;

    D = A'*A;
    e = A'*y;
    % inv((A'A)) * A'y
    c = D\e;


    % get the fit result
    fit_y = c(3).*x.^2 + c(2).*x + c(1);

    % plot out
    plot(x,fit_y,'-');

    % calculate the RMSE
    RMSE = sqrt(mean((fit_y - y).^2));
end
```

Q6.

The vector of the coefficients for the best $ae^x + be^{-x}$ fits the data:

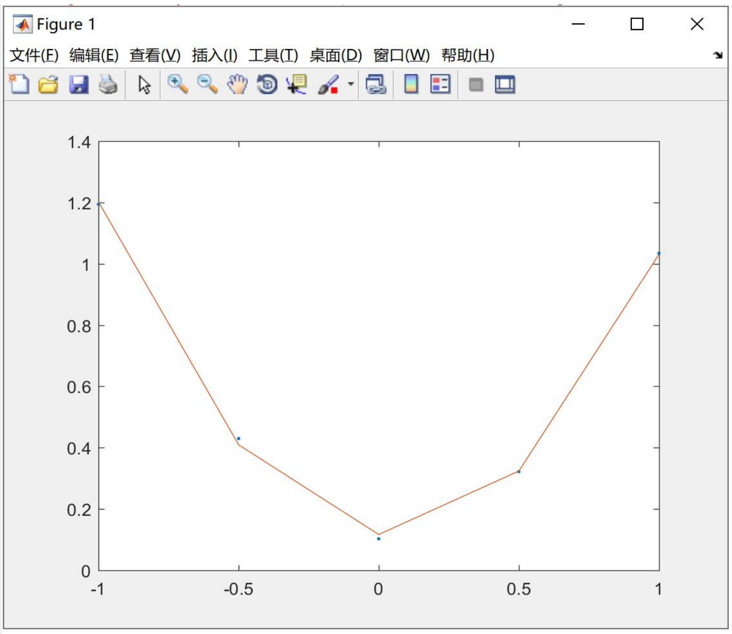
The first row for A is $(e^{x_1}, e^{x_2} \ldots, e^{x_n})^T$

The second row of A is $(-e^{x_1}, -e^{x_2} \ldots, -e^{x_n})^T$

```
function RMSE = coeexp(x,y)
    plot(x,y,'.');
    % Hold on and we can show more data in the plot
    hold on;

    % Create and initialize a matrix A
    A = zeros(length(x),2);
    A(:,1) = exp(x);
    A(:,2) = exp(-x);

    D = A' * A;
    e = A' * y;

    % inv((A'A)) * A'y
    c = D \ e;

    % get the fit result
    fit_y = c(1).*exp(x)+c(2).*exp(-x);
    plot(x,fit_y,'-');

    % calculate the RMSE
    RMSE = sqrt(mean((fit_y - y).^2));
end
```

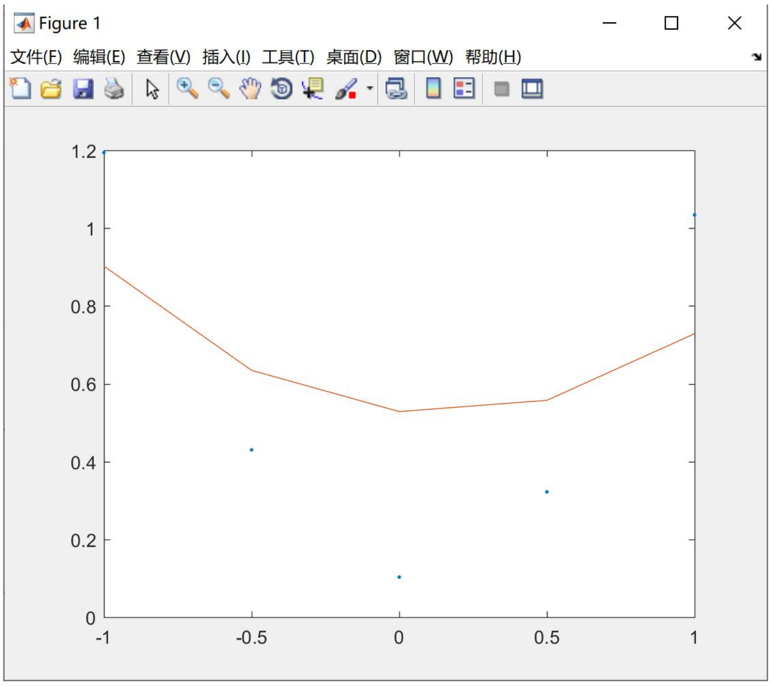
7. Get the RMSE result by the two fits:

The quadratic fits:




```
>> x = [-1; -0.5; 0; 0.5; 1];
>> y = [1.194; 0.43; 0.103; 0.322; 1.034];
>> coequad(x, y)

ans =

    0.0117
```

The $ae^{x} + be^{-x}$ fits:




```
>> coeexp(x,y)

ans =

    0.3022
```

$quad_{RMSE} = 0.0117, \ \exp_{RMSE} = 0.3022$

The RMSE of the quadratic fits is much smaller than $ae^{x} + be^{-x}$ one fits, so the quadratic model is better one.

So, display the quadratic one:

```
>> x = [-1; -0.5; 0; 0.5; 1];
>> y = [1.194; 0.43; 0.103; 0.322; 1.034];
>> coequad(x,y)


c =


    0.1169
   -0.0856
    0.9994
```

The quadratic model is: $f(x) = 0.1169 - 0.0856x + 0.9994x^2$.