Contents

- compute the extended knot vector
- compute coefficient vector for spline that approximates u
- print extended knot vector and coefficient vector
- compute errors
- plot s

```
% Nikhil Jayswal
% MATH 3890
% Machine Problem 5
% 22 Feb 2021

clc; clear; close all

% interval limits
a = 0;
b = 1;

% u'' + pu' + qu = f
p = @(x) exp(x);
q = @(x) sin(x.*x);

u = @(x) cos(x) + sin(3*x);
ux = @(x) -sin(x) + 3*cos(3*x);
uxx = @(x) -cos(x) - 9*sin(3*x);

f = @(x) uxx(x) + p(x).*ux(x) + q(x).*u(x);

% get inputs from user
% d = input('Enter the value of d: ');
% k = input('Enter the value of k: ');
% m = input('Enter the value of m: ');
d = 3;
k = 17;
m = 10;

% knot vector
knots = linspace(a, b, k+2);
```

compute the extended knot vector

dimension of spline space length of knot vector = k+2 -> spline consists of (k+1) pieces (k) interior points

```
dim = (k+1)*(d+1) - k*d;
```

```
% construct extended knot vector
y = zeros(1, dim+d+1);
% first (d+1) points
y(1:d+1) = knots(1);
% last (d+1) points
y(dim+1:dim+d+1) = knots(end);
% points in between
y(d+2:dim) = knots(2:end-1);
```

compute coefficient vector for spline that approximates u

```
ua = u(a); ub = u(b);
c = lsc(d, y, m, p, q, f, ua, ub);
```

print extended knot vector and coefficient vector

```
fprintf('The extended knot vector is: \n\n')
disp(y')
fprintf('\n\n')
fprintf('The coefficients are: \n\n')
disp(c)
```

```
The extended knot vector is:

         0
         0
         0
         0
    0.0556
    0.1111
    0.1667
    0.2222
    0.2778
    0.3333
    0.3889
    0.4444
    0.5000
    0.5556
    0.6111
    0.6667
    0.7222
    0.7778
    0.8333
    0.8889
```

```
    0.9444
    1.0000
    1.0000
    1.0000
    1.0000
```

```
The coefficients are:

    1.0000
   20.5501
   18.4496
   15.4803
   12.6866
   10.0624
    7.6014
    5.2972
    3.1439
    1.1357
   -0.7326
   -2.4658
   -4.0681
   -5.5435
   -6.8953
   -8.1265
   -9.2402
  -10.2391
  -11.1263
  -11.6454
    0.6814
```

compute errors

```
t = linspace(a, b, 301);
% evaluate
val = sval2(d, y, c, t);
% print maxmimum norm of (u-s)
e = u(t)-val;
fprintf('\nThe maximum error is: %f\n', norm(e, inf));
% print RMS error
fprintf('\nThe RMS error is: %f\n', erms(e));
```

```
The maximum error is: 17.809098
```

```
The RMS error is: 9.063477
```

plot s

```
plot(t, val, 'LineWidth', 1);
hold on
plot(t, u(t), 'LineWidth', 2);
xlabel('t')
legend('Spline s', 'Function u', 'Location', 'best')
```