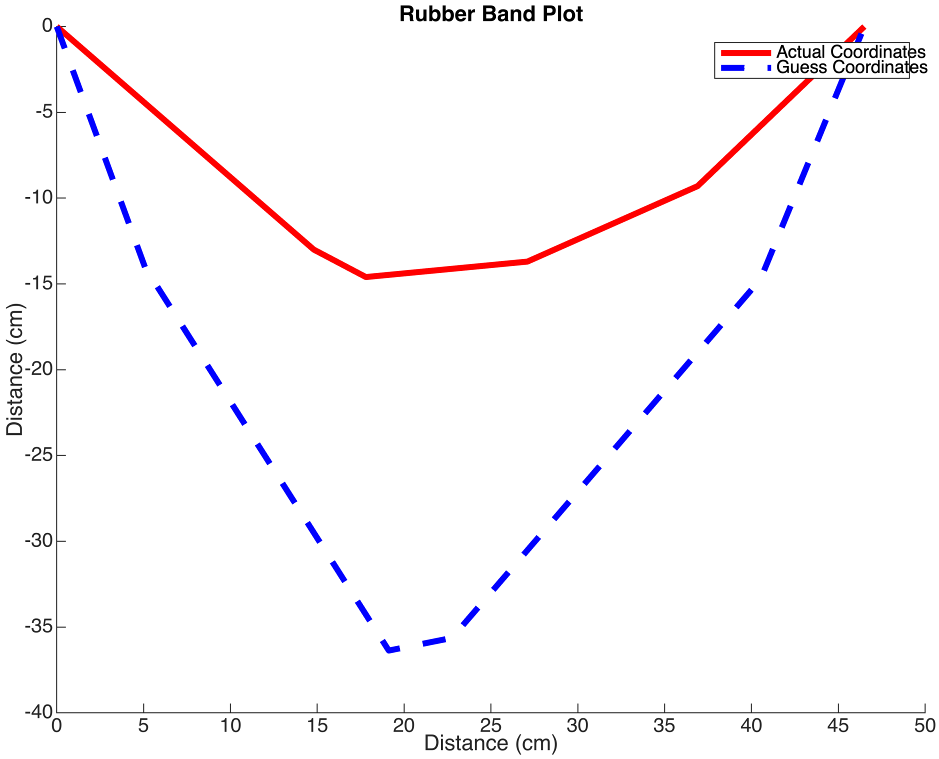
```
%initialize the system parameters
%which contains parameters describing behavior/measurements of bridge
% param_struct.r0 = [x_0;y_0]: coordinates of leftmost vertex
% param_struct.rn = [x_n;y_n]: coordinates of rightmost vertex
% param_struct.num_links: number of rubber bands in bridge
% param_struct.k_list = [k_1;...;k_n]: list of stiffnesses
% param_struct.l0_list = [l0_1;...;l0_n]: list of natural lengths
% param_struct.m_list = [m_1;...;m_(n-1)]: list of weight masses
% param_struct.g = 9.8 m/sec^2: gravitational acceleration
coords = [14.8; -13; 17.8; -14.6; 27.1; -13.7; 36.9; -9.3];


param_struct = struct();
param_struct.r0 = [0;0];
param_struct.rn = [46.5;0];
param_struct.num_links = 5;
param_struct.k_list = [106.556, 26.28, 150.142, 38.67, 77.77];
param_struct.l0_list = [8, 8.4, 2, 17.4    7.3];
param_struct.m_list = [30, 46, 25, 33];
param_struct.g = 9.8;
param_struct.iter =100;

x_list = [0;coords(1:2:end);46.5]';

% Extract y_list (elements at even indices)
y_list = [0;coords(2:2:end);0]';

U_RB_total = total_RB_potential_func(coords,param_struct);
U_total = total_potential_func(coords,param_struct);
clf
close all
[x_list_guess,y_list_guess] = generate_shape_prediction(param_struct);
figure()
hold on
plot(x_list, y_list, LineWidth=3, Color='r')
plot(x_list_guess', y_list_guess', LineStyle='--', LineWidth=3, Color='b')
legend('Actual Coordinates', 'Guess Coordinates')
ylabel('Distance (cm)')
xlabel('Distance (cm)')
title('Rubber Band Plot')
```


Rubber Band Plot
Actual Coordinates
Guess Coordinates
Distance (cm)
Distance (cm)
0
-5
-10
-15
-20
-25
-30
-35
-40
0
5
10
15
20
25
30
35
40
45
50

```
data_mat=load_excel_example();
```

```
Warning: Column headers from the file were modified to make them valid MATLAB identifiers
before creating variable names for the table. The original column headers are saved in the
VariableDescriptions property.
Set 'VariableNamingRule' to 'preserve' to use the original column headers as table variable names.
    RubberBandLabelOrColor              Var2               Measurement_1    Measurement_2    Measurement_3
    ______________________    _________________________    _____________    _____________    _____________

      {'Rubber Band #1'}      {'Mass (g)'             }           35               61              113
      {0×0 char        }      {'Stretched Length (cm)'}          8.2              8.3              8.9
      {'Rubber Band #2'}      {'Mass (g)'             }           50              102              137
      {0×0 char        }      {'Stretched Length (cm)'}          8.7             10.2             11.6
      {'Rubber Band #3'}      {'Mass (g)'             }           50               85              137
      {0×0 char        }      {'Stretched Length (cm)'}          2.2              2.4              2.8
      {'Rubber Band #4'}      {'Mass (g)'             }           52               87              137
      {0×0 char        }      {'Stretched Length (cm)'}         17.8             18.6             19.9
      {'Rubber Band #5'}      {'Mass (g)'             }           26               78              113
      {0×0 char        }      {'Stretched Length (cm)'}          7.5              8.6              8.7
      {0×0 char        }      {0×0 char               }          NaN              NaN              NaN
      {0×0 char        }      {0×0 char               }          NaN              NaN              NaN
      {0×0 char        }      {0×0 char               }          NaN              NaN              NaN
      {0×0 char        }      {0×0 char               }          NaN              NaN              NaN
      {0×0 char        }      {0×0 char               }          NaN              NaN              NaN
      {0×0 char        }      {'Rubber Band #1'       }          NaN              NaN              NaN
      {'natural length'}      {'8cm'                  }          8.4                2             17.4
      {'K'             }      {'0.9854'               }       0.0091           0.0065           0.0253

    Measurement_1    Measurement_2    Measurement_3    Measurement_4
    _____________    _____________    _____________    _____________

          35               61              113              163
         8.2              8.3              8.9              9.3
          50              102              137              163
         8.7             10.2             11.6             12.9
          50               85              137              163
         2.2              2.4              2.8              2.9
          52               87              137              163
        17.8             18.6             19.9             20.6
          26               78              113              163
         7.5              8.6              8.7              9.1

   35.0000   61.0000  113.0000  163.0000
    8.2000    8.3000    8.9000    9.3000
   50.0000  102.0000  137.0000  163.0000
    8.7000   10.2000   11.6000   12.9000
   50.0000   85.0000  137.0000  163.0000
    2.2000    2.4000    2.8000    2.9000
   52.0000   87.0000  137.0000  163.0000
   17.8000   18.6000   19.9000   20.6000
   26.0000   78.0000  113.0000  163.0000
    7.5000    8.6000    8.7000    9.1000
```

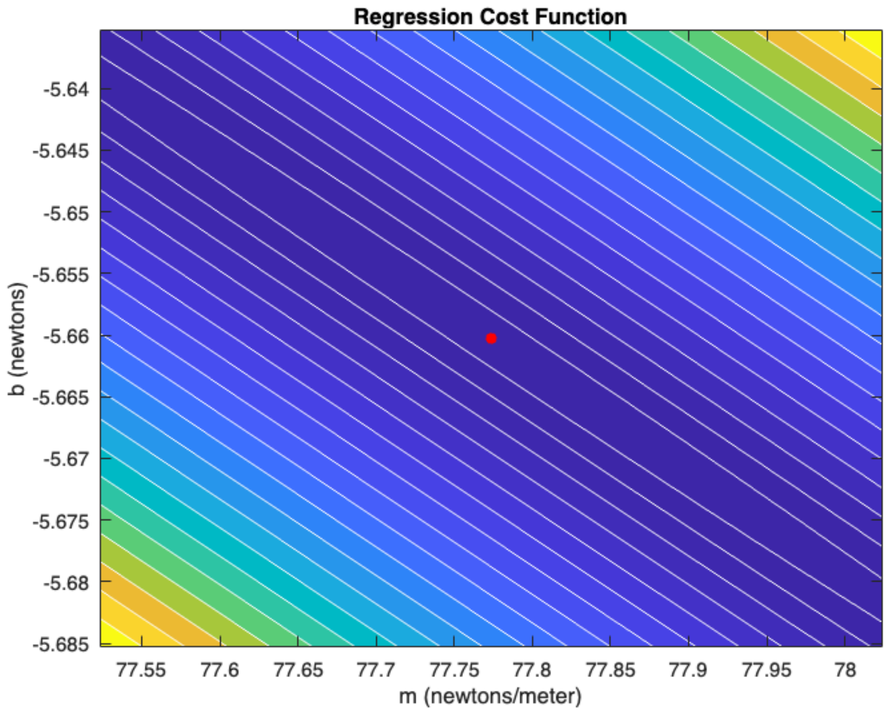
```
%day17
rubber_band_num = 5;
%gravitational acceleration in m/sec^2
g = 9.8;
%label of rubber band we want to measure k & l0 for
```

```
%compute the corresponding row indices for
%the mass/length measurements in the table
mass_row = rubber_band_num*2-1;
length_row = rubber_band_num*2;
%extract mass from table and convert to kg
mass_vals = data_mat(mass_row,:)/1000;
%compute force exerted by weights in Newtons
force_vals = g*mass_vals;
%extract stretched length from table and convert to meters
length_vals = data_mat(length_row,:)/100;
%Y is just the force measurements transposed from a row to a column
Y = force_vals';
%transpose the length measurements from a row to a column
X = length_vals';
%construct A from X (and using the ones function)
A = [X,ones(size(X))];

%compute the line of best fit
%note how we use \ instead of inv()
q = (A'*A)\(A'*Y);
%extract the slope and intercept values
m = q(1); b = q(2);
%compute the stiffness and natural length from m and b
k = m;
l0 = -b/m; % natural length

x = linspace(length_vals(1), length_vals(end), 10)
```

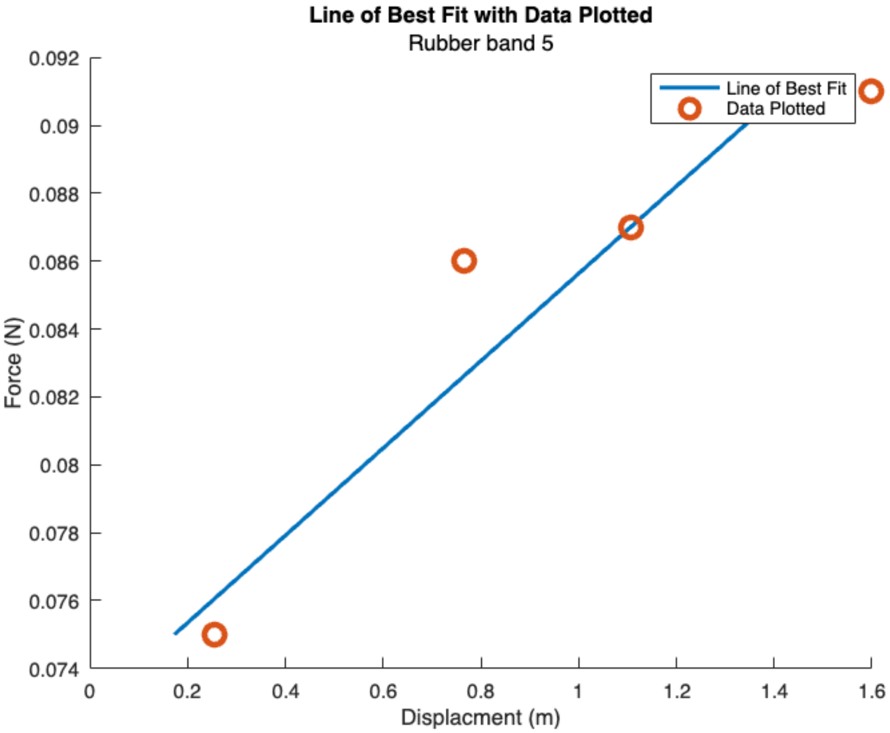
```
x = 1×10
    0.0750    0.0768    0.0786    0.0803    0.0821    0.0839    0.0857    0.0874 ···
```

```
fig2(m,b,force_vals,length_vals)
```

```
m_opt =
77.7734
b_opt =
-5.6603
levels = 1×21
    0.1089    0.1089    0.1089    0.1090    0.1092    0.1094    0.1096    0.1099 ···
```



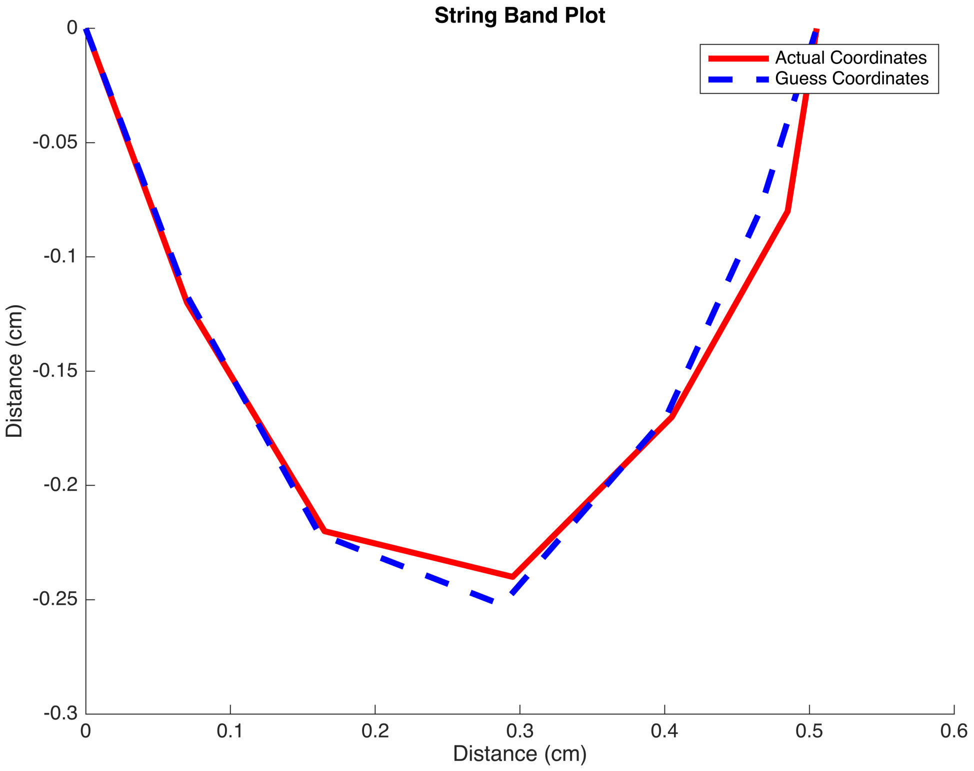
```
figure()
hold on
plot((m*x)+b, x, LineWidth=2)
plot(force_vals, length_vals, 'o',MarkerSize =10, LineWidth=3)
legend('Line of Best Fit', 'Data Plotted')
ylabel('Force (N)')
xlabel('Displacment (m)')
title('Line of Best Fit with Data Plotted')
subtitle('Rubber band 5')
```



```
Warning: Graphics timeout occurred. To share details of this issue with MathWorks technical
support, please include that this is an unresponsive graphics client with your service request.
Warning: Graphics timeout occurred. To share details of this issue with MathWorks technical
support, please include that this is an unresponsive graphics client with your service request.
Warning: Graphics timeout occurred. To share details of this issue with MathWorks technical
support, please include that this is an unresponsive graphics client with your service request.
Warning: Graphics timeout occurred. To share details of this issue with MathWorks technical
support, please include that this is an unresponsive graphics client with your service request.
Warning: Graphics timeout occurred. To share details of this issue with MathWorks technical
support, please include that this is an unresponsive graphics client with your service request.
Warning: Graphics timeout occurred. To share details of this issue with MathWorks technical
support, please include that this is an unresponsive graphics client with your service request.
Warning: Graphics timeout occurred. To share details of this issue with MathWorks technical
support, please include that this is an unresponsive graphics client with your service request.
```

```
% k 5 = 77.77
% k 4 = 38.67
% k 3 = 150.142
% k 2 = 26.28
% k 1 = 106.556
```

```
%coords: vector of vertex positions from i=1 to i=(n-1)
% [x_1;y_1;...;x_(n-1),y_(n-1)]
%param_struct: struct containing parameters of the bridge
% param_struct.r0 = [x_0;y_0]: coordinates of leftmost vertex
% param_struct.rn = [x_n;y_n]: coordinates of rightmost vertex
% param_struct.num_links: number of links in bridge
% param_struct.l0_list = [l_1;...;l_n]: list of link lengths
% param_struct.m_list = [m_1;...;m_(n-1)]: list of weight masses
% param_struct.g = 9.8 m/sec^2: gravitational acceleration

%(xA, yA): coordinates of first vertex
%(xB, yB): coordinates of second vertex
%l_max: maximum allowable distance between two vertices
coords = [7; 12; 16.5; 22; 29.5; 24; 40.5; 17; 48.5; 8]./100;
param_struct = {};
param_struct.r0 = [0;0];
param_struct.rn = [50.5;0]./100;
param_struct.num_links = 6;
param_struct.l0_list = [13.5; 14; 13; 14; 11; 9]./100; %in cm
param_struct.m_list = [21, 35, 38, 26, 25]./1000; % in kilograms
param_struct.g = 9.8;

x_list = [0;coords(1:2:end);0.505]';

% Extract y_list (elements at even indices)
y_list = [0;coords(2:2:end);0]'.*-1;

[x_list_guess,y_list_guess] = generate_shape_prediction_GD(param_struct);


figure()
hold on
plot(x_list, y_list, LineWidth=3, Color='r')
plot(x_list_guess', y_list_guess', LineStyle='--', LineWidth=3, Color='b')
legend('Actual Coordinates', 'Guess Coordinates')
ylabel('Distance (cm)')
xlabel('Distance (cm)')
title('String Band Plot')
```

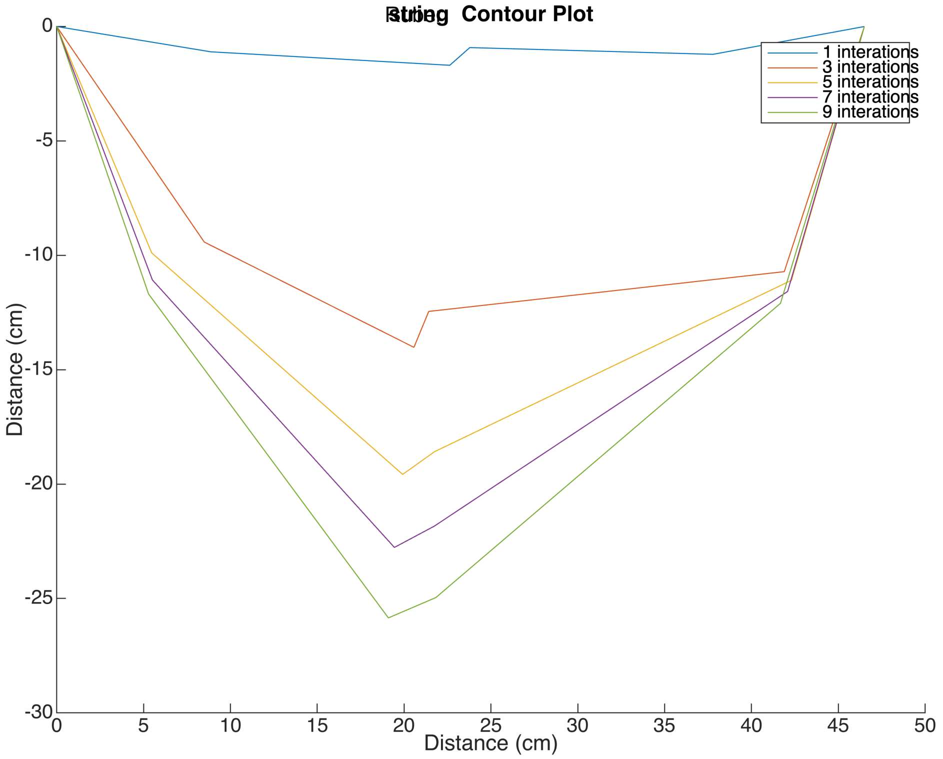

String Band Plot
Actual Coordinates
Guess Coordinates
Distance (cm)
Distance (cm)
0
-0.05
-0.1
-0.15
-0.2
-0.25
-0.3
0
0.1
0.2
0.3
0.4
0.5
0.6

```
%fig 7 week 11
```

```
%initialize the system parameters
%which contains parameters describing behavior/measurements of bridge
% param_struct.r0 = [x_0;y_0]: coordinates of leftmost vertex
% param_struct.rn = [x_n;y_n]: coordinates of rightmost vertex
% param_struct.num_links: number of rubber bands in bridge
% param_struct.k_list = [k_1;...;k_n]: list of stiffnesses
% param_struct.l0_list = [l0_1;...;l0_n]: list of natural lengths
% param_struct.m_list = [m_1;...;m_(n-1)]: list of weight masses
% param_struct.g = 9.8 m/sec^2: gravitational acceleration
coords = [14.8; 17.8; 27.1; 36.9; -13; -14.6; -13.7; -9.3]
```

```
coords = 8×1
   14.8000
   17.8000
   27.1000
   36.9000
  -13.0000
  -14.6000
  -13.7000
   -9.3000
```

```


param_struct = struct();
param_struct.r0 = [0;0];
param_struct.rn = [46.5;0];
param_struct.num_links = 5;
param_struct.k_list = [106.556, 26.28, 150.142, 38.67, 77.77];
param_struct.l0_list = [8, 8.4, 2, 17.4    7.3];
param_struct.m_list = [30, 46, 25, 33];
param_struct.g = 9.8;
param_struct.iter =100;



U_RB_total = total_RB_potential_func(coords,param_struct);
U_total = total_potential_func(coords,param_struct);
clf
close all
figure()
hold on
for i = 1 : 2 : 10
param_struct.iter =i;
[x_list,y_list] = generate_shape_prediction(param_struct);
plot(x_list',y_list')
end
legend('1 interations','3 interations','5 interations','7 interations','9
interations')
ylabel('Distance (cm)')
xlabel('Distance (cm)')
title('string  Contour Plot')
```


Rubber
String Contour Plot
1 interations
3 interations
5 interations
7 interations
9 interations
Distance (cm)
0
-5
-10
-15
-20
-25
-30
0
5
10
15
20
25
30
35
40
45
50
Distance (cm)