**ChE 310 Problem Set 4 (15 pts) Due Wed 2/13/19**

Collect all m-files in a single .zip file and upload the .zip file to the course webpage by midnight on Wednesday, February 13, 2019. Please note any collaborations as comments in your m-files. Each student must upload their own individual copy of the work.

**4\_1 (4 pts)** Perform steps in a script file ps4\_1.m; upload this script and any additional files that are necessary to run your script.

We collect the following steady-state temperature data (°C) on a two-dimensional computer chip with a heating source near (8,8):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  | 20.00 | 25.00 | 30.00 | 35.00 | 40.00 |
|  | 30.00 | 27.07 | 30.39 | 38.78 | 55.00 |
|  | 40.00 | 35.03 | 38.43 | 48.90 | 70.00 |
|  | 50.00 | 48.15 | 53.50 | 64.69 | 85.00 |
|  | 60.00 | 70.00 | 80.00 | 90.00 | 100.00 |

We have a temperature-sensitive component located at , and need to ensure that the temperature at that component isn’t too high (failure threshold 45 °C). Use a *bivariate interpolating spline* as described below to determine the temperature at , .

1. First, plot the 3-D surface of the chip and label axes accordingly.
2. For each fixed value of y, calculate a cubic spline that describes the temperature as a function of x and evaluate that spline at x = 4.2. (You’ll generate 5 splines and generate 5 total temperatures at the points (, ) for the 5 values.)
3. Now, use these data at the fixed value to calculate a cubic spline that describes the temperature as a function of for the fixed value . Evaluate this spline at to obtain your final temperature.

Output the component temperature to the command line, along with the value predicted by *piecewise bilinear interpolation*. (Ensure these outputs are labeled to clearly identify which is which.) Will the component fail? Also ensure your plot displays when your script is run.

**4\_2 (4 pts)** Perform steps in a script file ps4\_2.m; upload this script and any additional files that are necessary to run your script.

Consider the function ; this is the built-in “humps” function in MATLAB. Generate 8 data points (*x*,*y*) for this function using uniformly-spaced *x* values on the interval [0 1].

In a 2x2 subplot, plot the original function, all 8 data points, and a representation of the following (each in its own panel):

1. Piecewise linear interpolation
2. Full polynomial interpolation
3. Not-a-knot cubic spline interpolation
4. Clamped cubic spline interpolation, using the true derivative of at x = 0 and 1. (Note: the analytical derivative of this function is:

Finally, use the four interpolating methods to interpolate the values of at 0.1, 0.5, and 0.9. Calculate the absolute value of the percent error for each interpolating method relative to the true value () and print the resulting percent errors to the command line. The percent errors for each of the four interpolations, along with the 2x2 subplot, should be the only output of running your script.

**4\_3 (4 pts)** Perform steps in a script ps4\_3.m.

Solve Problem 18.2 from the text using centered finite differences to compute the first and second derivatives. Use a subplot to plot the spline fit (T vs. depth), first derivative vs. depth, and second derivative vs. depth on the same figure. We have not covered root finding yet, so find the inflection point of interest by eye. You can zoom in on the plot to get a pretty good estimate. Record the coordinates of the inflection point on your first-as and second-derivative plots by printing an arrow showing the value of interest. Print the heat flux value to the screen.

**4\_4 (3 pts)** Perform steps in a script ps4\_4.m.

The velocity () of air flowing past a flat surface is measured at several distances () away from the surface. Newton’s viscosity law describes the shear stress () at the surface ():

Assume a dynamic viscosity of

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *x* (m) | 0 | 0.002 | 0.006 | 0.012 | 0.018 | 0.024 |
| *v* (m/s) | 0 | 0.287 | 0.999 | 2.215 | 3.448 | 4.699 |

Use three methods to evaluate the derivative at and calculate . Output your value of for each method, and include a recommendation as to which you think will work best.

**Group Credit –** Interact with group members on slack. Collaborate on difficult problems.