University of Colorado
Department of Computer Science

Numerical Computation CSCI 3656

Spring 2016

Problem Set 9

Issued: 15 March 2016

Due: 29 March 2016

The purpose of the first three problems in this assignment is to explore how to deal with sparse, real-world data in interpolation methods. You will need your interpolation code from previous problem sets in order to do this assignment, and you will need to do some creative thinking. Do *not* start this one the night before it's due!

1. *[5 pts]* Here are (x, y, z) coordinates for 22 points on the Worthington glacier near Valdez, Alaska — twelve on the top surface and ten on the bottom:

top surface	bottom surface
(33.44 87.93 105.88)	(15.59 35.07 12.88)
(8.81 84.07 103.11)	(38.57 37.17 13.33)
(15.62 34.83 105.98)	(61.10 67.15 17.31)
(40.16 38.71 108.13)	(58.97 92.05 19.09)
(61.45 67.07 108.12)	(36.98 63.24 16.51)
(58.81 91.44 107.72)	(64.45 42.66 20.01)
(36.97 63.29 107.14)	(89.18 46.85 27.71)
(64.71 42.38 109.07)	(66.87 18.48 14.24)
(89.11 46.49 109.93)	(65.90 31.93 21.0)
(67.24 18.32 109.99)	(76.55 44.51 22.0)
(65.90 31.93 109.51)	
(76.55 44.51 109.91)	

   Plot the points on the top surface (just the *points*, not the whole surface!) using your favorite 3D plotting tool. Repeat for the bottom set. Notice how changing the perspective affects your ability to make any sense of the surface from the points. If your language/package supports it, try connecting the points in a "wireframe" plot. Turn in one copy of the best version of each of the two surfaces, and please write down what language/command you used.

2. *[20 pts]* Your task in this problem is to perform a 3D surface interpolation of these data. Fitting a smooth surface to these data is nontrivial; each triple of numbers in the table above required laser/GPS surveying (and, for the bottom points, it also entailed drilling a hole completely through the glacier!), so the points are unevenly spaced and very sparse — too sparse for most surface interpolation functions to deal with. You will have to improvise, creatively, in order to solve this problem well.

   Using whatever *smooth* surface interpolation function is available in your favorite programming environment (e.g., Matlab's `griddata` or Mathematica's `Interpolation` command), fit a surface to the *top* points. The surface may pass through the points or not, as you wish. **NOTE!** because the glacier data are sparse and irregularly spaced, you will not be able to blindly toss them into these garden-variety tools and expect to get a reasonable solution. Simply plugging these points into `griddata` will garner very little credit, and plugging them into `Interpolation` won't work at all. Rather, you will have to figure out what the glacier surface looks like *between* and *beyond* the existing data points — using some sensible interpolation and extrapolation methods based on your code from previous problem sets in this course — and use that information to add some new (made-up, but hopefully consistent) points to the data set. Depending on the method you use, you may have to improvise at the edges to get surfaces that end smoothly — e.g., adding extra copies of the end points.

Plot your interpolated surface in 3D, experimenting with shading, point size, and other plotting parameters — contour versus perspective plot, various shading or coloring schemes, etc. — until it looks as good as possible. Turn in a printout of this plot, together with a one-paragraph discussion of your results and observations, including at least a few sentences on *how* and *why* you added interpolated points between those in the data set. (This description is worth half the credit for this problem.)

There is no single right answer here, and this is the hardest part of this assignment. Your creativity in working around this problem and manufacturing "good" points to add will determine how nice your surface looks. Hint: start by using some form of interpolation to add points between the existing ones. You may also use any other method you wish (e.g., neural nets); if you use a method that does not appear in the textbook, please include an explanation of how it works.

3. *[optional for 10 pts extra credit]* Repeat problem 2 for the bottom surface. Now try *superimposing* the two surfaces, removing the hidden part of the bottom surface if your plotting tool lets you do that easily. Note that the scales in the two data sets are quite different, so when you plot them on the same axes, some surface detail may be lost.

4. *[10 pts]* Problem 8(c) on page 198 of the textbook.

5. *[10 pts]* Fit these data to a power-law model by using linearization. Find the RMSE of the fit:

t	y
1	2
1.5	3.7
2.5	7.9
3	10.4

(This last problem may go away, depending on how far I get in lecture on 3/15. Please check the moodle for announcements if you missed that class.)