.,

CSS 455 Winter 2012 C. Jackels

Activity No. 11 answers February 27, 2012

Part I. Polynomial Interpolant of Runge function.

Download the file *activity11Polydemo.m* from the web site. Examine the file to see that it is using *polyfit and polyval* to fit and evaluate the interpolant.

1. For 11 data pts, what order of polynomial will give an exact interpolant? Is that the order being used within the script? How do you know?

A polynomial of order k has k+1 constants, so order (n-1) will fit n pts. For 11 data pts, we need a 10-th order polynomial. Yes: In the script, *polyfit* is called with *npts-1* as the parameter indicating the order.

2. Run the script and examine the plot. To try to improve the quality, add four points, to give a set of 15 evenly spaced interpolation points. This requires only changing line 17 to be *npts=15.* Did the interpolant improve or not? What was your basis for this conclusion?

It did not improve – but seemed to get worse. The large loops at the other segments, got much large in the case with 15 pts. They deviated much further from the actual curve.

3. Another idea is to add 4 new points, again using a total of 15, but adding them more strategically. To execute this attempt, comment out line 18 , which generates the 15 evenly spaced points, and uncomment line 19. Line 19 defines x as 11 evenly spaced points, with ones added at -4.5, -3.5, 3.5, and 4.5. Run it. What is your conclusion about the utility of using unevenly distributed points? *Do you think there tradeoffs involved?*

The "wiggles" are much smaller in magnitude now – the overall fit is more satisfactory. It is clearly more effective here to strategically place extra points rather than evenly distributing them. Tradeoff: some algorithms are designed to be efficient for evenly placed nodes – they are no longer useful here.

Part II. Cubic Piecewise Interpolants

Download the file *activity*11*splinedemo. m* from the web site. Examine the file to note that it is using the built-in *spline* and *pchip* functions to evaluate the splines. It then uses ppval to evaluate the interpolant. Be sure to notice how the outputs of the spline building functions are stored as a data structure in the structure names such as **PP.** These could be saved and used in other contexts.

1. Run the program, stepping it along with *Enter* keys at each pause. The curves are plotted in the order: Exact Runge curve; Not-a-knot cubic spline; Natural cubic spline; and piecewise cubic Hermite interpolating polynomial *(pchip).* What differences do you observe? Do they seem significant?
   All of these fits are quite good. There are subtle differences at some x-values, but they are minor and all fits are good.

2. Change the number of points to be interpolated from 11 to 12 at line 17. Now rerun the program and examine the plots.Now there is no point at the origin,.

   a. How does the Not-a-knot spline (first one run) differ from the exact curve?
      The center peak is too low and there are large deviations between the two curves at $|x| < 1$.

   b. How does the Natural spline differ from the Not-a-know? Differs only a little from (a).

c. How does the *pchip* interpolant differ from the splines above? The *pchip* curve has no maximum at the origin, but is horizontal between adjacent data points there. The deviations from the true curve are a bit larger than in (a) or (b), at least at smaller x-values.

d. Can you see evidence from the plot that the 2nd derivative of the *pchip* curve is discontinuous? Where? Yes – the shapes of the curve at x = + or – ½ or so do not look perfenctly smooth. While the 1st derivative is continuous, they suggest that the 2nd may not be.

e. Leaving *npts* set at 12 (line 17) comment out line 18 and uncomment line 19. This provides twelve slightly different points. Run the script, noting especially the behavior of the *pchip* curve.
Now the points are not symmetric about the origin.

   a. Which interpolant would be described as "smoother": *pchip* or *cubic spline?* What is your evidence for this?
   spline fits seem smoother. The *pchip* fit has a maximum and rather abrupt turn at the point at x= -0.5. Again – evidence of a discontinuous 2nd derivative.

   b. Which interpolant may introduce an extremum (maximum or minimum) at a location <u>not</u> dictated by the data? The spline fits have maxima at the origin, which are not at all indicated by the data. The underlying curve does have a maximum there, but the data do not suggest this. Splines can introduce unwarranted minima and maxima. This is a reason that pchip curves are sometimes preferred.