MATH 511

Winter Semester 2020

Homework 2: Due Friday, January 24, 2020

1. Determine the interpolating polynomial $p(x)$ discussed in Example 1.3 and verify that the evaluation $p'(\bar{x})$ gives equation (1.11). Once the coefficients of the finite difference formula (1.11) has been determined , obtain an analytical formula for the truncation error by hand computation. Verify the validity of your truncation error formula using the `fdstencilRatMain.m` code.

2. (a) Use the method of undetermined coefficients to set up the 5×5 Vandermonde system that would determine a fourth-order accurate finite difference approximation to $u''(x)$ based on 5 equally spaced points,

$$u''(x) = c_{-2}u(x-2h) + c_{-1}u(x-h) + c_0u(x) + c_1u(x+h) + c_2u(x+2h) + O(h^4).$$

   (b) Compute the coefficients using the matlab code `fdstencil.m` available from the website, and check that they satisfy the system you determined in part (a). Also, obtain an analytical formula for the truncation error by hand computation using the previous coefficients. Verify your truncation error formula using the `fdstencilRatMain.m` code.

   (c) Test this finite difference formula to approximate $u''(1)$ for $u(x) = \sin(2x)$ with values of h from the array `hvals = logspace(-1, -4, 13)`. Make a table of the error vs. h for several values of h and compare against the predicted error from the leading term of the expression printed by `fdstencil`. You may want to look at the m-file `chap1example1.m` for guidance on the definition of `hvals` and on how to make such a table.

   Also produce a log-log plot of the absolute value of the error vs. h. You should observe the predicted accuracy for moderate small values of h. For much smaller values, numerical cancellation in computing the linear combination of u values impacts the accuracy observed.

   (d) (3 points Bonus) Compute the order p of the error ($O(h^p)$) between consecutive computations as you refine h. Include that column in your table.

3. Analyze the tables in Figs 3.3 and 3.4 of Hornbeck (1975). They are in `https://www.math.byu.edu/ vianey/Math511/ClassNotes/Chap1LevequeM.pdf`.
Then, establish a relationship (a formula) between the number of grid points n, the order of the derivative k, and the order of the truncation error p; for the following:

   (a) Centered difference of $O(h^2)$

   (b) Centered difference of $O(h^4)$

   (c) One-sided difference of $O(h^2)$

   What is the difference between even and odd derivatives for the centered differences. Is this difference also present for the one-side formulas?

4. (a) Derivate the finite difference formulas (3-43) and (3.44) in Table 3-1 of Tanehill-Anderson-Pletcher, with their corresponding truncation errors, using `fdstencil`. These formulas are found in
`https://www.math.byu.edu/ vianey/Math511/ClassNotes/Chap1LevequeM.pdf`.

   (b) Using `fdstencil` obtain a centered finite difference formula for $u'(x)$ and its truncation error based on the points: x_{i-4}, x_{i-3}, x_{i-2}, x_{i-1}, x_i. Compare with (3.44). What is your conclusion?

   (c) Using `fdstencil` obtain a centered finite difference formula for $u''(x)$ and its truncation error based on 7 points.

5. (a) **Subroutine or function** `[c,err0,err1] = fdstencilnonuniform(k,xb,xpts)`.
Use the book matlab codes `fdcoeffF.m` and `fdstencil.m` to write the subroutine `fdstencilnonuniform(k,xb,xpts)` that computes a vector **c** of the coefficients of the finite difference and produces as an output the finite difference formula for $u^{(k)}(xb)$ corresponding to the grid points $x_1,\ x_2, \dots x_n$ that form the vector `xpts` (not necessarily uniform). The point `xb` does not have to be one of the grid points.

   Your subroutine `fdstencilnonuniform(k,xb,xpts)` should also compute the coefficients of the first two leading order terms of the error (err0 and err1) and should include in his output these error terms, *i.e.,* `[c,err0,err1] = fdstencilnonuniform(k,xb,xpts)`. The book code `fdstencil.m` can be adapted for this purpose.

   (b) Find a finite difference formula of $u''(0)$ (it means `k`=2 and `xb`= 0) for `xpts`= $[-1;\, 0;\, 1]$. Comment on your result.

   (c) **Subroutine or function** `FDnonuniform(f,fkp,fnp,fn1p,k,xb,xpts)`.
Write another subroutine `FDnonuniform(f,fkp,fnp,fn1p,k,xb,xpts)` that has as additional parameters `f`, `fkp`, `fnp`, and `fn1p` which represent the function `f`, its k-derivative `fkp`, to be approximated, its n-derivative, `fnp` , and n+1-derivative, `fn1p`, (needed to estimate the error), respectively.

   This subroutine should print the following results:
The Approximation of $u^{(k)}(xb)$,
the *Exact Value* of $u^{(k)}(xb)$,
the *Actual Error* in the approximation, and
the *Estimated Error*
The last one computed from the leading order terms of the truncation error. Try to produce an output similar to the attached `OutputFDnonuniform.pdf`

   (d) **Application of the subroutine** `FDnonuniform(f,fkp,fnp,fn1p,k,xb,xpts)`.
For the function $f(x) = e^{\frac{x}{3}}$, approximate its second derivative at $\bar{x} = 1/2$ by using the following grid points:

      i. $xpts = [xb - 1e - 1;\ xb;\ xb + 1e - 1]$
      ii. $xpts = [xb - 1e - 1;\ xb - 1e - 3;\ xb; xb + 1e - 2]$
      iii. $xpts = [-3;\ 0;\ 1/2;\ 1;\ 4]$
      iv. $xpts = [-3;\ 0;\ 1/3;\ 1; 4]$

   Report the errors as described in the previous item. Write a paragraph of at least four lines making comments about your results.

   Hint: To call the subroutine you may use:
`FDnonuniform('f','fkp','fnp','fn1p',k,xb,xpts)`, *where f, fkp, fnp, fn1p are external functions. Also, for computation of $f(x)$, use $feval(f, x)$ in the subroutine.*