MATH 3340 - Scientific Computing Final Project

Due: Firday, 12/11/2020, 05:00 PM

Submit online; make sure to sign the clause. All your work, including the signed page, should be together as one PDF file.

Instruction

1. Go to `https://www.overleaf.com` and sign in (required).
2. Open template, click *Menu* (up left corner), then *Copy Project*.
3. Go to `LaTeX/meta.tex` (the file `meta.tex` under the folder `LaTeX`) to change the section and your name, e.g.,
   - change author to `\author{Carl F. Gauss}`
4. You need to write function/script files, store results to output/plot files. Here are suggested names for function files, script files, output files, and plot files:

Problem	Function File	Script File	Output File	Plot File
1	`cubic_spline.m`	`final_p1.m`	`final_p1.txt`	`final_p1a.pdf`
1				& `final_p1b.pdf`
2	`polyfit_recursive.m`	`final_p2.m`	`final_p2.txt`	
3	`golden_section.m`	`final_p3.m`	`final_p3.txt`	`final_p3.pdf`
3	& `successive_parabolic.m`			

   Once finished, you need to upload these files to the folder `src` on Overleaf. If you have different filenames, please update the filenames in `\lstinputlisting{../src/your_script_name.m}` accordingly. You can code in the provided files in final.zip, and use the MATLAB script `save_results.m` to generate the output files and store the graphs to `.pdf` files automatically (the script filenames should be exactly same as listed above).
5. Recompile, and download the generated `.pdf` file.
6. **Important**: Enter your name and the date in the above boxes *before* you submit it on WyoCourses.

Problem 1. Consider the data in the following table:

k	0	1	2	3
x_k	0.0	1.761062	3.522123	5.283185
y_k	1.0	−0.1891196	−0.9284676	0.5403023

The values of y in this table have been obtained as $y_k = \cos(x_k)$. Your goal will be to create two different spline interpolants using these data points, and compare them with the original function.

(a) Compute the usual spline interpolant ${}^1S(x)$ that uses the natural boundary conditions. Plot this interpolant versus the original function $\cos(x)$ using 100 data points equally spaced between x_0 and x_3; make sure to also indicate the four data points on the plot using a special marker (you may use $*$ or $\circ$ for example).

(b) You will probably agree that the comparison at point (a) above doesn't look very good. This is due to the fact that the natural boundary conditions do not match the behavior of the $\cos(x)$ function. Your task for this point is to modify your code to account for the correct second derivatives at the end points. In other words, create a new interpolant ${}^2S(x)$ that satisfies the following conditions:

$$ {}^2S''(x_0) = -\cos(x_0); \quad {}^2S''(x_3) = -\cos(x_3). $$

You can do this by modifying the system of equations for the spline coefficients accordingly. Plot again the newly-obtained interpolant versus the original function in the same manner as above. Turn in your codes together with the two plots.

Problem 2. Consider again this table of data values for a certain function $y = f(x)$:

k	0	1	2	3	4
x_k	0.1	0.3	0.5	0.7	0.9
y_k	1.1	1.25	1.18	1.32	1.36

Using the recursive formulation based on the polynomials $Q_{ij}(x)$ described in section 6.2.3 starting on page 86 in the notes, create the full table of possible interpolation values of the function at the point $x = 0.35$. You should write a computer code for this purpose, which must print the values with the default MATLAB precision. Turn in the code and the output (i.e. the tabulated values).

Problem 3. Use both successive parabolic interpolation and the golden section search method to find the minimum of the function $f(x) = |x^2 - 2| + |2x + 3|$ on the interval $[-4, 0]$ with a tolerance of 10^{-8} and an identical accuracy. You are again requested to do this with MATLAB codes which you must show, together with the output; recall that golden search method code was discussed in detail in class.