Numerical Methods (2021–22) Assignment 1

Due: 2021-11-05

Your submission must be a `.pdf` file containing a written account of your solutions to all of the questions below, together with any `.py` files created for use in Python. All figures requested must be included in the `.pdf` file and annotated with axis labels and figure captions. The source code of all Python functions specifically requested must be included in the `.pdf` file. All files must be zipped together in a single `.zip` file and uploaded to Moodle by the submission deadline. Further details on the submission format are available on Moodle. Please, **do not deviate from the submission instructions.**

Question 1

Given the function $f(x) = \cos(x)$ with $x \in [-\pi/2, \pi/2]$,

1. write a function that takes the limits of an interval and the order n of the polynomial interpolation $p_n(x)$, and returns the data points x_i that can be used for the Chebyshev interpolation. Test this with an example input (your choice).

   (Marks: 7)

2. write a function that takes a function f, order n, array (of any length) z and the array x with the data points for the polynomial interpolation, and returns the values $p_n(z)$ of the polynomial interpolation. You can use any of the three methods for finding the polynomial interpolation from the lecture notes.

   (Marks: 22)

3. Use your interpolation function on the given function and the given interval using second order interpolation and both, Chebyshev economisation and equidistant data points. Plot the two interpolants in the same plot using different colours and linestyles together with the data points for both methods using markers. Add axis labels and a legend.

   (Marks: 12)

Question 2

1. Given the function $f(x) = e^x + x^2$, use the composite trapezium rule to compute the integral $\int_a^b f(x)\, \mathrm{d}x$ using n equally spaced sub-intervals. Write this into a function that takes f, a, b and n as inputs and returns the integral.

   (Marks: 13)

2. Using the interval $[a, b] = [0, 2]$ and the values $n = 2^p$ with $p = 0, 1, ...20$, compute the integral and plot the result in dependence of n using a semi-logarithmic (in n) plot. In a different figure, plot the absolute difference between your numerical result and the exact value of the integral against n using a log-log plot. As always, add axis labels. Comment on your second figure.

   (Marks: 11)

Question 3

Given the function $f(x) = e^x - x^2$,

1. use the Newton-Raphson method to find its approximate unique real root.

   (Marks: 11)

2. with your Newton-Raphson solution, find the minimum number of iterations needed to find a solution of $f(x) = 0$ to a tolerance of $\tau = 10^{-5}$ with starting points $x_0 = -10, -9, ...4, 5$. Plot the minimum number of iterations n_{min} against x_0.

   (Marks: 11)

3. use the interval bisection method to find its approximate unique real root.

   (Marks: 13)

Notes

1. Use of inbuilt Python functions or functions imported from libraries that would solve significant parts of the questions, except the ones explicitly mentioned, is prohibited.

2. All plots must contain labels for the axis.

3. The programming component is a significant part of this assignment. All Python codes should be fully commented and included in your submission both as `.py` files as well as printed out in your final `.pdf` report along with figures, tables and discussion of your solutions.

4. You can save a figure after it is plotted using the command `plt.savefig('my_plot.eps')`. If you use the `.eps` ending instead of `.png` or `.jpg` then your figure file will be in vector graphics format, which is of much higher quality than a pixel based image that can easily look blurry.

END