MACM 316 – Computing Assignment 5

- Read the *Guidelines for Assignments* first.
- Write a one-page PDF report summarizing your finding. This report must also include all your figures.
- Submit the one-page PDF report using Crowdmark and add your Matlab code to it (as extra pages). You must use the link sent by Crowdmark to your SFU email address. Follow the instructions on the Crowdmark to upload your file. If the report is hand-written then use the CamScanner app with your cellphone to scan the report and every page of the code. You will lose marks for poor quality pictures.
- You must acknowledge any collaborations/assistance from fellow students, discussion forums, TAs, instructors, etc.

Convergence of polynomial interpolation

As we have seen in Part 4.II of the lecture notes, when Chebyshev nodes are used the polynomial interpolant $P(x)$ of a smooth function $f(x)$ appears to converge with linear order; see the demo *nodedistribution.m* and the figure on p.10 of the lecture notes. In other words, the error satisfies

$$\mathrm{err}_n = \max_{-1 \le x \le 1} |f(x) - P(x)| \le C\rho^{-n}, \tag{$\star$}$$

for constants $C > 0$ and $\rho > 1$. Your objective in this assignment is to investigate the behaviour of the constant ρ. You will do this by considering the function

$$f(x) = \frac{1}{(x-a)^2 + b^2},$$

where a and b are nonzero real numbers. Your goal is to understand how the number ρ changes as a and b are varied.

Download the file *CA5single_ab.m* and run it. For fixed values of a and b, this computes the error versus n and stores it in an array `err`. The range of n is set so that the minimum error is no smaller than a specified tolerance `tol`, which is set to 10^{-10}. You do not need to change `tol`.

This file also plots the error versus n. See the figure on the next page. Notice that the order of convergence appears to be linear, in accordance with ($\star$).

(a) Using the `polyfit` function in Matlab, find a linear fit of this line. Add the corresponding line of fit to the figure generated by *CA5single_ab.m*, and include it in your report. Make sure that all axes, fonts, plotmarkers, etc are readable.

(b) Using the results of `polyfit` determine the corresponding estimate for the value of ρ in ($\star$) for the given values $(a, b) = (0.2, 0.2)$. Explain how you found this estimate. Report the corresponding values of ρ for $(a, b) = (0.5, 0.5)$ and $(a, b) = (0.7, 1)$.

(c) Having done this, briefly explain why the code uses a range of n defined by the tolerance `tol` and not some fixed maximum value of n, as was done in the demo *nodedistribution.m*.

(d) Your next goal is to produce a contour plot of ρ with values of a on the x-axis and values of b on the y-axis. You should use the `contour` function in Matlab to do this. Choose a reasonable

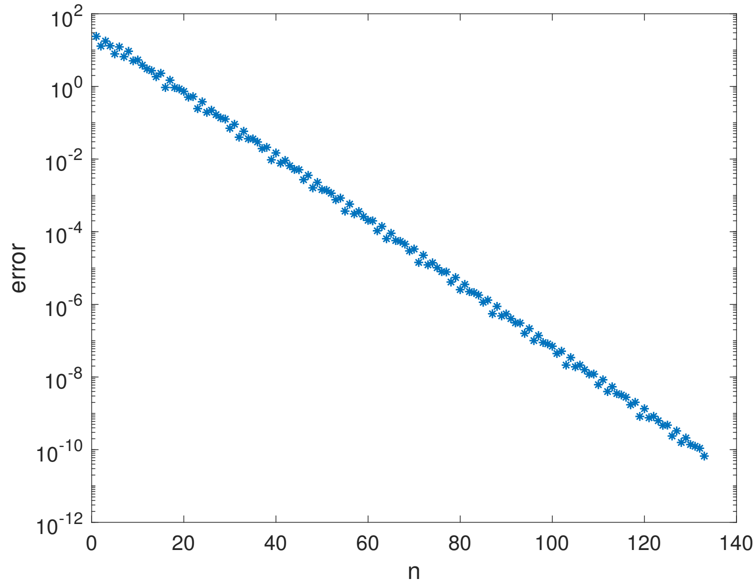

Figure 1: The error ($\star$) versus n for the values $(a, b) = (0.2, 0.2)$.

range of nonzero values of a and b, with a reasonable stepsize, and write a code that computes the corresponding values of ρ and then produces the appropriate contour. Include your contour plot in your report. Make sure to make the axes labels correct, and all fonts, plotmarkers, etc readable.

Time-saving hint: you may assume that ρ depends on the absolute values of a and b only. In other words, you only need to run the code for positive values of a and b. However, your contour plot should show both positive and negative values.

Other hint: I suggest you use a minimum value at least 0.05 and a maximum value of at most 1.25 for both a and b.

(e) Discuss the qualitative behaviour of ρ as a and b vary. For instance, when is ρ increasing/decreasing, what do the level curves of ρ look like, and so forth.