Computational Methods and Modelling

Antonio Attili & Edward McCarthy

antonio.attili@ed.ac.uk
ed.mccarthy@ed.ac.uk

School of Engineering
University of Edinburgh
United Kingdom

Tutorial 3
Error in root finding methods, Polynomial Deflation.

THE UNIVERSITY *of* EDINBURGH

Exercise 1: Analyse the convergence of root finding methods

- Consider the quadratic equation $f(x) = x^2 + 4x - 12 = 0$
- Write a python code to find one of the two roots with the Newton-Raphson method for different numbers of iterations N, starting from $N = 1$. Hint: use a for loop inside which you call the Newton-Raphson python function several times.
- Plot the solution x_{root} and a measure of the error, e.g., $f(x_{root})$, for the different values of N.
- Repeat the two steps above for the secant method.
- Comment your results: how fast does the error decreases with N? Which method is faster?
- Check your conclusions about the error, repeating the same analysis for $f(x) = sin(x) * e^{x^{0.1}}$ (to find the smallest positive, non-zero, root).

Exercise 2: Polynomial Deflation

- Suppose we determine a single root of an nth–order polynomial. If you repeat your root location procedure, you might find the same root. Therefore, it would be nice to remove the found root before proceeding. This removal process is referred to as **polynomial deflation**.
- Consider the polynomial $f(x) = x^2 + 2x - 24$. One of its root is $t = 4$, so we can divide it by $(x - 4)$. Since 4 is a root of $f(x)$, this division does not have a reminder r
- Consider the more general case of the division $q(x) = f(x)/(x - t)$, where t is not necessarily a root. Write python code to perform this division. Assume that $f(x)$ is a polynomial of degree n, so the quotient $q(x)$ will have degree $n - 1$. Your code should output the coefficients of the polynomial $q(x)$ and the value of the reminder r.
- The synthetic division algorithm is a good approach for this task. Consider the following pseudocode for the synthetic division, see section 7.2.2 of textbook (Chapra and Canale):

```
r = a(n)
a(n) = 0
DOFOR i = n−1, 0, −1
  s = a(i)
  a(i) = r
  r = s + r * t
END DO
```