MA3227 Numerical Analysis II

Tutorial 2: Nonlinear Equations

Bisection and Newton's methods in practice

Use the bisection and Newton methods implemented in the `Roots` package to determine the unique real root of the function

$$f(x) = 4 - 3x + 2x^2 - x^3.$$

To do so, you will have to determine an initial bracketing interval $[a_0, b_0]$ and an initial guess x_0. These quantities can be easily read off from a plot of $f(x)$ over a suitable range of x-values.

Newton's method for computing complex square roots

Write a Julia function `square_root(w)` which uses Newton's method to compute complex square roots

$$x + iy = \sqrt{u + iv} \qquad \Longleftrightarrow \qquad (x + iy)^2 = u + iv.$$

To do so, you must translate the equation on the right into a 2×2 system of nonlinear equations $f(x) = 0$, manually compute the Jacobian $\nabla f(x)$ and then implement the Newton iteration

$$x_{k+1} = x_k - \nabla f(x_k)^{-1}\, f(x_k)$$

in Julia (you can use `df(xk) \ f(xk)` to solve the linear system).

Can you observe the quadratic convergence of Newton's method?

Algorithmic details:

- Use $(x_0, y_0) = (u, v)$ as initial guess.
- Terminate the iteration once $\big|(x + \imath y)^2 - (u + \imath v)\big| \leq 10\,\texttt{eps()}\,|x + \imath y|$.
- Throw an error if the iteration does not terminate after 20 steps.
  You can do so using the command `error("[error message]")`.

Julia hints:

- The imaginary unit is called `im`. (Example: `im^2 -> -1+0im`)
- You can obtain the real and imaginary parts of a complex number `z` using `real(z)` and `imag(z)`, respectively. (Example: `real(1+2im) -> 1`, `imag(1+2im) -> 2`)