Root Finding via Newton-Raphson

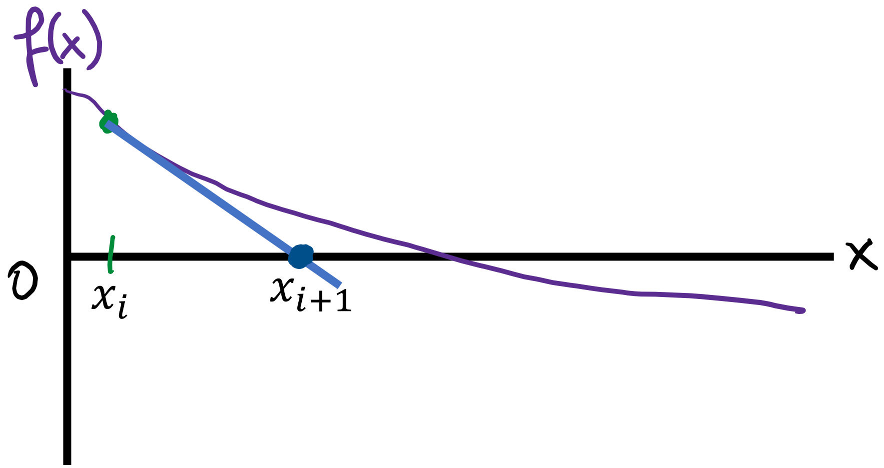
A Quick Review

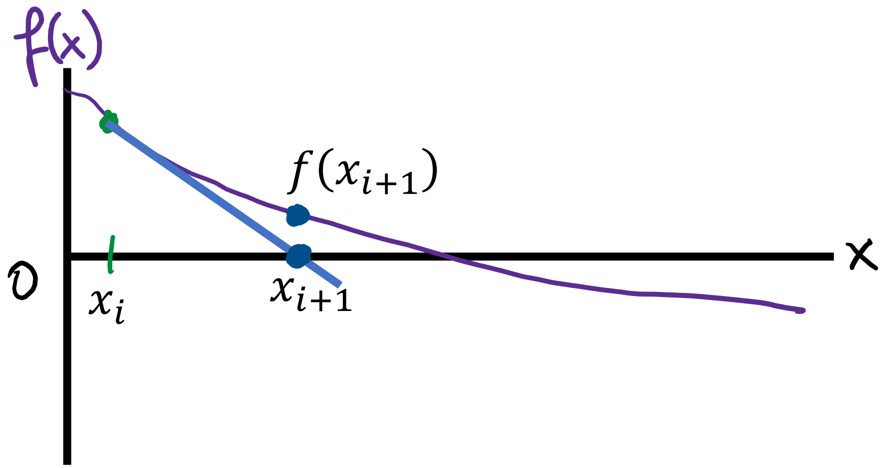
Newton-Raphson Method: Overview

- Newton-Raphson/Newton’s Method: another root finding technique
- Premise: combine an initial starting point with the derivative of the function to iteratively trace the root
  - WARNING: May diverge!!! (unable to locate root)
- Called an *open method* because the user must supply the algorithm with a singular starting point
  - Contrast: Bisection is a *bracketing method* because it needs two initial guesses which bracket the root

Brief Algorithm Overview

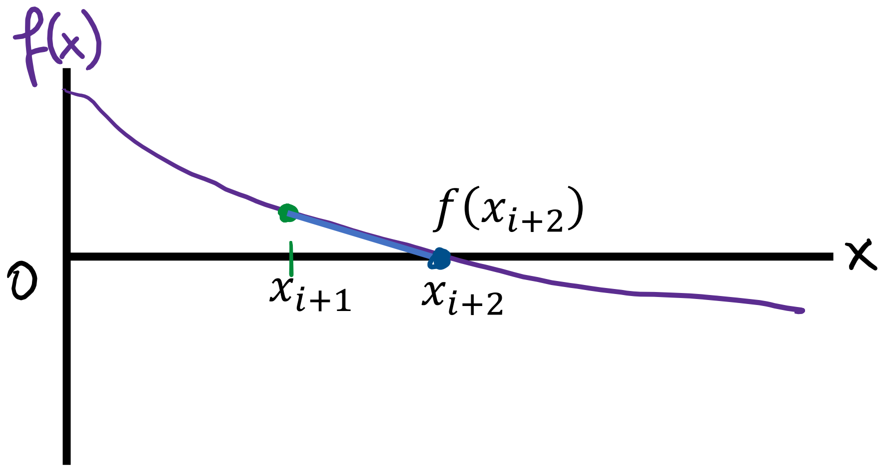
- Input initial guess x_i and compute $f'(x)$
- From the starting point x_i, extend the tangent $f'(x_i)$ until it hits the x-axis
- This new location is the new estimate of the root, x_{i+1}
- Compute $|f(x_{i+1})|$ and iterate if the stopping criterion is not met





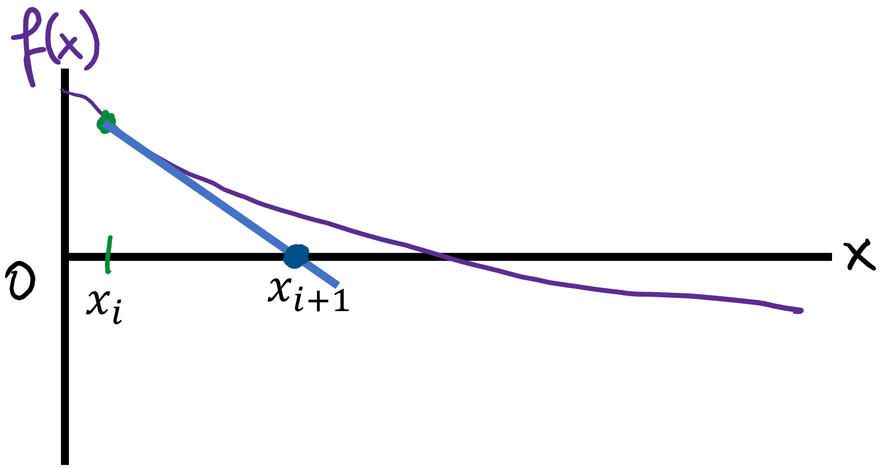
- Start with an initial guess x_i
- Draw the tangent $(f'(x))$ to the curve at that point and “follow” it until it hits the x-axis
- This point is x_{i+1} (new root estimate)

- Evaluate $|f(x_{i+1})|$. Assume it doesn’t meet our stopping criterion (e_s) → must iterate.



- We draw the tangent at x_{i+1} and extend it until it crosses the x-axis
- This point is x_{i+2} (new root estimate)
- Assume $|f(x_{i+2})| \leq e_s$ $\rightarrow$ we've located the root!

Mathematics Behind Newton-Raphson



- Definition of slope:

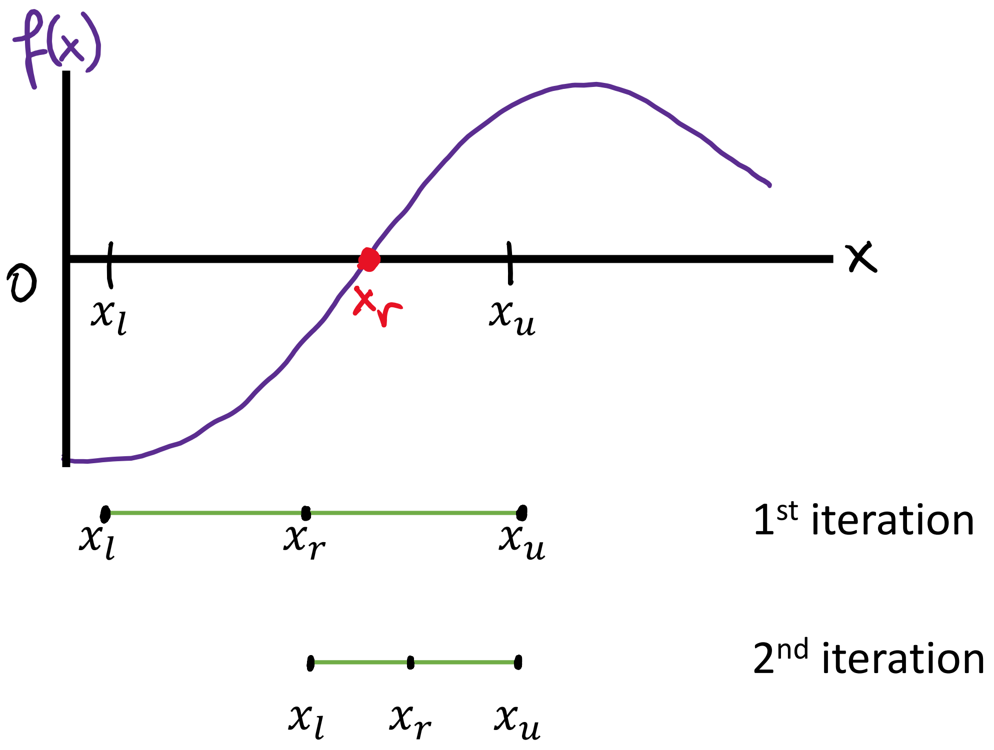
$$f'(x) = \frac{f(x_i) - \cancelto{0}{f(x_{i+1})}}{x_i - x_{i+1}}$$

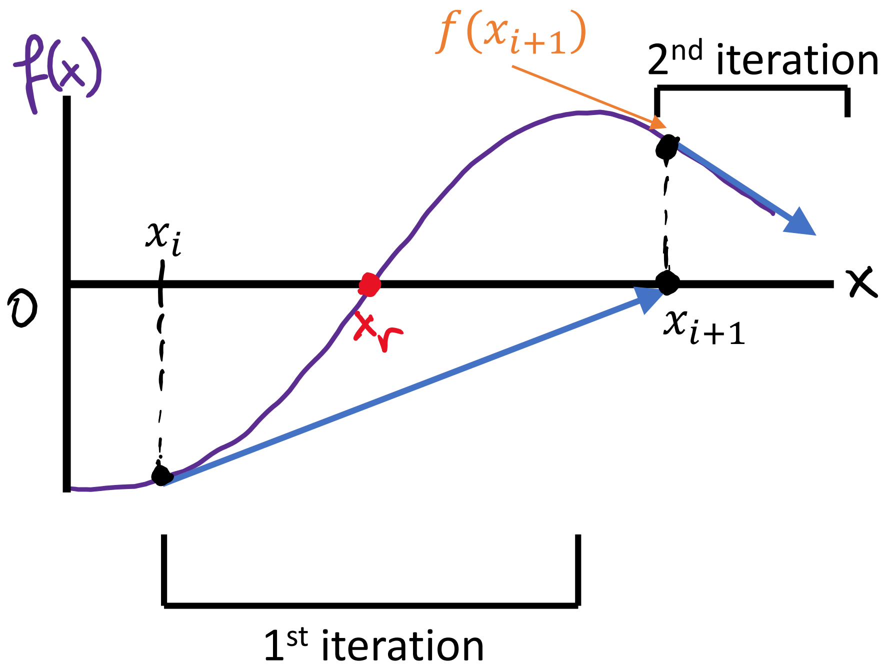
- Rearranged:

$$\boxed{x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}}$$

Implement this in MATLAB!

Pitfalls of Newton-Raphson





Bisection always converges (given a valid initial bracket)!

NR may diverge!

Pitfalls of Newton-Raphson

- Newton-Raphson equation:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

- What happens if $f'(x_i) = 0$? Division-by-zero error!

Bisection vs. Newton-Raphson

Consider Picking Bisection If…

- You MUST converge on x_r
- You don't care about convergence speed
- You don't know the behavior of $f'(x)$
- $f'(x)$ is hard to compute

Consider Picking NR If…

- You accept the possibility of divergence
- You want to quickly find x_r
- You know $f'(x)$ won't return a division-by-0 error
- $f'(x)$ is relatively easy to compute

Food For Thought 

- What *graphically* happens when you get a division-by-zero error?
- Can you generate a function $f(x)$ which causes bisection to converge <u>faster</u> than NR?
- Can you derive the function $f(x)$ which causes NR to cycle infinitely around a point?