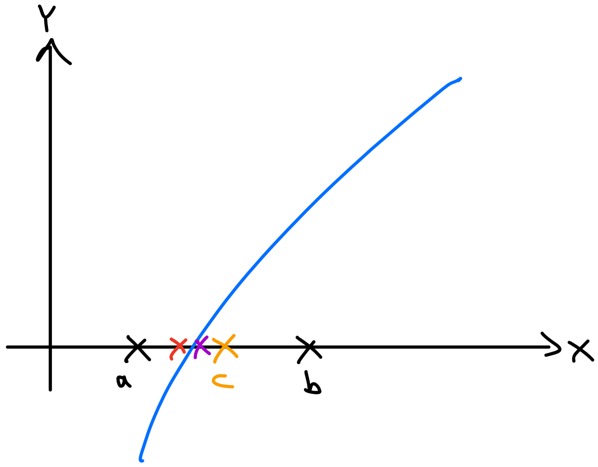
Root Finding Algorithms:

Bisection method:



Steps of the method:
- Choose $a<b$ s.t. $f(a)f(b)<0$ (If $f(a)f(b)=0$ $\Rightarrow$ done, root is either a or b)
- set $c = \frac{a+b}{2}$
- If $f(a)f(c)<0 \rightarrow$ root in $[a,c]$

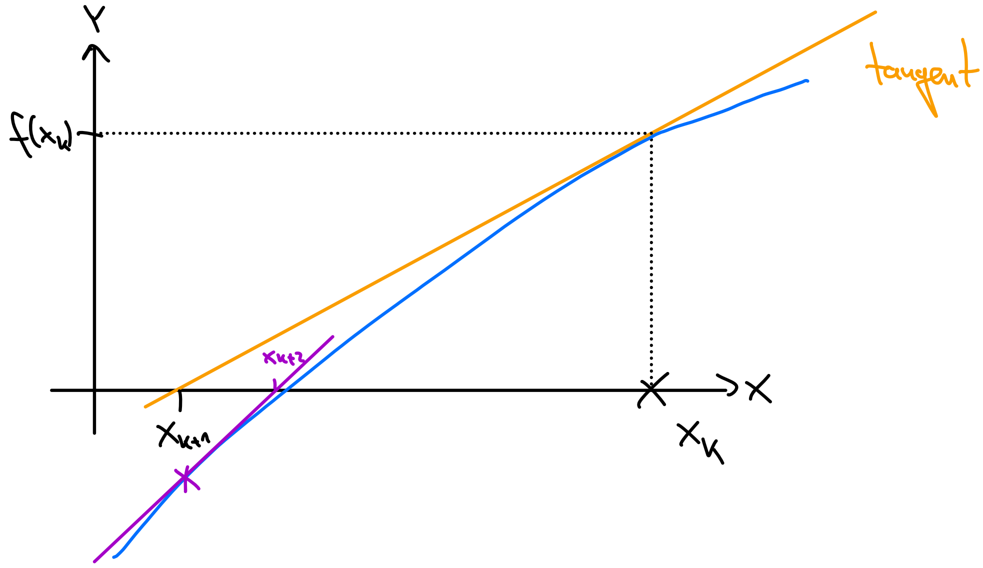
  If $f(c)f(b)<0 \rightarrow$ root in $[c,b]$

  $\Rightarrow$ repeat with either $[a,c]$ or $[c,b]$

Advantages:
- only continuity of f necessary

Disadvantages:
- If $f(x)\geq 0$ or $f(x)\leq 0$ in a neighborhood around a root, method does not work
- If the error after $n+1$ steps is ε_{n+1}, then $\varepsilon_{n+1} = \frac{1}{2}\varepsilon_n$, so here the rate of convergence r from $\varepsilon_{n+1} = c\,\varepsilon_n^r$ is 1, so convergence is linear $\Rightarrow$ rather slow

Newton's Method (Newton-Raphson-Method)

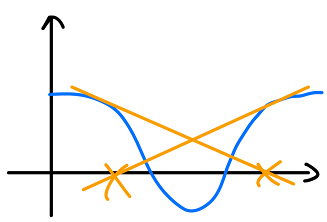


- Start: choose initial data x_k
- Then slope of orange line $= f'(x_k) = \dfrac{f(x_k)}{x_k - x_{k+1}}$

$$\Rightarrow \quad x_k - x_{k+1} = \frac{f(x_k)}{f'(x_k)}$$

$$\Rightarrow \quad \boxed{x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}} \quad \Rightarrow \text{iteration}$$

- Advantages: • fast, see later
- Disadvantages: • fct. needs to be differentiable
  - problem if $f'(x_k) = 0$
  - either need explicit expression or numerical evaluation of derivative
  - certain initial data might not work, e.g., could get stuck in a loop, or initial data too far away from root



What is rate of convergence here?

Use Taylor expansion around x_k:

$$f(z) = f(x_k) + f'(x_k)(z-x_k) + \frac{f''(x_k)}{2}(z-x_k)^2 + \underbrace{\overbrace{\mathcal{O}\left((z-x_k)^3\right)}^{=:R}}_{\text{"rest term" or "remainder term"}}$$

Now: let $z = \text{root}$, i.e., $f(z)=0$, and use iteration:

$$\Rightarrow 0 = f(x_k) + f'(x_k)(z - \underbrace{x_k}_{x_k = x_{k+1} + \frac{f(x_k)}{f'(x_k)}}) + \frac{f''(x_k)}{2}(z-x_k)^2 + R$$

$$\Rightarrow 0 = \cancel{f(x_k)} + f'(x_k)(z - x_{k+1}) - \cancel{f'(x_k)}\frac{\cancel{f(x_k)}}{\cancel{f'(x_k)}} + \frac{f''(x_k)}{2}(z-x_k)^2 + R$$

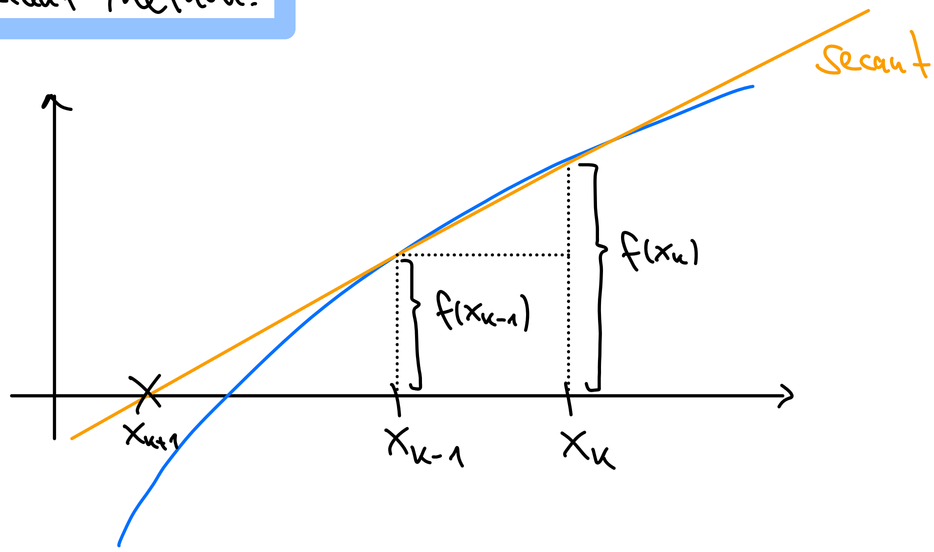
$$\Rightarrow \underbrace{|z - x_{k+1}|}_{\varepsilon_{k+1}} = \left|\frac{f''(x_k)}{2f'(x_k)}\right| \underbrace{|z-x_k|}_{\varepsilon_k}^{2} + \text{Rest}$$

rate of convergence is $r = 2$, i.e., we have quadratic speed/rate of convergence

$\Rightarrow$ Advantage: quadratic rate of convergence if f'' continuous

Disadvantage:
- speed of convergence might be slower if f'' not continuous
- problem if $f'(x_k) \to 0$

$\boxed{\text{Secant Method:}}$



$$\frac{f(x_k)}{x_k - x_{k+1}} = \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$$

Thales theorem (intercept thm.)
("Strahlensatz")

$$x_k - x_{k+1} = \frac{f(x_k)}{f(x_k) - f(x_{k-1})}(x_k - x_{k-1})$$

$$\Rightarrow \boxed{x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}}$$

Compared to Newton's method:

- Advantage: No derivative needed here.

Rate of convergence is the golden ratio ≈ 1.62, which is very good
(much faster than bisection, only a bit slower than Newton).

In python there is a bilt-in fct. $\boxed{\text{brentq}}$:

- combines robustness of bisection with speed of secant metho
- works for all continuous fct.s ($\rightarrow$ look up documentation)