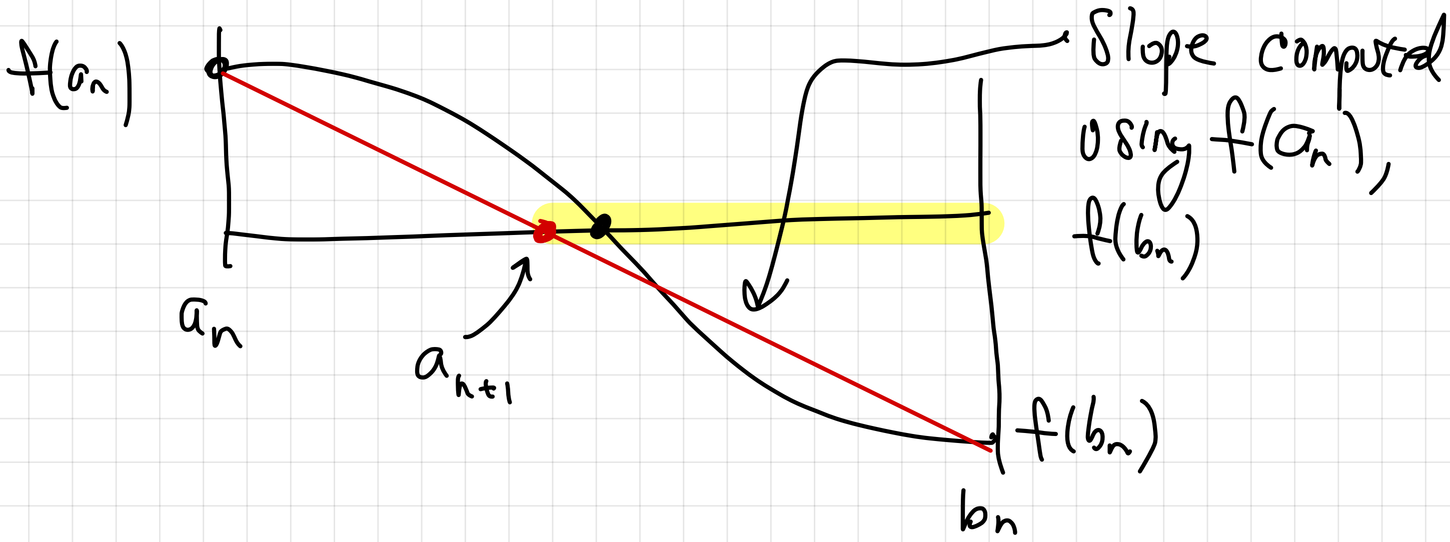
Newton's Method
and
Secant Method

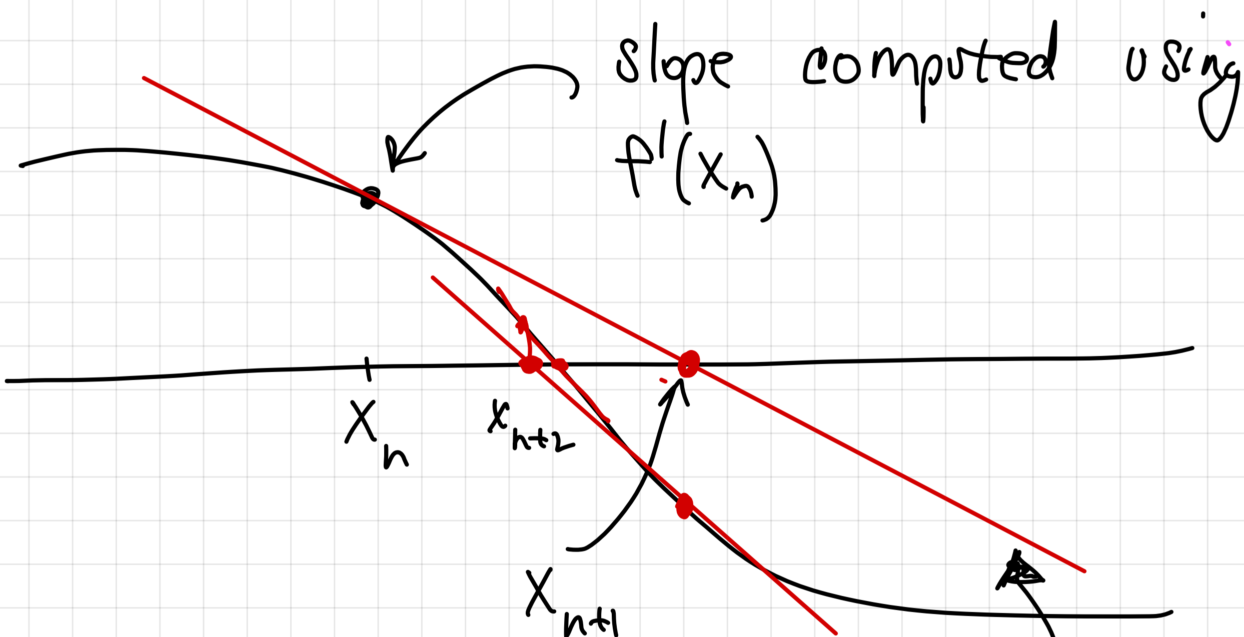
Newton's method

Method of False Position:



Use slope to construct new iterate c.

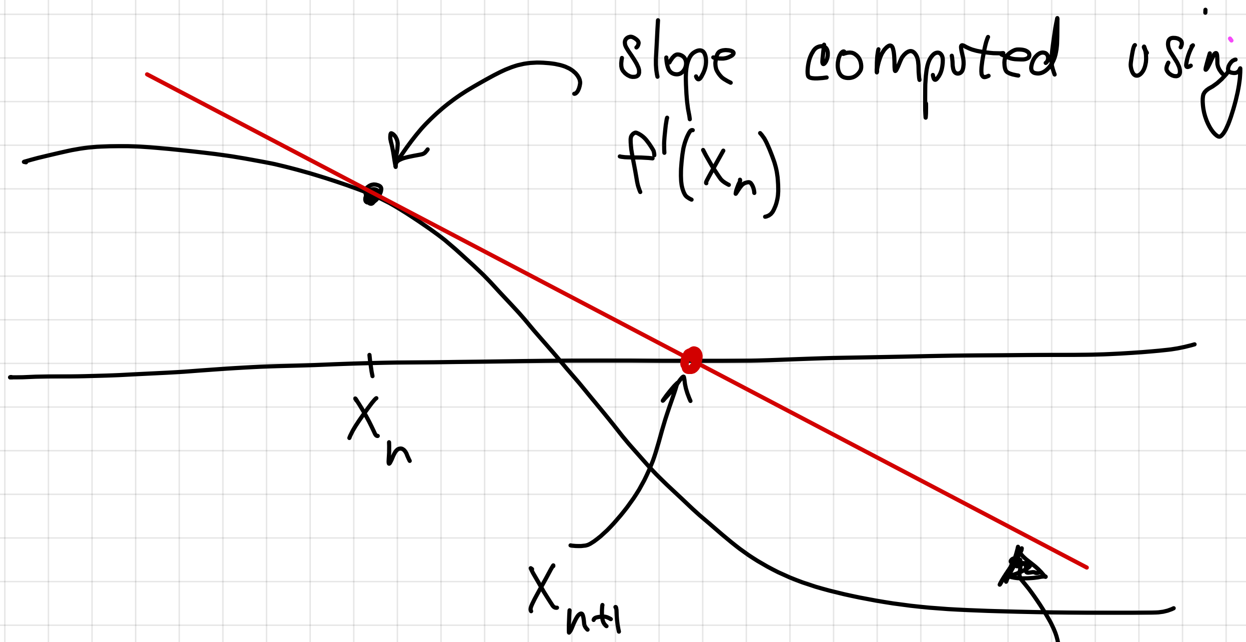
Newton's method uses a similar idea:



line: $y = f'(x_n)(x-x_n) + f(x_n)$

"point slope"

Newton's Method



line: $y = f'(x_n)(x - x_n) + f(x_n)$

Solve for point $(x_{n+1}, 0)$:

$$0 = f'(x_n)(x_{n+1} - x_n) + f(x_n)$$

$$\Rightarrow x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad \Big\} \text{ Newton step.}$$

Algorithm needs a good starting guess, rather than a bracketing interval.

Algorithm: Newton's method.

```
function [xroot, en] = newton(f,fp,x0,tol,kmax)

xk = x0;
for k = 1:kmax
    xkp1 = xk - f(xk)/fp(xk);
    en(k) = abs(xkp1 - xk);
    fprintf('%5d %20.16f %12.4e\n',k,xkp1,en(k));
    if (en(k) < tol)
        fprintf('Tolerance achieved\n');
        xroot = xk;
        break;
    end
    xk = xkp1;
end
xroot = xk;



end
```

Newton Step

$en(k) \approx |x_k - \bar{x}|$

- Requires call to function and derivative.
- Stopping criteria is simple.

Newton's Method – Example

$$f(x) = \frac{1}{3}x^3 - x^2 + \frac{4}{3}\beta \quad , \quad \beta = 0.1$$

$$f'(x) = x^2 - 2x$$

k	x_k	$\lvert x_{k+1} - x_k \rvert$
1	0.466666666666666	5.3333e-01
2	0.3959972394755003	7.0669e-02
3	0.3916186407833392	4.3786e-03
4	0.3916002116462435	1.8429e-05
5	0.3916002113181835	3.2806e-10
6	0.3916002113181834	5.5511e-17

Tolerance achieved

Newton :   0.3916002113181835

approximate doubling of digits of accuracy.

- Digits of accuracy approximately double with each iteration
- Suggest Quadratic convergence.

Newton's Method - Order of Convergence

① Write down iteration:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

② Derive iteration for the error $e_n = x_n - \bar{x}$

$$e_{n+1} = e_n - \frac{f(x_n)}{f'(x_n)}$$

③ Expand $f(x_n)$, $f'(x_n)$ about $\bar{x}$:

$$\boxed{\begin{aligned} f(x_n) &\approx f'(\bar{x})e_n + \tfrac{1}{2}f''(\bar{x})e_n^2 \\ f'(x_n) &\approx f'(\bar{x}) + f''(\bar{x})e_n \end{aligned}}$$

④ Simplify expression for e_{n+1}:

$$e_{n+1} = e_n - \frac{f'(x_n)e_n + \frac{1}{2}f''(\bar{x})e_n^2}{f'(\bar{x}) + f''(\bar{x})\cancel{e_n}^{\approx 0}}$$

$e_n \approx 10^{-3}$

$e_{n+1} \approx e_n^2 \approx 10^{-6}$

$$e_{n+1} = \frac{f''(\bar{x})}{2f'(\bar{x})} e_n^2$$

$\alpha = 2$.

Newton's Method is quadratically convergent

Newton's Method: $f'(\bar{x}) = 0$

What happen's if $f'(\bar{x}) = 0$?

3

$$f(x_n) \approx \underbrace{\cancel{f'(\bar{x})}}_{=0} e_n + \frac{1}{2} f''(\bar{x}) e_n^2$$
$$= \frac{1}{2} f''(\bar{x}) e_n^2$$

$$f'(x_n) \approx \underbrace{\cancel{f'(\bar{x})}}_{=0} + f''(\bar{x}) e_n$$
$$= f''(\bar{x}) e_n$$

4 Simplify expression for e_{n+1}:

$$e_{n+1} = e_n - \frac{\underbrace{\cancel{f'(x_n)} e_n}_{=0} + \frac{1}{2} f''(\bar{x}) e_n^2}{\cancel{f'(\bar{x})} + f''(\bar{x}) e_n}$$

$$= e_n - \frac{1}{2} e_n = \frac{1}{2} e_n^{1} \quad \alpha = 1$$

Newton's Method converges only linearly if $f'(\bar{x}) = 0$.

Another approach to convergence analysis

View Newton's Method as a fixed point iteration:

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

where

$$g(x) = x - \frac{f(x)}{f'(x)} = x$$

We can show that $g'(\bar{x}) = 0$, but $g''(\bar{x}) \neq 0$, so we get quadratic convergence.

Compute $g'(x)$, $g''(x)$:

$$g'(\bar{x}) = 1 - \frac{f'(\bar{x})^2 - \underbrace{f(\bar{x})}_{=0} f''(\bar{x})}{(f'(\bar{x}))^2} \qquad f'(\bar{x}) \neq 0$$

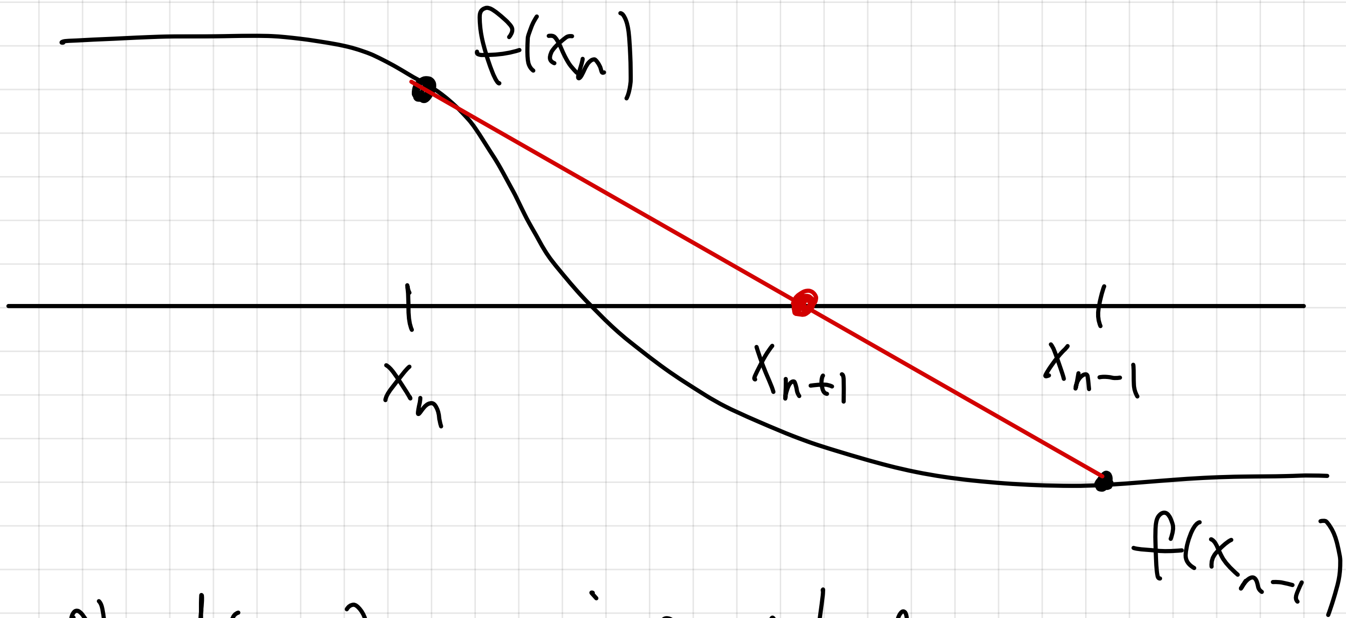
$$= \frac{f'(\bar{x})^2 - f'(\bar{x})^2}{f'(\bar{x})^2} = 0$$

Show $f''(\bar{x}) \neq 0 \Rightarrow$ quadratic converge.

Secant method

Can we avoid the computation of the derivative?

Idea: Use an approximation to the derivative (sort of like MFP)



- Needs 2 previous values
- Convergence less than that for Newton's method
- Uses only one function evaluation per iteration.

Newton's method

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

replace $f'(x_n)$ with an approximation to the derivative based on two previous values:

$$f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

Secant method

$$\underbrace{x_{n+1} = x_n - f(x_n)\left[\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}\right]}_{\text{exactly the MFP iteration.}}$$

But no attempt is made to keep the root inside an interval.

Secant method

```matlab
function [xroot, en] = secant(f,x0,x1,tol,kmax)

xkm1 = x0;
xk = x1;
fk = f(xk);
fkm1 = f(xkm1);
for k = 1:kmax
    xkp1 = xk - f(xk)*(xk-xkm1)/(fk-fkm1);
    fkp1 = f(xkp1);
    en(k) = abs(xkp1 - xk);
    fprintf('%5d %20.16f %12.4e\n',k,xkp1,en(k));
    if (en(k) < tol)
        fprintf('Tolerance achieved\n');
        xroot = xk;
        break;
    end
    xkm1 = xk;
    xk = xkp1;
    fkm1 = fk;
    fk = fkp1;
end
xroot = xk;

end
```

2 starting values needed.

only one function call/iteration

```
    1    0.2000000000000000    1.8000e+00
    2    0.3333333333333333    1.3333e-01
    3    0.4083601286173633    7.5027e-02
    4    0.3905936753703533    1.7766e-02
    5    0.3915842969362032    9.9062e-04
    6    0.3916002268150462    1.5930e-05
    7    0.3916002113179452    1.5497e-08
    8    0.3916002113181834    2.3820e-13
    9    0.3916002113181835    5.5511e-17
Tolerance achieved
         Secant :    0.391600213181834
```

Secant Method - Order of convergence

Outline of steps:

① Start with numerical scheme

$$x_{n+1} = x_n - f(x_n)\left[\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}\right]$$

② Convert this to a scheme for the error $e_n = x_n - \bar{x}$:

$$e_{n+1} = e_n - f(x_n)\left[\frac{e_n - e_{n-1}}{f(x_n) - f(x_{n-1})}\right]$$

③ Expand $f(x_n), f(x_{n-1})$ about $\bar{x}$:

$$f(x_n) \approx f'(\bar{x})e_n + \frac{1}{2}f''(\bar{x})e_n^2$$

$$f(x_{n-1}) \approx f'(\bar{x})e_{n-1} + \frac{1}{2}f''(\bar{x})e_{n-1}^2$$

$$f(x_n) - f(x_{n-1}) \approx (e_n - e_{n-1})\left(f'(\bar{x}) + \frac{1}{2}f''(\bar{x})(e_n + e_{n-1})\right)$$

(4) Simplify the expression and drop term in the denominator involving $e_n + e_{n+1}$:

$$\Rightarrow \quad e_{n+1} = e_n e_{n-1} \frac{f''(\bar{x})}{2f'(\bar{x}) + f''(\bar{x})\cancel{(e_n + e_{n-1})}}$$

$$\Rightarrow \quad e_{n+1} = \underbrace{\left(\frac{f''(\bar{x})}{2f'(\bar{x})}\right)}_{C} e_n e_{n-1} \equiv C e_n e_{n-1}$$

(5) Solve for α: Use $e_{n+1} = \lambda e_n^{\alpha}$; $e_n = \lambda e_{n-1}^{\alpha}$

$$\lambda e_n^{\alpha} = C \lambda^{-1/\alpha} e_n^{1+1/\alpha}$$

equate powers of α:

$$\alpha = 1 + \frac{1}{\alpha}$$

Choose positive root

$$\boxed{\alpha = \frac{1+\sqrt{5}}{2}}$$

Golden ratio!

Order of convergence for the secant method is approximately 1.618...

Secant Method - Order of Convergence

Details

$$x_{n+1} = x_n - f(x_n)\left[\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}\right]$$

$$\underbrace{x_{n+1} - \bar{x}}_{e_{n+1}} = \underbrace{x_n - \bar{x}}_{e_n} - f(x_n)\left[\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}\right]$$

Taylor Series about $\bar{x}$:

$$f(x_{n-1}) \approx \cancel{f(\bar{x})} + f'(\bar{x})(x_{n-1} - \bar{x}) + \frac{1}{2} f''(\bar{x})(x_{n-1} - \bar{x})^2$$

$$f(x_n) \approx \cancel{f(\bar{x})} + f'(\bar{x})(x_n - \bar{x}) + \frac{1}{2} f''(\bar{x})(x_n - \bar{x})^2$$

$$f(x_n) - f(x_{n-1}) = f'(\bar{x})(x_n - x_{n-1})$$

$$+ \frac{1}{2} f''(\bar{x})\left[x_n - x_{n-1}\right]\left[x_n + x_{n-1} - 2\bar{x}\right]$$

$$= \frac{(x_n - x_{n-1})}{2}\left[2f'(\bar{x}) + f''(\bar{x})(x_n + x_{n-1} - 2\bar{x})\right.$$

$$\underbrace{x_{n+1}-\bar{x}}_{e_{n+1}} = \underbrace{x_n-\bar{x}}_{e_n} - f(x_n)\left[\frac{x_n - x_{n-1}}{f(x_n)-f(x_{n-1})}\right]$$

$$= (x_n-\bar{x}) - 2\left[\frac{f'(\bar{x})(x_n-\bar{x})+\frac{1}{2}f''(\bar{x})(x_n-\bar{x})^2}{2f'(\bar{x})+f''(\bar{x})(x_n+x_{n-1}-2\bar{x})}\right]$$

$$= (x_n-\bar{x})\left[1-2\left[\frac{f'(\bar{x})+\frac{1}{2}f''(\bar{x})(x_n-\bar{x})}{2f'(\bar{x})+f''(\bar{x})(x_n+x_{n-1}-2\bar{x})}\right]\right]$$

$$= (x_n-\bar{x})\left[\frac{f''(\bar{x})\big(x_n-\bar{x}+(x_{n-1}-\bar{x})\big)-f''(\bar{x})(x_n-\bar{x})}{2f'(\bar{x})+f''(\bar{x})(x_n+x_{n-1}-2\bar{x})}\right]$$

$$= (x_n-\bar{x})(x_{n-1}-\bar{x})\left[\frac{f''(\bar{x})}{2f'(\bar{x})+f''(\bar{x})(x_n+x_{n-1}-2\bar{x})}\right]$$

as $n\to\infty$, the term $(x_n-\bar{x})+(x_{n-1}-\bar{x})$ approaches 0 and can be dropped. The result is

$\Longrightarrow$

Secant Method - continued:

$$(*) \quad e_{n+1} = e_n e_{n-1}\left(\frac{f''(\bar{x})}{2f'(x)}\right) \equiv c\, e_n e_{n-1}$$

c

Now assume that $e_n \approx \lambda e_{n-1}^{\alpha}$

$$e_{n+1} = \lambda e_n^{\alpha}$$

$$e_n = \lambda e_{n-1}^{\alpha} \Rightarrow e_{n-1} = \left(\frac{1}{\lambda} e_n\right)^{1/\alpha}$$

Substitute these expressions into $(*)$ to get

$$\lambda e_n^{\alpha} = c\, e_n \left(\frac{1}{\lambda} e_n\right)^{1/\alpha}$$

Equate powers of α:

$$\lambda e_n^{\alpha} = c \lambda^{-1/\alpha} e_n^{1+1/\alpha}$$

Golden ratio!

$$\Rightarrow \quad \alpha = 1 + \frac{1}{\alpha}$$

Choose positive root

$$\Rightarrow \quad \alpha^2 - \alpha - 1 = 0 \Rightarrow \alpha = \frac{1+\sqrt{5}}{2}$$

Order of convergence: $\alpha \sim 1.618$