Therefore ... $\alpha^3 = 1+\alpha+\alpha^2$ this is cubic equation and the order of the method is the largest real root.

Solve using numerical techniques

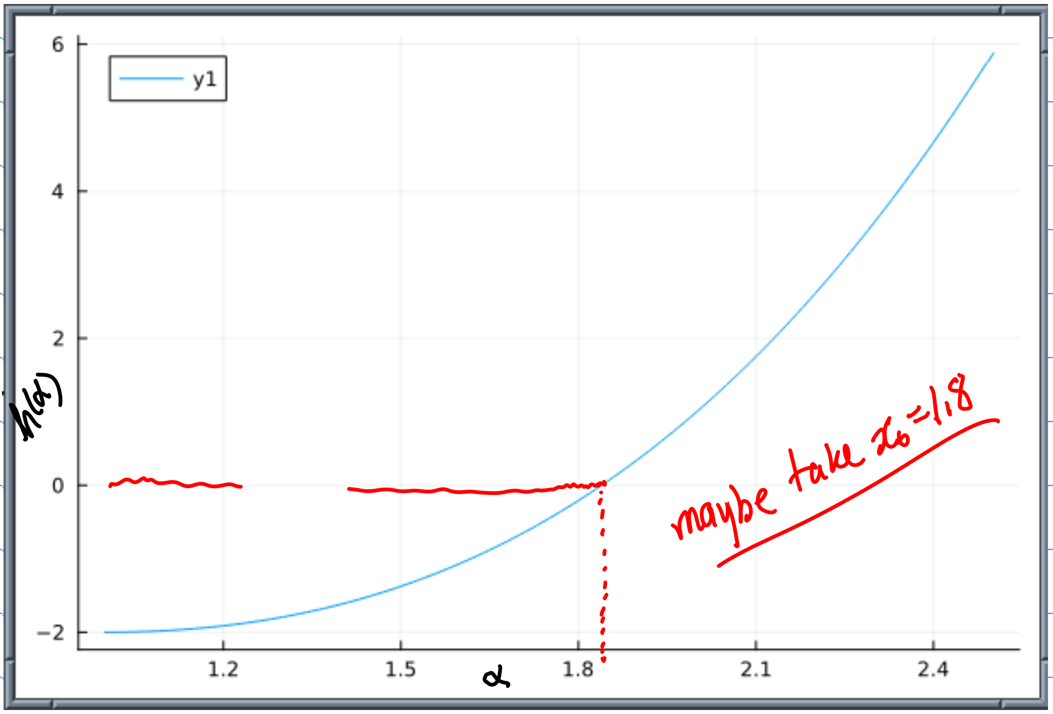
$$h(\alpha) = \alpha^3 - \alpha^2 - \alpha - 1$$
$$= (\alpha^2 - \alpha)\alpha - \alpha - 1$$
$$= (\alpha^2 - \alpha - 1)\alpha - 1$$
$$= ((\alpha - 1)\alpha - 1)\alpha - 1$$

$$h'(\alpha) = 3\alpha^2 - 2\alpha - 1$$
$$= (3\alpha - 2)\alpha - 1$$

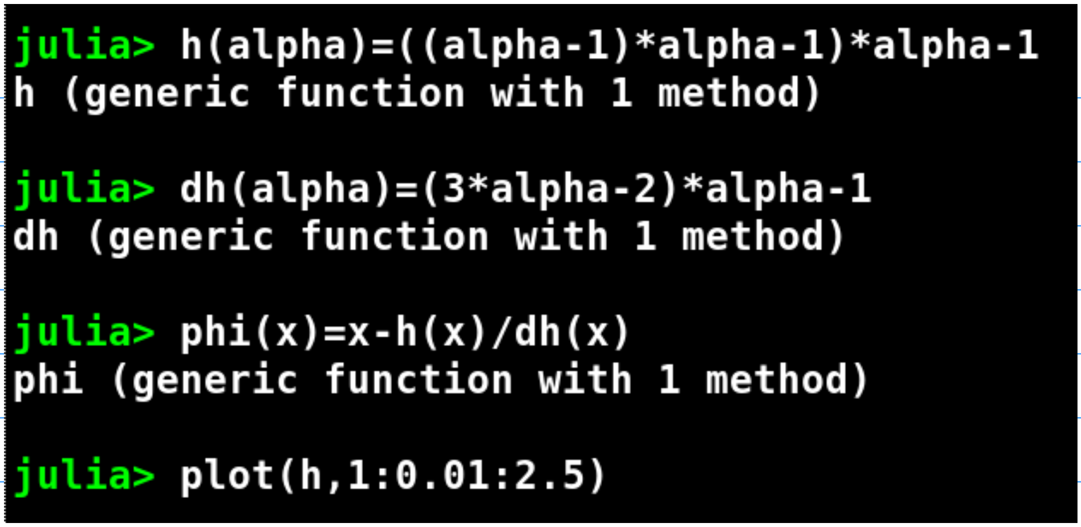
$$\phi(x) = x - \frac{h(x)}{h'(x)}$$ ← Newton

x_0 is an approximation of the root

$x_{n+1} = \phi(x_n)$ improved approximation ... iterate



maybe take $x_0 = 1.8$

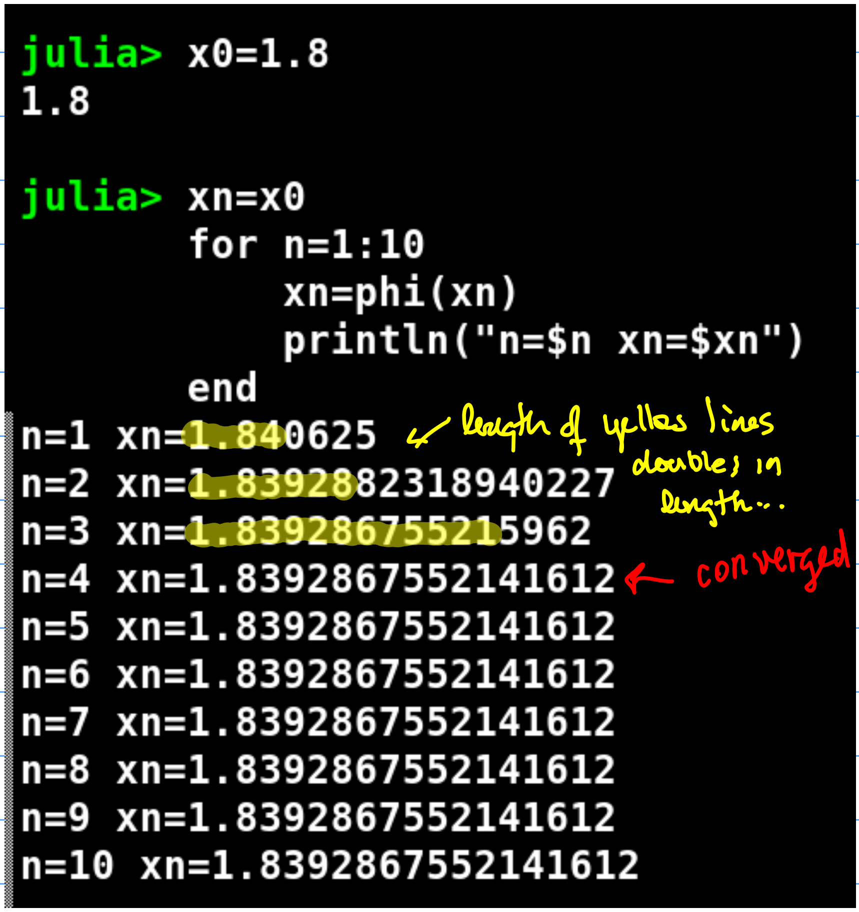


```
julia> h(alpha)=((alpha-1)*alpha-1)*alpha-1
h (generic function with 1 method)

julia> dh(alpha)=(3*alpha-2)*alpha-1
dh (generic function with 1 method)

julia> phi(x)=x-h(x)/dh(x)
phi (generic function with 1 method)

julia> plot(h,1:0.01:2.5)
```



```
julia> x0=1.8
1.8

julia> xn=x0
       for n=1:10
           xn=phi(xn)
           println("n=$n xn=$xn")
       end
n=1 xn=1.840625
n=2 xn=1.8392882318940227
n=3 xn=1.839286755215962
n=4 xn=1.8392867552141612
n=5 xn=1.8392867552141612
n=6 xn=1.8392867552141612
n=7 xn=1.8392867552141612
n=8 xn=1.8392867552141612
n=9 xn=1.8392867552141612
n=10 xn=1.8392867552141612
```

← length of yellow lines doubles in length...

← converged at n=4

Thus $\alpha = 1.839\ldots$ and for inverse interpolation using quadratics we have

$$|E_{n+1}| \approx M|E_n|^{1.839}$$

← the # of sig digits increases by 84% each iteration...

Recall Newton's method has $\alpha=2$ convergence and the number of significant digits approximately doubles with each iteration

Note that superlinear convergence (when $\alpha > 1$) is noticeably faster than linear convergence (when $\alpha = 1$).

recall linear convergence meant

$$|\varepsilon_{n+1}| \le \gamma |\varepsilon_n|^1$$

by induction $|\varepsilon_n| \le \gamma^n |\varepsilon_0|$

same number of digits more are ✓ correct each time

$\gamma < 1$ and so the error decreases as an exponential function of # of iterations.

Increasing the # of significant digits each iteration by a certain percent is much faster.

$n=2$ quadratic inverse interpolation

From $\varepsilon_{k+1} \approx M \varepsilon_k \varepsilon_{k-1} \varepsilon_{k-2}$

$$\alpha^{\overset{n+1}{3}} = 1 + \alpha + \alpha^2$$

arbitrary n

$$\varepsilon_{k+1} \approx M \underbrace{\varepsilon_{k-n} \varepsilon_{k-n+1} \cdots \varepsilon_k}_{n+1 \text{ terms}}$$

$$\alpha^{n+1} = 1 + \alpha + \cdots + \alpha^n$$

order given by this polynomial.

n=1 linear inverse interpolation (since a line is still a line when reflected along the 45° diagonal then this is also the same a direct interpolation)

$\alpha^2 = 1 + \alpha$ $\qquad$ $\alpha^2 - \alpha - 1 = 0$

$a=1 \quad b=-1 \quad c=-1$

Then

$$x = \frac{-b \pm \sqrt{b^2-4ac}}{2a} = \frac{1 \pm \sqrt{1+4}}{2} = \frac{1 \pm \sqrt{5}}{2}$$

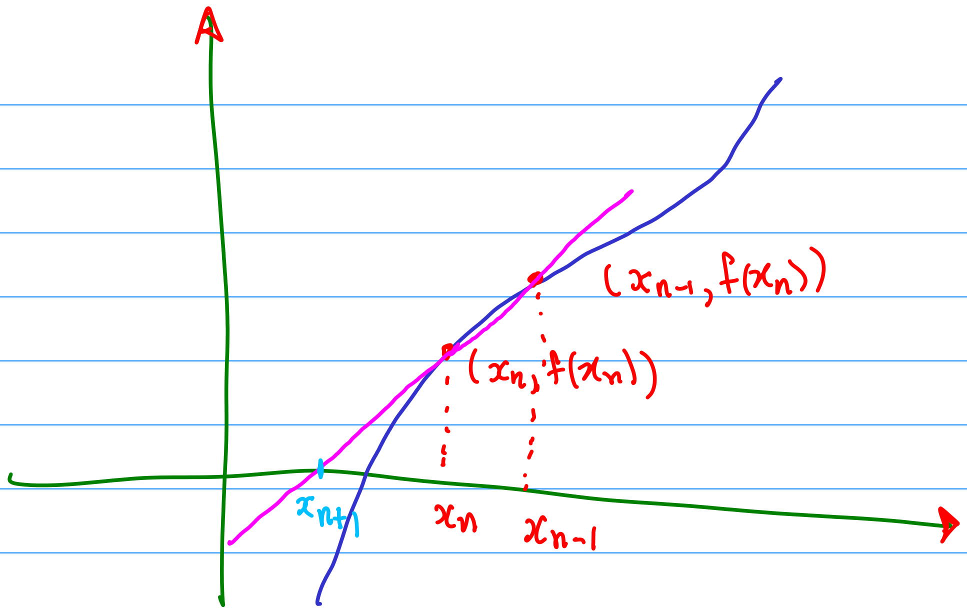
largest root is $\alpha = \frac{1+\sqrt{5}}{2}$ ↖ (Golden ratio)

Also known a secant method.

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

$$x_{n+1} \approx x_n - f(x_n) \Big/ \frac{f(x_n)-f(x_{n-1})}{x_n - x_{n-1}}$$

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



What about larger n?

$$\alpha^{n+1} = \underbrace{1+\alpha+\dots+\alpha^n}_{\text{geometric sum}}$$

$$S_n = 1+\alpha+\dots+\alpha^n$$

$$\alpha S_n = \alpha+\alpha^2+\dots+\alpha^{n+1}$$

$$(1-\alpha)S_n = 1-\alpha^{n+1}$$

$$S_n = \frac{1-\alpha^{n+1}}{1-\alpha}$$

geometric sum formula

Thus

$$\alpha^{n+1} = \frac{1-\alpha^{n+1}}{1-\alpha}$$

$$(1-\alpha)\alpha^{n+1} = 1-\alpha^{n-1}$$

$$\alpha^{n+1}-\alpha^{n+2} = 1-\alpha^{n+1}$$

$$\boxed{\alpha^{n+2}-2\alpha^{n+1}+1=0}$$

∞ What is the limit equation as $n \to \infty$?

$$\frac{\alpha^{n+2}}{\alpha^{n+1}} - \frac{2\alpha^{n+1}}{\alpha^{n+1}} + \frac{1}{\alpha^{n+1}} = 0$$

$$\alpha - 2 + \underset{\substack{\downarrow \\ 0}}{\frac{1}{\alpha^{n+1}}} = 0$$

assume $\alpha > 1$

so $\frac{1}{\alpha^{n+1}} \to 0$ as $n \to \infty$

$n \to \infty$

$$\alpha - 2 = 0$$

- Solution $\alpha = 2$. What happens in the limit. Basically the order of convergence of all inverse interpolation method's is less than 2 but approaching 2 as the degree of the polynomial increases.

* * *

Newtons method for systems:

$f : \mathbb{R}^m \to \mathbb{R}^n \leftarrow$ nonlinear function...

Want to find roots to $f(x) = 0$

How did we get Newton's method for one equation.

$$g(x) = x + \overset{}{\underset{\uparrow}{c}} f(x)$$

then choose c so converges

$$\varepsilon_k = x_k - r$$

$$\varepsilon_{k+1} = x_{k+1} - r = g(x_k) - g(r)$$

mean value thm

$$= g'(c_k)\,(x_k - r) = g'(c_k)\,\varepsilon_k$$

Thus $|\varepsilon_{k+1}| = \underset{\uparrow}{|g'(c_k)|}\,|\varepsilon_k|$ where c_k is between x_k and r.

want as small as possible at the root ($g'(r)$ governs how large $g'(c_k)$ is when c_k is near r.)

$$g(x) = x + cf(x)$$

$$g'(x) = 1 + cf'(x)$$

$$g'(r) = 1 + cf'(r) = 0$$

smallest.

Therefore $c = -\dfrac{1}{f'(r)} \approx \dfrac{-1}{f'(x_n)}$

Same for systems...

$f: \mathbb{R}^m \to \mathbb{R}^n$ $\qquad x \in \mathbb{R}^m$

$$g(x) = \underset{m}{x} + \underset{m\times n}{A}\,\underset{n}{f(x)}$$

Find A so that the derivative of g is as small as possible at the root.

$$g: \mathbb{R}^m \to \mathbb{R}^m$$

$$\underset{m\times m}{Dg(x)} = \underset{m\times m}{I} + \underset{m\times n}{A}\underbrace{\overbrace{Df(x)}^{\text{Matrix}}}_{n\times m}$$

Find A so $\| I + ADf(\ ) \|$ is minimized.