

Lecture 04 - B

§3.2 Newton's Method

T. Takahashi

Department of Mathematics and Statistics
University of New Mexico



Goals

We will answer these questions:

- How is Newton's method derived?
- How fast does Newton's method converge?
- What are some drawbacks to Newton's method?



Linearization



Linearization of a Function

Definition: Linearization

The **linearization** of a function $f(x)$ at $x = a$ is the equation of the tangent line to $f(x)$ at that point:

$$L(x) = f(a) + f'(a)(x - a)$$

Steps to Find the Linearization:

- 1 Evaluate the function at $x = a$: $f(a)$
- 2 Compute the derivative $f'(x)$ and evaluate it at $x = a$: $f'(a)$
- 3 Substitute these values into the formula for the linearization:

$$L(x) = f(a) + f'(a)(x - a)$$



Linearization

Example

Example

Find the linearization of $f(x) = \cos(x) - x$ at $x = \frac{\pi}{2}$.

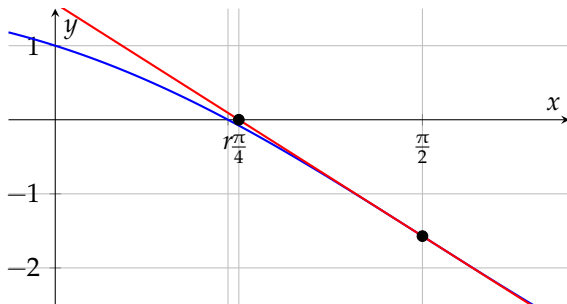
$$\begin{aligned}L(x) &= f(a) + f'(a)(x - a) \\&= f\left(\frac{\pi}{2}\right) + f'\left(\frac{\pi}{2}\right)(x - \frac{\pi}{2}) \\&= \left(-\frac{\pi}{2}\right) + \left(-\sin\left(\frac{\pi}{2}\right) - 1\right)(x - \frac{\pi}{2}) \\&= -\frac{\pi}{2} - 2(x - \frac{\pi}{2}) \\&= -2x + \frac{\pi}{2}\end{aligned}$$

Therefore, near $x = \frac{\pi}{2}$,

$$\cos(x) - x \approx -2x + \frac{\pi}{2}.$$



Can a Linearization Find a Root?

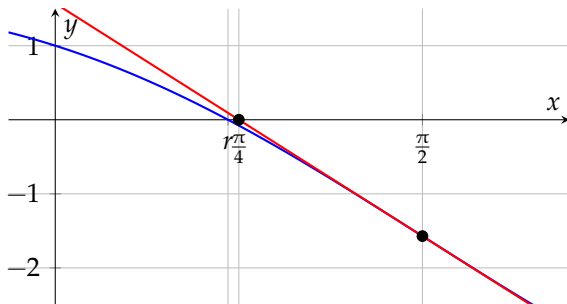


Question

Can we use the easier function $L(x)$ to estimate the root of $f(x)$?



Can a Linearization Find a Root?



$$L(x) = 0 \implies -2x + \frac{\pi}{2} = 0 \implies x = \frac{\pi}{4} \approx 0.785.$$

(True answer is about 0.73)



Linearization

Nonlinear Problem

Solving

$$\cos(x) - x = 0$$

is *impossible*.

Linearized Problem

Approximate the function with a straight line

$$\cos(x) - x \approx -2x + \frac{\pi}{2}$$

Solving

$$-2x + \frac{\pi}{2} = 0$$

is *trivial*.

Key Idea

Linearization can help us approximate the root of nonlinear functions.

Newton's Method



Do It Again

- Start with an initial guess x_0 and linearize

$$f(x) \approx L_0(x)$$

- Solve $L_0(x) = 0$ to get x_1

$$x_0 \longrightarrow x_1$$



Do It Again

- Start with an initial guess x_0 and linearize

$$f(x) \approx L_0(x)$$

- Solve $L_0(x) = 0$ to get x_1

$$x_0 \longrightarrow x_1$$

- Then we can use x_1 as the center of a new linearization $L_1(x)$ and get x_2

$$x_0 \longrightarrow x_1 \longrightarrow x_2$$



Do It Again

- Start with an initial guess x_0 and linearize

$$f(x) \approx L_0(x)$$

- Solve $L_0(x) = 0$ to get x_1

$$x_0 \longrightarrow x_1$$

- Then we can use x_1 as the center of a new linearization $L_1(x)$ and get x_2

$$x_0 \longrightarrow x_1 \longrightarrow x_2$$

- We can do this an infinite number of times

$$x_0 \longrightarrow x_1 \longrightarrow x_2 \longrightarrow x_3 \cdots$$



Do It Again

- Start with an initial guess x_0 and linearize

$$f(x) \approx L_0(x)$$

- Solve $L_0(x) = 0$ to get x_1

$$x_0 \longrightarrow x_1$$

- Then we can use x_1 as the center of a new linearization $L_1(x)$ and get x_2

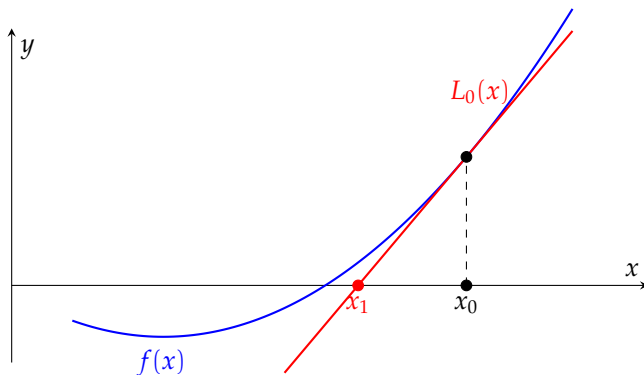
$$x_0 \longrightarrow x_1 \longrightarrow x_2$$

- We can do this an infinite number of times

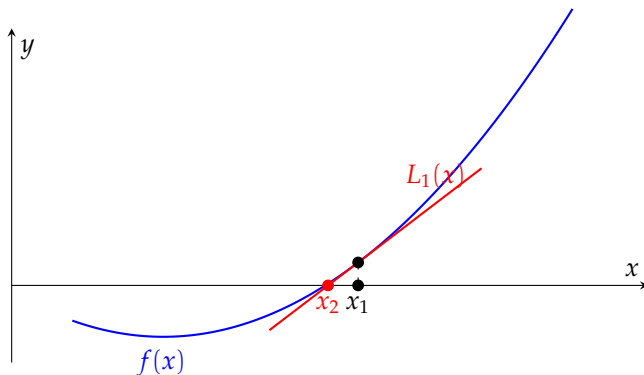
$$x_0 \longrightarrow x_1 \longrightarrow x_2 \longrightarrow x_3 \cdots$$

Key Idea

We can repeatedly use linearization to get closer to the true root.



- Linearize around x_0 and solve $L_0(x_1) = 0$



- Linearize around x_0 and solve $L_0(x_1) = 0$
- Linearize around x_1 and solve $L_1(x_2) = 0$

Formula

- Suppose our current approximation is x_k .
- Linearize f around x_k is

$$f(x) \approx L(x) = f(x_k) + f'(x_k)(x - x_k).$$

- We choose x_{k+1} to be the root of this tangent line:

$$L(x_{k+1}) = 0.$$



Formula

- Suppose our current approximation is x_k .
- Linearize f around x_k is

$$f(x) \approx L(x) = f(x_k) + f'(x_k)(x - x_k).$$

- We choose x_{k+1} to be the root of this tangent line:

$$L(x_{k+1}) = 0.$$

- Therefore,

$$0 = f(x_k) + f'(x_k)(x_{k+1} - x_k).$$



Formula

- Suppose our current approximation is x_k .
- Linearize f around x_k is

$$f(x) \approx L(x) = f(x_k) + f'(x_k)(x - x_k).$$

- We choose x_{k+1} to be the root of this tangent line:

$$L(x_{k+1}) = 0.$$

- Therefore,

$$0 = f(x_k) + f'(x_k)(x_{k+1} - x_k).$$

Solving for x_{k+1} gives us the update formula

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



Newton's Method

Formula

Start with an initial guess x_0 then iterate using the recurrence formula:

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

Method: Newton's Method

- **Given:** $f(x)$, $f'(x)$, initial guess x_0
- **Output:** approximate solution to $f(x) = 0$
- **Idea:** Leverage function's derivative to converge quickly.

Why “Newton's Method”?

Newton developed an early version of the method in the 1600s. Raphson later simplified the iteration into a form closer to the one used today. For this reason, it is also called the **Newton-Raphson method**.

Newton By Hand

Example

Write a Newton recurrence formula to solve the equation

$$x = x^{1/3} + 2$$

starting with $x_0 = 3$.

- First, rewrite the equation in the form $f(x) = 0$:

$$x = x^{1/3} + 2$$

$$x - x^{1/3} - 2 = 0$$

- Next, find the derivative

$$f'(x) = 1 - \frac{1}{3}x^{-2/3}$$

- Substitute into the recurrence formula

$$x_{k+1} = x_k - \frac{x_k - x_k^{1/3} - 2}{1 - \frac{1}{3}x_k^{-2/3}}$$



Numerical Test

$$x_{k+1} = x_k - \frac{x_k - x_k^{1/3} - 2}{1 - \frac{1}{3}x_k^{-2/3}}$$

k	x_k	$f'(x_k)$	$f(x)$
0	3	0.83975005	-0.44224957
1	3.52664429	0.85612976	0.00450679
2	3.52138015	0.85598641	3.771×10^{-7}
3	3.52137971	0.85598640	2.664×10^{-15}
4	3.52137971	0.85598640	0.0

- Newton's method finishes in only **4 iterations**
- Bisection requires around **50 iterations**
- Newton's method has a fast rate of convergence



Convergence Analysis



Convergence Theorem

Recall: Convergence Rate

Convergence of a method is said to be of order r if there is a constant $C > 0$

$$\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|^r} \leq C$$

Theorem

Suppose

- f is sufficiently smooth near the root x_*
- The derivative at the root $f'(x_*) \neq 0$
- Initial guess x_0 is sufficiently close to x_*

Then Newton's Method converges with order $r = 2$ (quadratic).



- Suppose the true root of $f(x)$ is x_* . Then by Taylor's theorem:

$$f(x_*) = f(x_k) + (x_* - x_k)f'(x_k) + \frac{1}{2}(x_* - x_k)^2 f''(\xi) \equiv 0$$

- Factor out $f'(x_k)$ to get

$$\frac{f(x_k)}{f'(x_k)} + x_* - x_k + (x_* - x_k)^2 \left(\frac{1}{2}\right) \frac{f''(\xi)}{f'(x_k)} = 0$$

- Then

$$\left(\frac{f(x_k)}{f'(x_k)} - x_k\right) + x_* + (x_* - x_k)^2 \left(\frac{1}{2}\right) \frac{f''(\xi)}{f'(x_k)} = 0$$
$$x_* - x_{k+1} + (x_* - x_k)^2 \left(\frac{1}{2}\right) \frac{f''(\xi)}{f'(x_k)} = 0$$

- Factor out $(x_* - x_k)^2$ to get

$$\frac{x_* - x_{k+1}}{(x_* - x_k)^2} = - \left(\frac{1}{2}\right) \frac{f''(\xi)}{f'(x_k)}$$



Proof

Part 2

- If we define error to be $e_k = x_* - x_k$ then in absolute value

$$\frac{|x_* - x_{k+1}|}{|x_* - x_k|^2} = \frac{|e_{k+1}|}{|e_k|^2} = \left(\frac{1}{2}\right) \left| \frac{f''(\xi)}{f'(x_k)} \right|$$

- We assumed f to be smooth and $f'(x_*) \neq 0$ so

$$\left| \frac{f''(\xi)}{f'(x_k)} \right| < C$$

- We have shown that

$$\frac{|e_{k+1}|}{|e_k|^2} \leq \left(\frac{C}{2}\right)$$

Key Idea

Equation $|e_{k+1}| = C|e_k|^2$ means that errors decreases **quadratically** ($r = 2$). This is an entire order faster than bisection ($r = 1$).

Quadratic Convergence

Question

How fast is quadratic convergence?

- Suppose your starting error is

$$E_0 = 10^{-2}$$



Quadratic Convergence

Question

How fast is quadratic convergence?

- Suppose your starting error is

$$E_0 = 10^{-2}$$

- After one iteration

$$E_1 \approx E_0^2 = 10^{-4}$$



Quadratic Convergence

Question

How fast is quadratic convergence?

- Suppose your starting error is

$$E_0 = 10^{-2}$$

- After one iteration

$$E_1 \approx E_0^2 = 10^{-4}$$

- Second iteration

$$E_2 \approx E_1^2 = 10^{-8}$$



Quadratic Convergence

Question

How fast is quadratic convergence?

- Suppose your starting error is

$$E_0 = 10^{-2}$$

- After one iteration

$$E_1 \approx E_0^2 = 10^{-4}$$

- Second iteration

$$E_2 \approx E_1^2 = 10^{-8}$$

- And third

$$E_3 \approx E_2^2 = 10^{-16} \approx \epsilon_m$$

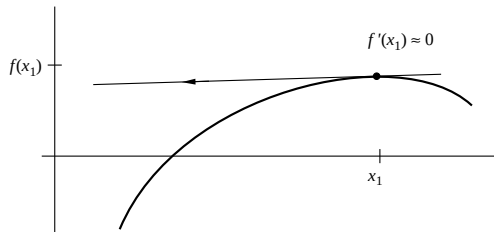
- Quadratic convergence is *very* fast



Some Issues



Issue 1: Divergence



- If $f'(x_1) \approx 0$, then the update x_2 may jump very far away
- In this case, Newton's method may **diverge** ($|x_k| \rightarrow \infty$)
- If $f'(x_k) = 0$ then we have division by zero in

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



Issue 2: Infinite Cycle

Consider finding the root of the following function with Newton's method

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

- The iteration becomes

$$x_{k+1} = x_k - \frac{x_k^3 - 2x_k + 2}{3x_k^2 - 2}.$$

- Starting with $x_0 = 0$:

k	x_k	$f'(x_k)$	$f(x_k)$
0	0	-2	2
1	-1	1	3
2	1	1	1
3	-1	1	3
4	1	1	1

- Newton's method is stuck in an infinite two-stage cycle



Issue 3: Slow Convergence

Definition: Multiple Zeros

- A function has a **double zero** at x_* if $f(x_*) = 0$ and $f'(x_*) = 0$.
- More generally, a function has a root of **multiplicity** m if

$$f^{(k)}(x_*) = 0, \quad 0 \leq k < m, \quad \text{and} \quad f^{(m)}(x_*) \neq 0.$$

- For example, $p(x) = x^2 - 2x + 1 = (x - 1)^2$ has a double zero at $x = 1$
 - $p(1) = 0$
 - $p'(1) = 2 - 2 = 0$
 - $p''(1) = 2 \neq 0$
- Newton's method exhibits slow convergence when the root has multiple zeros

Theorem

*If $f(x)$ has a root at x_* of multiplicity $m \geq 2$ then Newton's method converges to x_* at a **linear rate**.*

Slow Convergence Proof

Part 1

1. **Taylor expansion of f .** We assume that f has a double zero $f(x_*) = f'(x_*) = 0$. Then

$$\begin{aligned} f(x) &= f(x^*) + f'(x^*)(x - x^*) + \frac{1}{2}f''(x^*)(x - x^*)^2 + \cdots \\ &\implies \\ f(x) &\approx \frac{1}{2}f''(x^*)(x - x^*)^2 \end{aligned}$$

2. **Taylor expansion of f'** Likewise, the Taylor expansion of $f'(x)$ yields

$$\begin{aligned} f'(x) &= f'(x_*) + f''(x_*)(x - x_*) + \cdots \\ &\implies \\ f'(x) &\approx f''(x_*)(x - x_*) \end{aligned}$$



Slow Convergence Proof

Part 2

3. Approximate f and f' at the current iterate.

$$f(x_k) \approx \frac{1}{2}f''(x_*)(x_k - x_*)^2, \quad f'(x_k) \approx f''(x_*)(x_k - x_*).$$

4. Insert these into Newton's update.

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \approx x_k - \frac{\frac{1}{2}f''(x_*)(x_k - x_*)^2}{f''(x_*)(x_k - x_*)} = x_k - \frac{1}{2}(x_k - x_*).$$

5. Subtract x^* both sides.

$$x_{k+1} - x^* \approx \frac{1}{2}(x_k - x^*).$$



Slow Convergence Proof

Part 3

6. **Absolute value both sides.** We finally obtain an approximate relation between the new error and current error

$$|x_{k+1} - x^*| \approx \frac{1}{2} |x_k - x^*|.$$

Key Idea

- If $f'(x^*) \neq 0$ (simple root), Newton's method is **quadratic**: $|e_{k+1}| \propto |e_k|^2$.
- If $f'(x^*) = 0$ (double zero), the rate drops to **linear**: $|e_{k+1}| \approx \frac{1}{2}|e_k|$.



- All of the issues with Newton's Method can be fixed by using an adaptive step size

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

- Many several choices for α_k
 - $\alpha = 1/2$
 - $\alpha = m$ (the multiplicity)
 - $\alpha = \operatorname{argmin}_{\alpha} \left| f \left(x_k - \alpha \frac{f(x_k)}{f'(x_k)} \right) \right|$

Key Idea

Drawbacks to the Newton's include:

- Sensitivity to small slopes
- Infinite cycles
- Slow convergence due to multiple zeros

but these obstacles can be overcome with a small modification.

Extra Topic: Fractals



Fractals: What?

Definition

Fractal: A mathematical pattern (geometric object) that is reproducible at **any** level of magnification or reduction.

Definition

Fractal: A term used by Benoit Mandelbrot to refer to geometric objects with fractional dimensions rather than integer dimensions. Also used "fractal" to refer to shapes that are self-similar: they look the same at any zoom level.



Fractals: Application

Scientificly used to describe highly irregular objects

- Fractal image compression
- Seismology
- Cosmology
- Life sciences:
 - Clouds and fluid turbulence
 - Trees
 - Coastlines

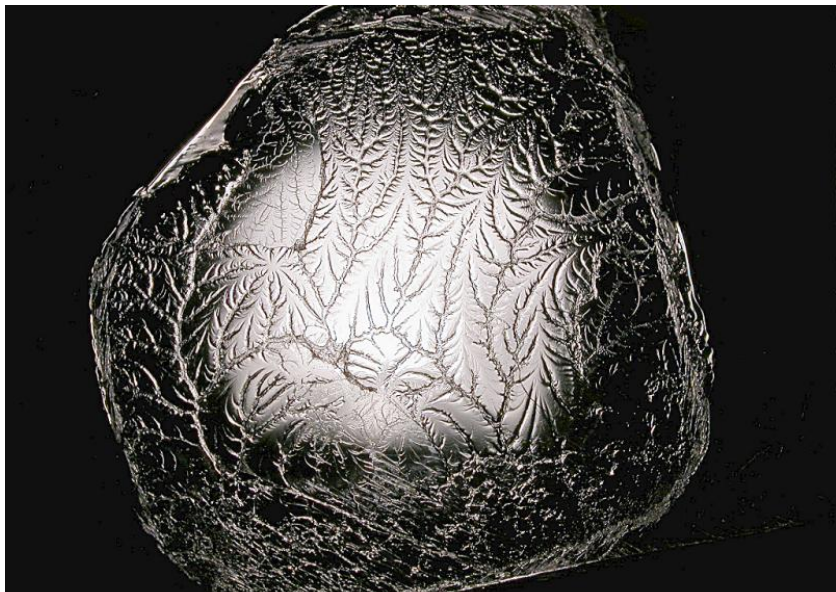
More interesting observations:

- New music/New art (Jackson Pollock)
- Video games/graphics
- Chaos theory
- The Butterfly effect: small changes produces large effects



Fractals: Air Pressure

Air channels between two glued pieces of acrylic



Fractals: high voltage dielectric breakdown

Lichtenberg: Branching discharges decrease to hairlike then to molecular



Fractals: Microwaving a CD

Heat vaporizes the aluminum leaving fractal metallic islands



Fractals: Romanesco Broccoli

Growth follows fractal pattern



Fractals: Jupiter

Atmosphere modeled with fractals



Fractals: Caves

Stalactite/Stalagmite formation



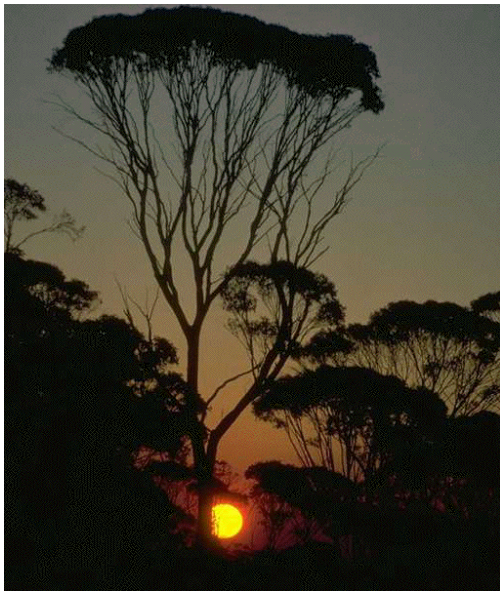
Fractals: Ferns

Growth



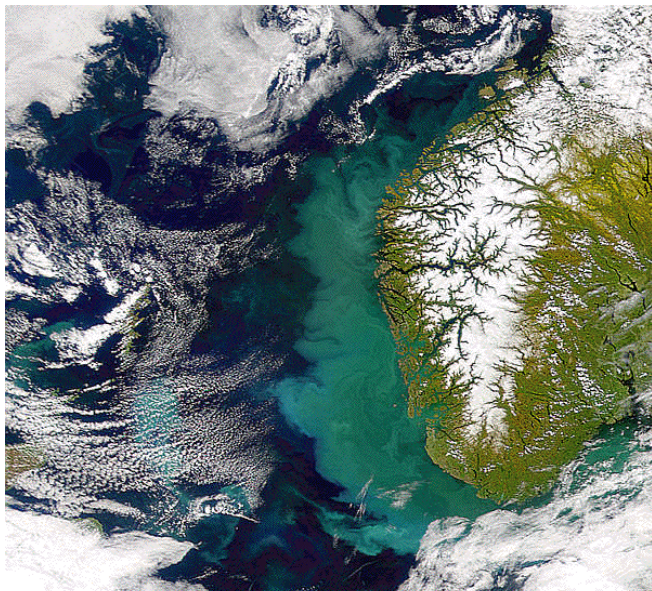
Fractals: Big Trees

Growth



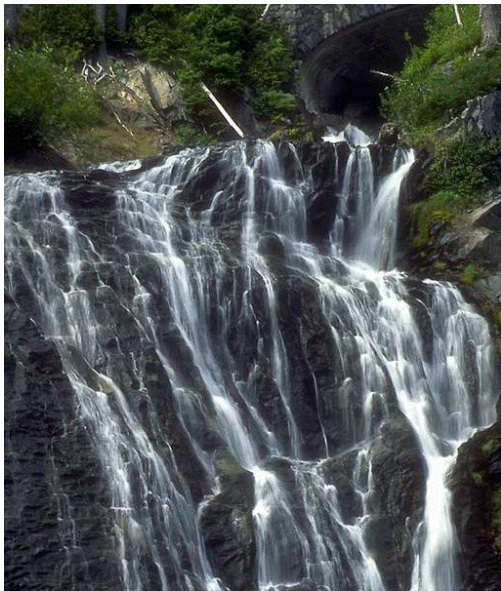
Fractals: Norwegian rivers

Structure



Fractals: waterfalls

Pattern



Recall Complex Numbers: $z \in \mathbb{C}$ means

$$z = x + iy,$$

where $i = \sqrt{-1}$

Things to notice:

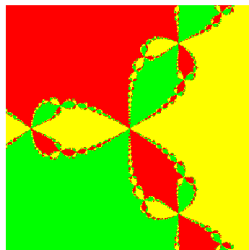
- still think of the x - y plane, but now it's in $\mathbb{C}^1$ instead of $\mathbb{R}^2$
- $f(z) = z^2 + 1$ has two roots: $z_{1,2} = \pm i$
- $f(z) = z^3 + 1$ has three roots: $z_1 = 1, z_{2,3} = \frac{-1 \pm i\sqrt{3}}{2}$
- $f(z) = z^4 + 1$ has four roots: $z_{1,2} = \pm 1, z_{3,4} = \pm i$



Fractals: Newton's Algorithm

How to make fractal patterns:

- 1 Choose a complex polynomial like $f(z) = z^3 + 1$ (has 3 roots)
- 2 Pick a bunch of initial guesses Z_0
- 3 Run Newton's Method for all initial points until they converge to one of the roots
- 4 Choose a color for each root
- 5 Color the initial points based on the root it converged to



Newton's Method for Complex Numbers

Derivatives for complex functions work just like real functions

- For a complex function $f(z) = z^3 + 1$

$$\frac{df}{dz} = 3z^2$$

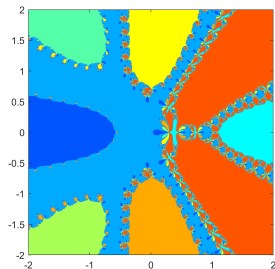
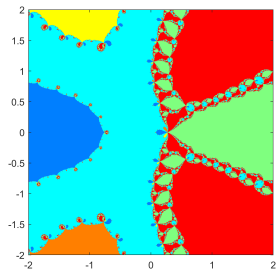
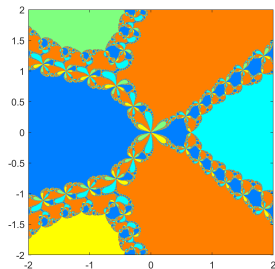
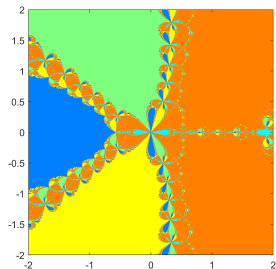
- The Newton method does not change at all

$$z_{k+1} = z_k - f(z_k)/f'(z_k)$$

- To make a fractal, you mesh the Complex plane very finely.
- Track where each point in the mesh converges to.



Newton Fractals

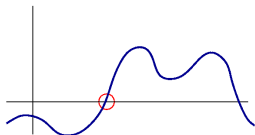


Extra Topic: Higher Dimensions



Root-finding in higher dimensions

- When x is in 1D, root finding $f(x) = 0$ is easy



- In 2D, solving $\mathbf{F}(x, y) = 0$ is harder but can be plotted
- For 3D and higher, there are no graphical tools
- Solving $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ can be difficult



Newton's Method in Higher Dimensions

Consider roots of $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ where,

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \vdots \\ f_n(\mathbf{x}) \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad \mathbf{0} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Example:

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \sin(x) + y + 4 \\ x^2 - xy \end{bmatrix}$$

Newton in one dimension:

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

Newton for system of nonlinear equations:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - [\mathbf{J}_{\mathbf{f}}(\mathbf{x}_k)]^{-1} \mathbf{f}(\mathbf{x}_k)$$



Newton's Method in Higher Dimensions

$$\mathbf{x}_{k+1} = \mathbf{x}_k - [\mathbf{J}_f(\mathbf{x}_k)]^{-1} \mathbf{f}(\mathbf{x}_k)$$

Here $\mathbf{J}_f$ is the Jacobian matrix (higher dimensional equivalent to a first derivative),

$$\mathbf{J}_f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

Each derivative is evaluated at $\mathbf{x}_k$.

Example:

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \sin(x) + y - 4 \\ x^2 - xy \end{bmatrix}, \quad \mathbf{J}_f(\mathbf{x}) = \begin{bmatrix} \cos(x) & 1 \\ 2x - y & -x \end{bmatrix}$$

Question: How do we compute $[\mathbf{J}_f(\mathbf{x}_k)]^{-1}$?



Newton's Method in Higher Dimensions

Solve (using Gaussian elimination, CG, etc...)

$$\mathbf{J}_f(\mathbf{x}_k)\mathbf{h}_k = -\mathbf{f}(\mathbf{x}_k)$$

and then set $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{h}_k$.

Example with initial guess $\mathbf{x}_0 = [1, -2]^T$:

$$\begin{bmatrix} 0.54 & 1 \\ 4 & -1 \end{bmatrix} \mathbf{h}_0 = \begin{bmatrix} -2.8 \\ -3 \end{bmatrix}$$

We get $\mathbf{h}_0 = [-1.2, -2.1]^T$ and so

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ -2 \end{bmatrix} + \begin{bmatrix} -1.2 \\ -2.1 \end{bmatrix} = \begin{bmatrix} -0.2 \\ -4.1 \end{bmatrix}$$



N-dimensional Newton's Method

For a particular $\mathbf{f}(\mathbf{x})$, the Jacobian $\mathbf{J}_f(\mathbf{x})$ is hardcoded. Then an initial guess $\mathbf{x}_0$ is chosen. Let $k = 0$.

The method:

- 1 Compute matrix $A = \mathbf{J}_f(\mathbf{x}_k)$
- 2 Compute vector $\mathbf{b} = -\mathbf{f}(\mathbf{x}_k)$
- 3 Solve the system $A\mathbf{h}_k = \mathbf{b}$
- 4 Update $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{h}_k$
- 5 $k++$
- 6 Repeat

Method: Newton's Method

- **Given:** Nonlinear problem $\mathbf{F}(\mathbf{x}) = \mathbf{0}$, initial guess $\mathbf{x}_0$
- **Output:** Approximate root $\mathbf{x}^*$
- **Idea:** Generalized version of the 1D Newton's Method

