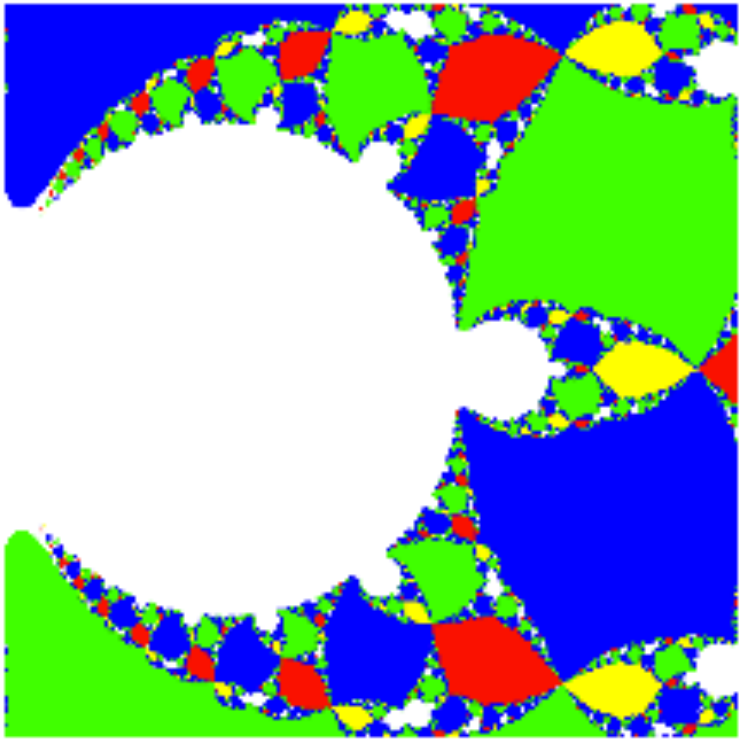


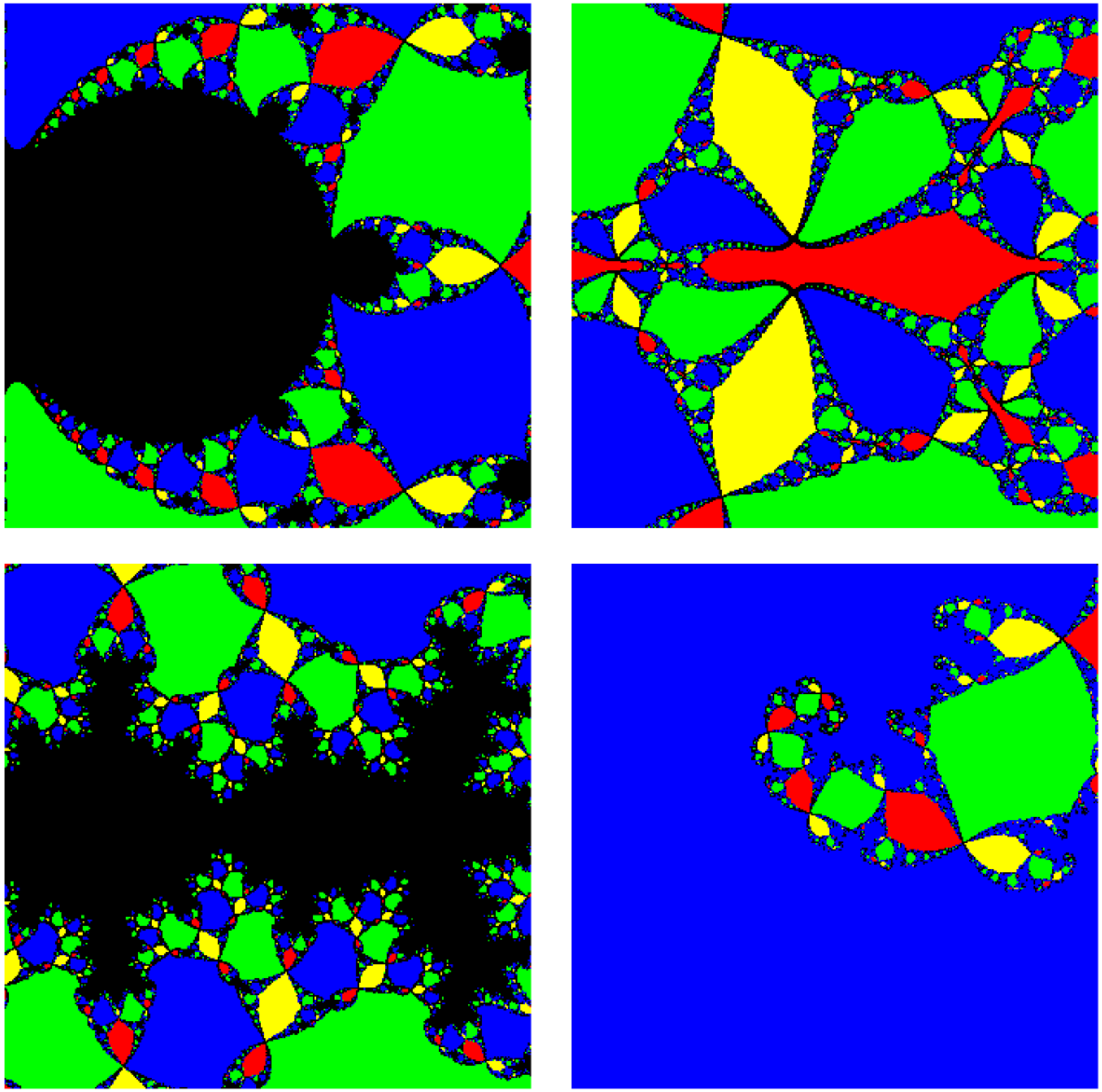
Newton Basins

Lab 4

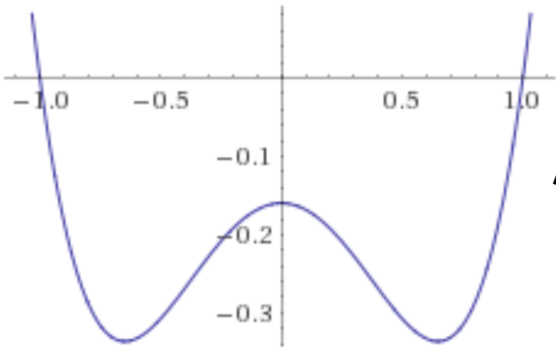
Who am I?

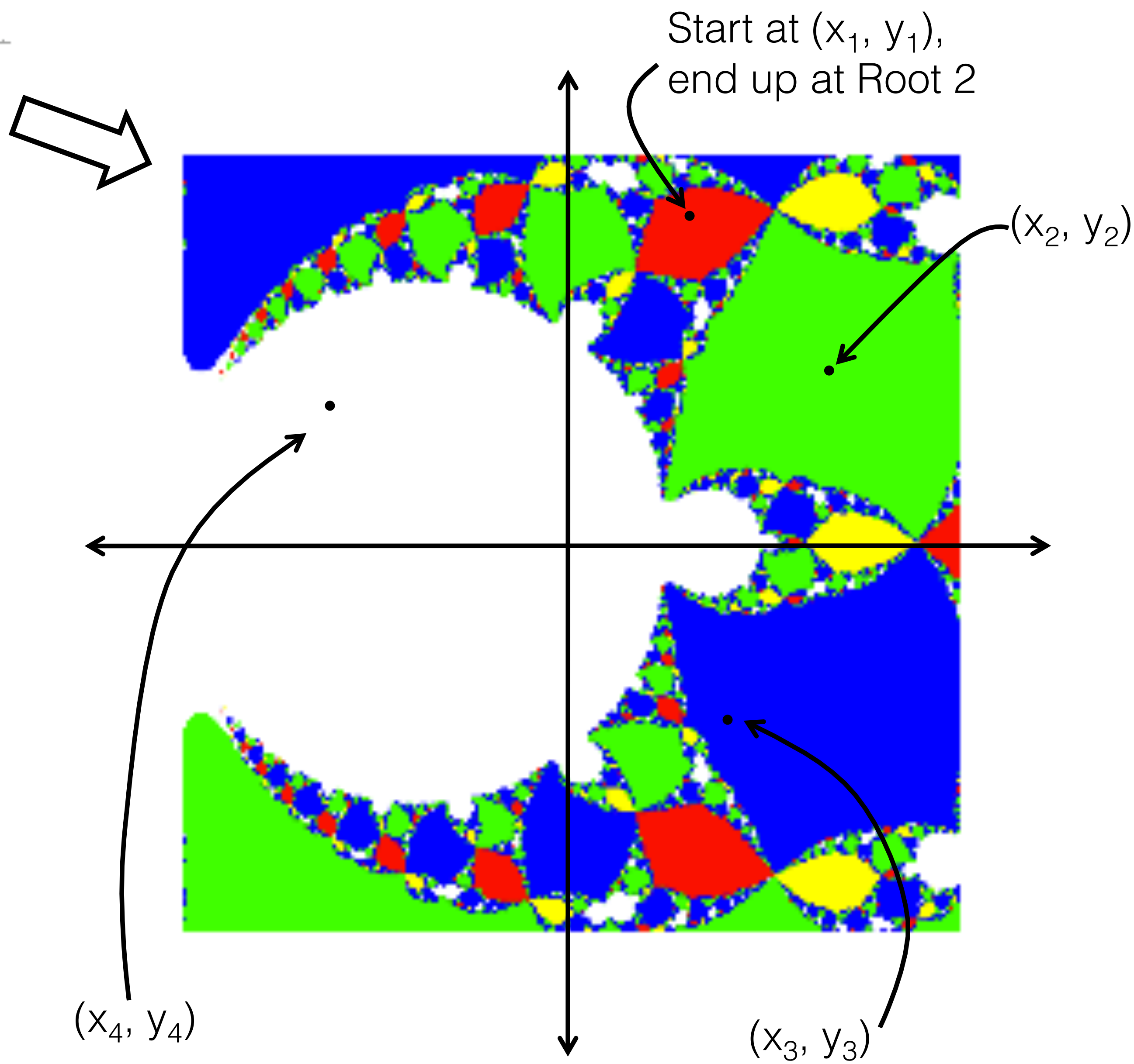
Sarah Schwettmann, a student of Dr. Cox's

You can find me (and this presentation) online at www.cogconfluence.com



We’ll call them “Newton Basins”





Root 1: Yellow

Root 2: Red

Root 3: green

Root 4: blue

I claim that you only need four color cases. What does it mean when we see white on the plot?

Each picture corresponds to a polynomial.

Specifically, we will be working with **quartics.**

How many roots do quartics have?

How many colors will we see on the plot?

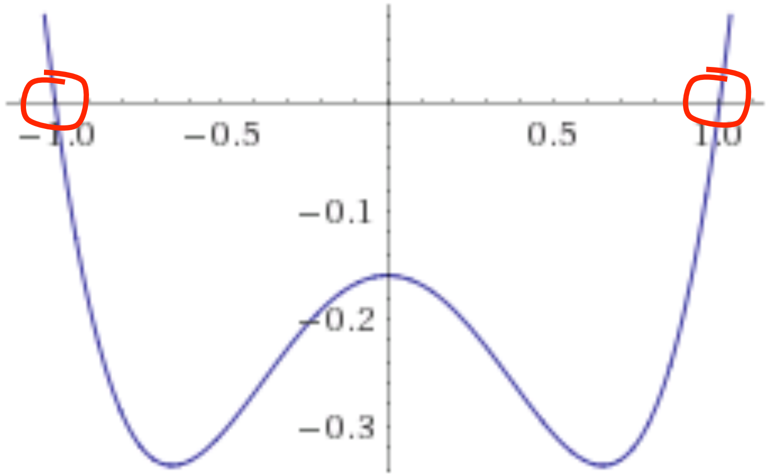
The grid is a coordinate plane, a large collection of points.

We take each point as a "starting guess" in Newton's method, and see which of the multiple roots of the polynomial Newton's method finds for it.

We then color that point, on the original axes, the color assigned to the found root.

…Voila!

But there is something strange here…

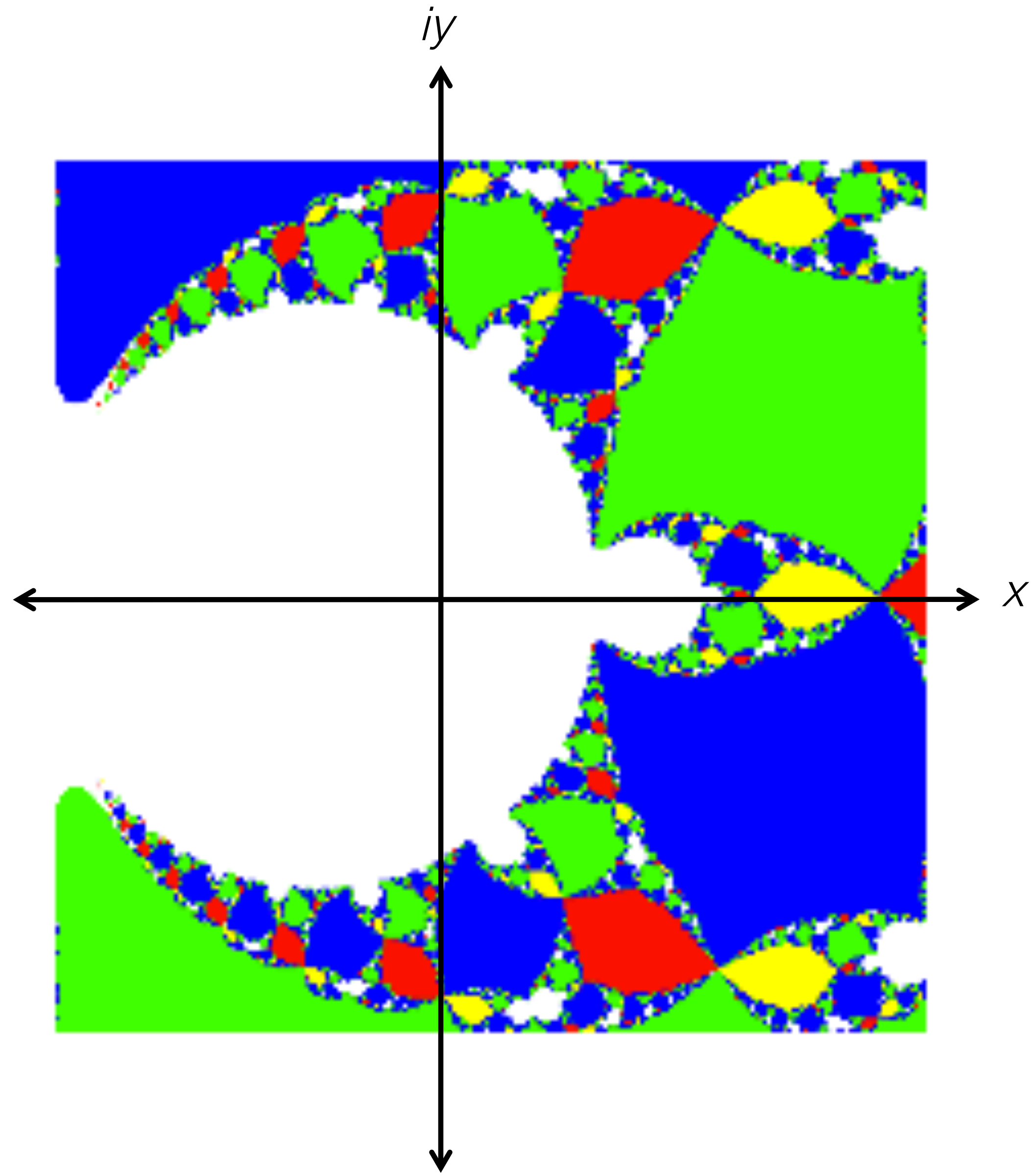


Root 1: Yellow

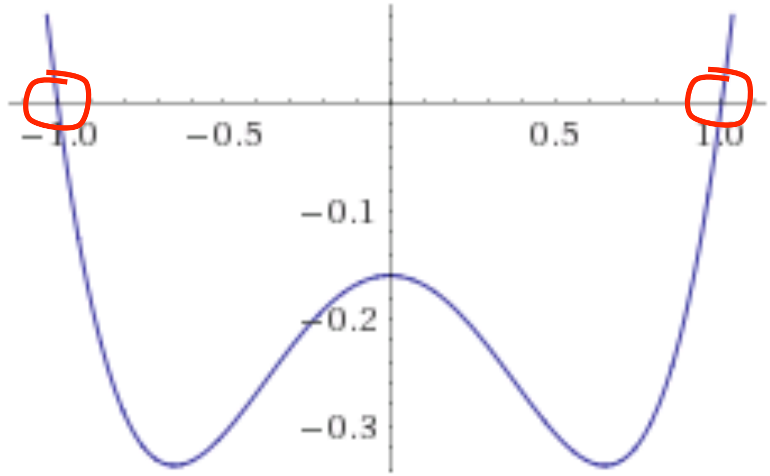
Root 2: Red

Root 3: green

Root 4: blue



But there is something strange here…



2 visible roots

4 colors for 4 roots

Root 1: Yellow
Root 2: Red
Root 3: green
Root 4: blue

$$z^4 - 0.84z^2 - 0.16$$

has roots:

$\left.\begin{array}{l} -1 \\ \;\;1 \end{array}\right\}$ visible on real axes (left)

$-0.4i$

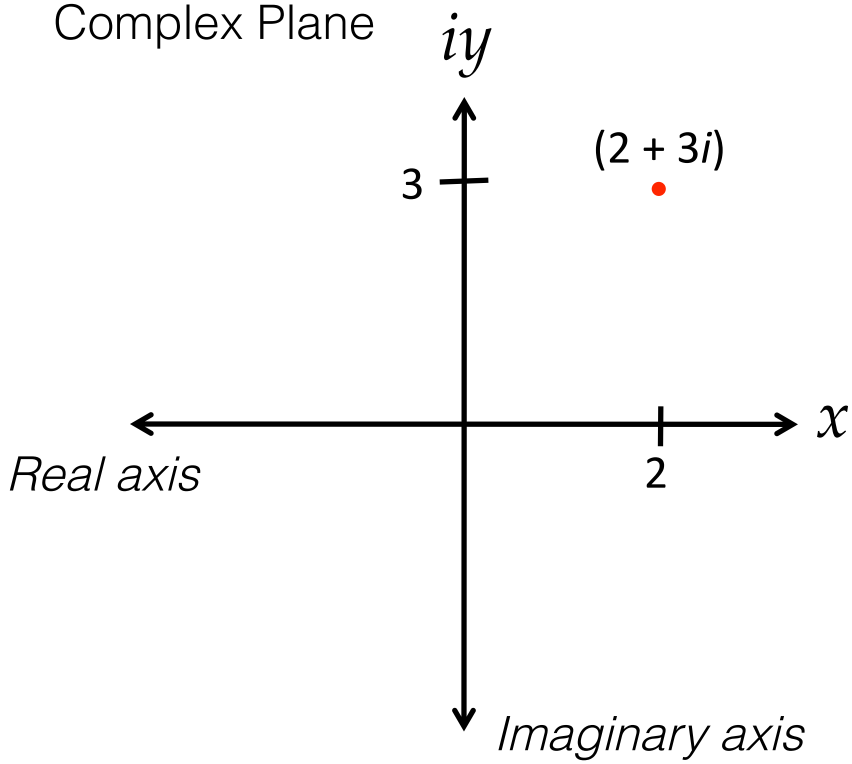
$0.4i$

…we must move to the *complex* plane!

i

Complex numbers

Complex Plane



$$z = x + iy$$

Real part
Re(z)

Imaginary part
Im(z)

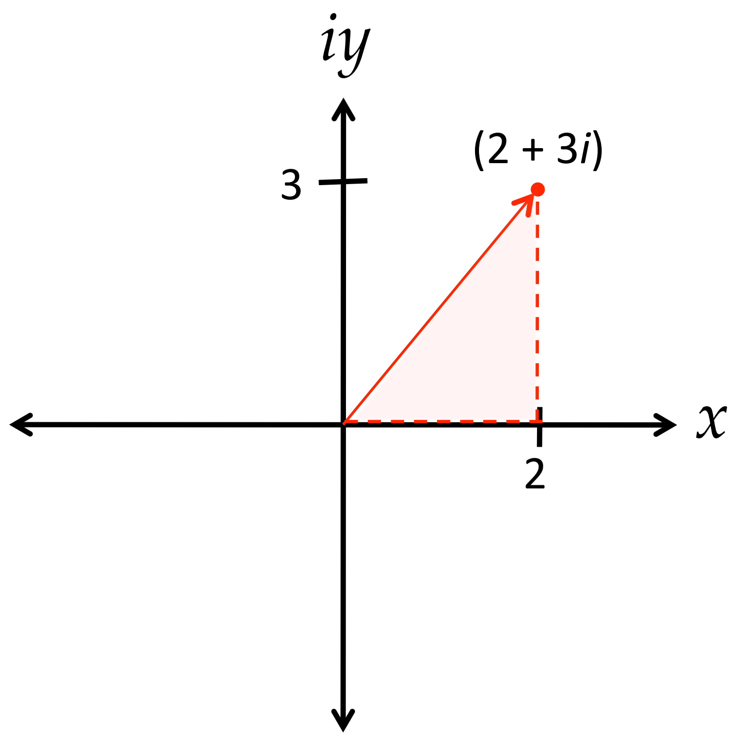
- Addition/subtraction
- Multiplication
- Magnitude

Complex numbers

$$z = x + iy$$

Real part
Re(z)

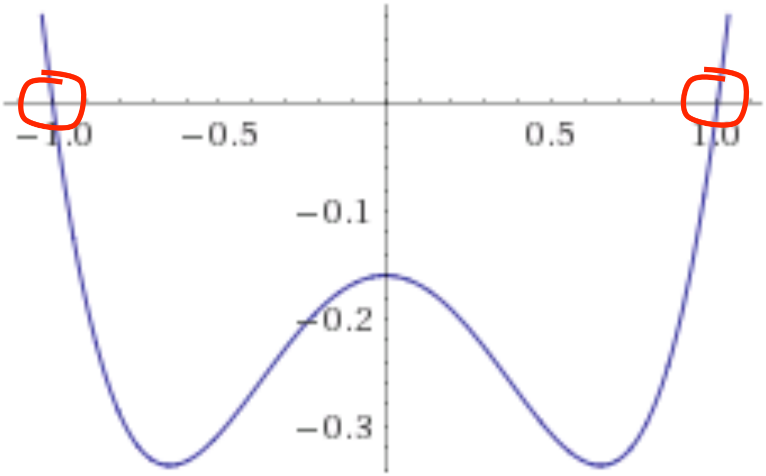
Imaginary part
Im(z)



- Addition/subtraction
- Multiplication
- Magnitude

$$|z| = \sqrt{2^2 + 3^2}$$

But there is something strange here…



2 visible roots

4 colors for 4 roots

Root 1: Yellow
Root 2: Red
Root 3: green
Root 4: blue

$z^4 - 0.84z^2 - 0.16$

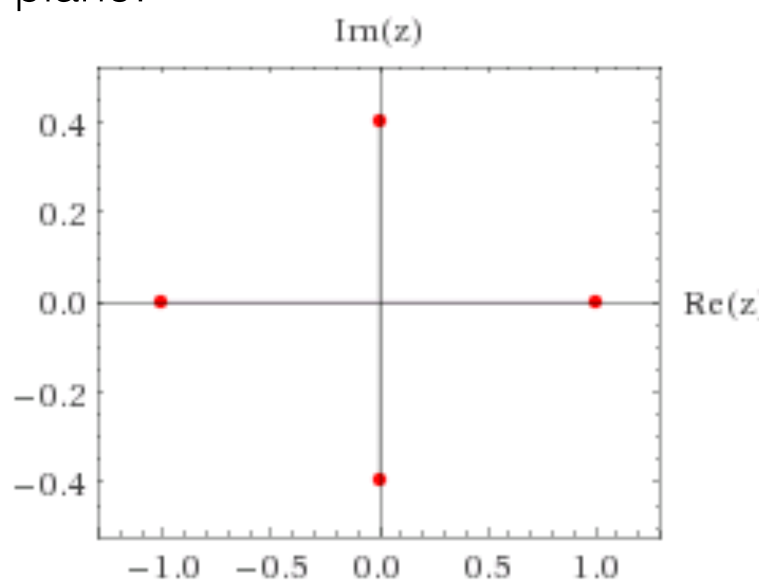
has roots:

-1
1
visible on real axes (left)

$-0.4i$

$0.4i$

…we must move to the *complex* plane!

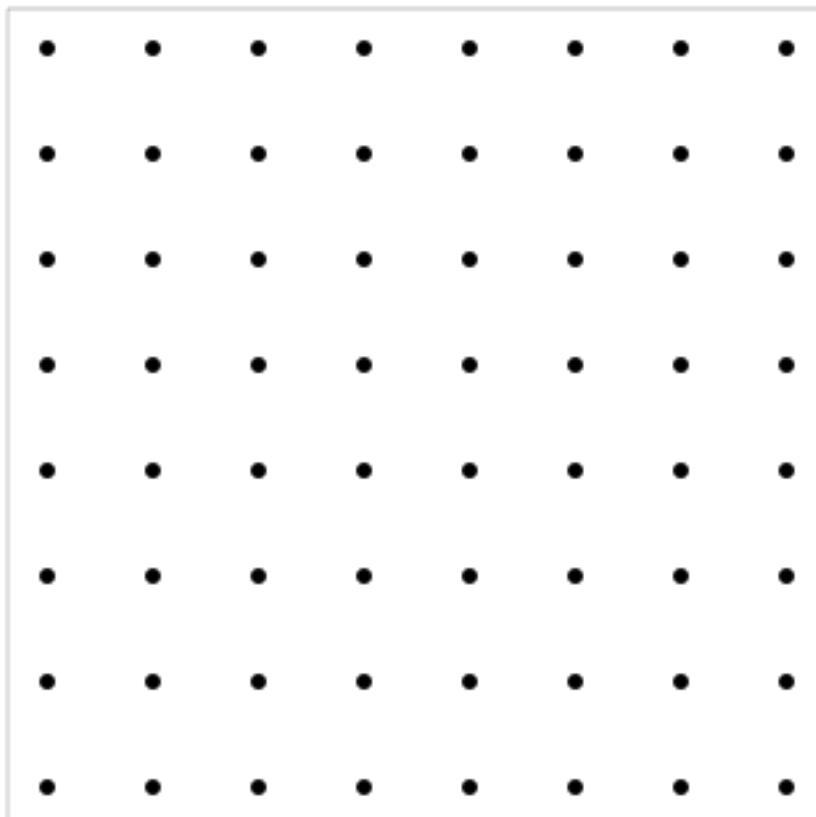


It would be extremely expensive to invoke a for loop and run through *every* point in the window.

In fact, unless we were working with a finite set of points, the loop would go on forever, and the picture would become more and more detailed

So we make a lattice that covers the plane and *perform Newton's method on all of the points at once!*

Then we have a huge collection (matrix) of the roots that Newton's method finds for each starting point in the lattice.



Your project: Outline

Driver

```
function qdrive
    *run qnewt on all 4 quartics*
return


function qnewt(q,xt,yt,maxiter)
    -create grid
    -run Newton's method on entire grid of points at once
    -find roots of polynomial and see which points end up at which root
    -color and plot points!
return


function d=myownpolyder(q)
    *Write your own derivative-taking function (one-liner)*
return
```