Draw fractals from root finding iteration in R

LA R users group: April Meeting

Keren Xu

PhD student in Epidemiology at USC

2020/04/23

Overview

1. Root-finding Algorithm
2. Newton's Method
3. Secant Method
4. Fractals
5. Getting Started - Creating your fractals

Root-finding Algorithm

$f(a) = 0$

$f(x) = g(x)$ ➡️ $h(x) = f(x) - g(x)$

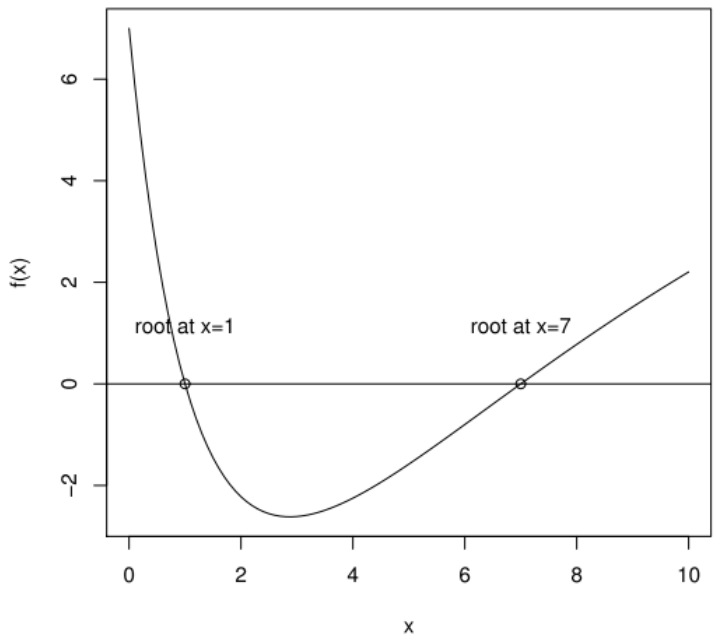


Figure 10.1 *The roots of the function f.*

Root-finding Algorithm

- Requires one or more **initial guesses** of the root
- We need to set a maximum number of iterations
- The algorithm converges when it finds a sufficiently accurate guess of the root within the maximum number of iterations

Newton's Method

Our curent "guess" for a: x_0 ➡️ $f'(x_0) = \frac{f(x_0)-y}{x_0-x}$
➡️ $f'(x_0) = \frac{f(x_0)-0}{x_0-x_1}$ ➡️ $x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$ ➡️ $x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$

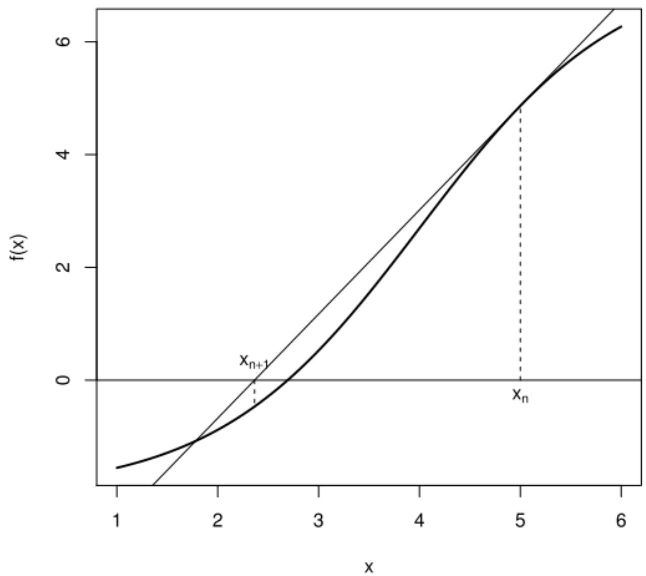
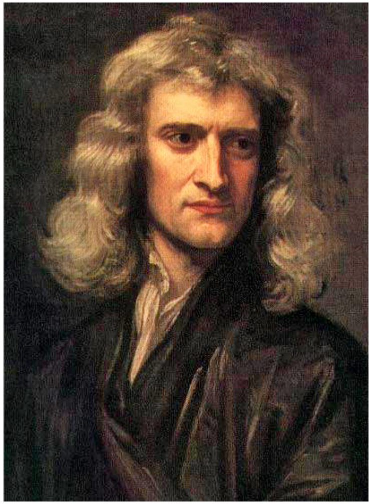
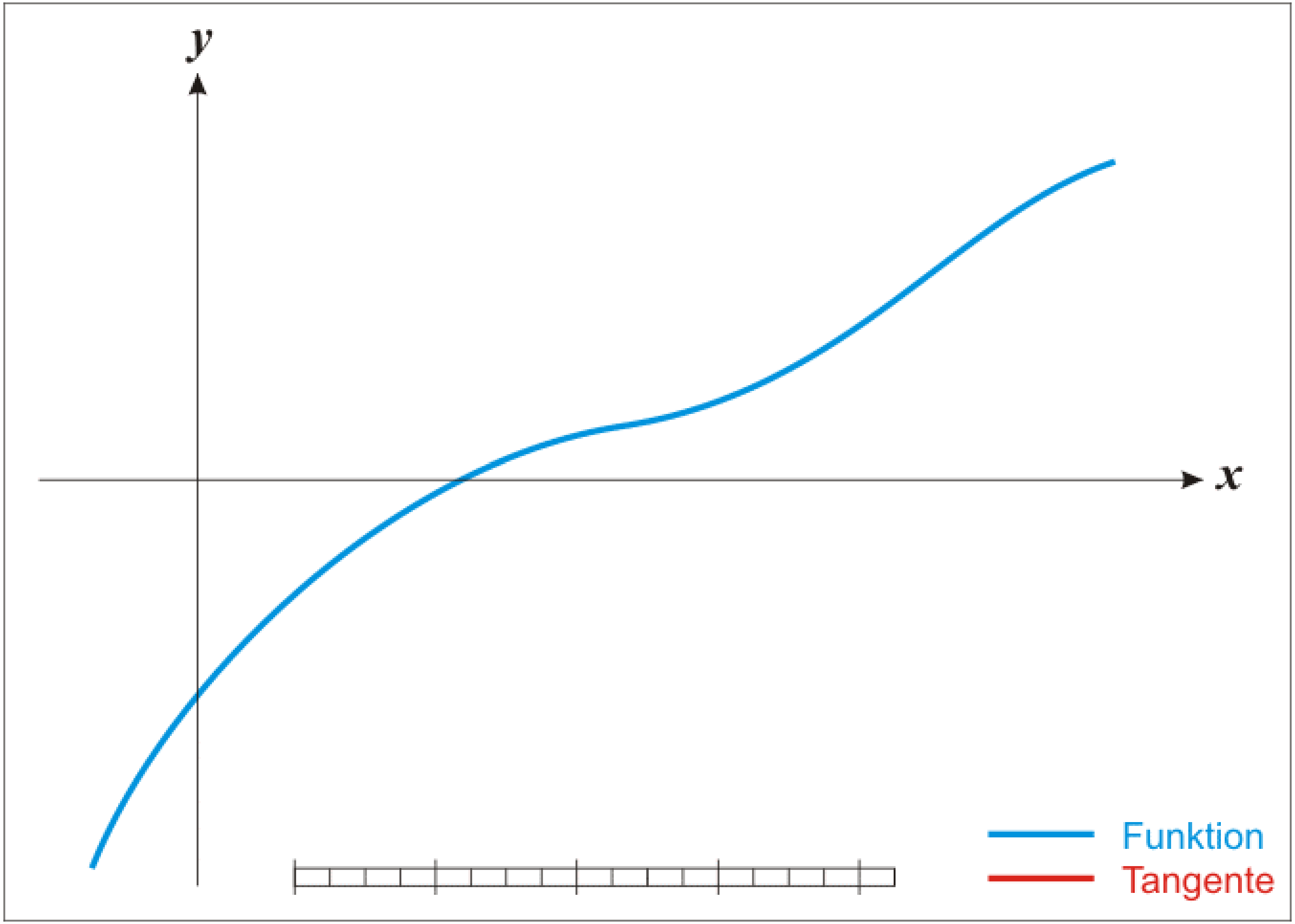


Figure 10.3 *A step in the Newton–Raphson root-finding method.*



Newton's Method



Newton's Method

In general:

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

Newton's Method

Computational example:

Find roots for $x^3 - 1$

```
F1 <- function(x){
  return(c(x^3-1, 3*(x^2)))
}
```

Newton's Method

Computational example:

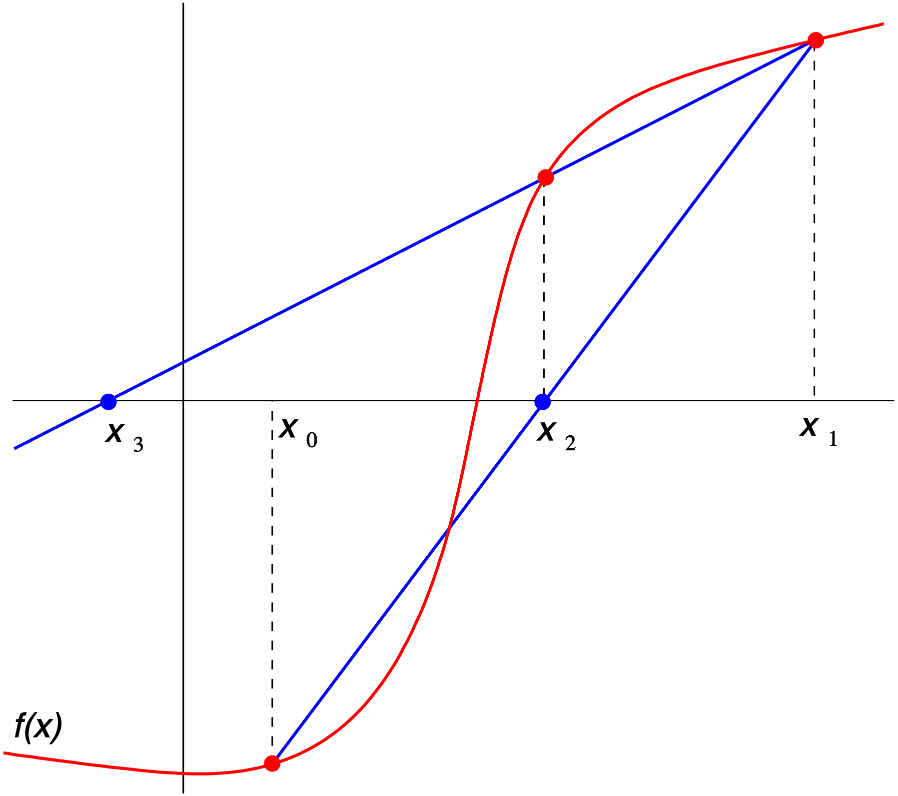
```
newtonraphson <-
  function(ftn, x0, tol = 1e-9, max.iter) {
  # initialize
  x <- x0
  fx <- ftn(x)
  iter <- 0

  # continue iterating until stopping conditions are met
  while((abs(fx[1]) > tol) && (iter < max.iter)) {
    x <- x - fx[1]/fx[2]
    fx <- ftn(x)
    iter <- iter + 1
    cat("At iteration", iter, "value of x is:", x, "\n")
  }

  # output depends on the success of the algorithm
  if (abs(fx[1]) > tol){
    cat("Algorithm failed to converge\n")
    return(data.frame(x0, root = NA, iter = NA))
  } else {
    cat("Algorithm converged\n")
    return(data.frame(x0, root = x, iter))
  }
```

Secant Method

- Do not need to compute a derivative
- Need to provide two initial guesses

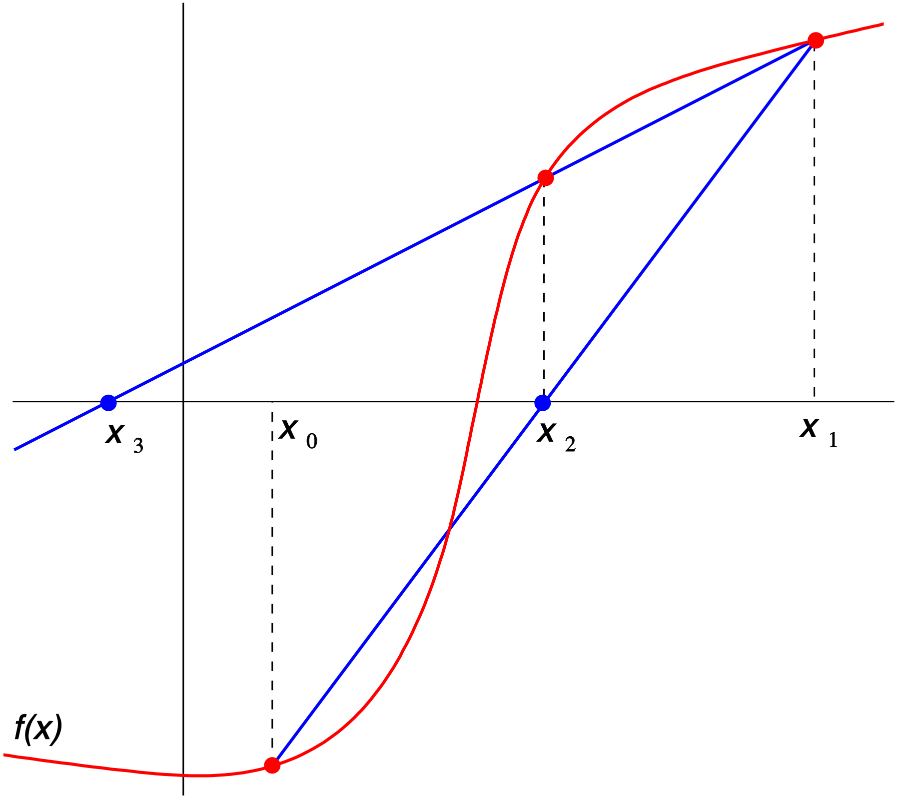


Secant Method

Two initial "guesses" x_0 and x_1, assuming x_0 is the older one

➡️ $\frac{y-f(x_1)}{x-x_1} = \frac{f(x_0)-f(x_1)}{x_0-x_1}$

➡️ so x_2 can be found from $\frac{0-f(x_1)}{x_2-x_1} = \frac{f(x_0)-f(x_1)}{x_0-x_1}$ ➡️ $x_2 = x_1 - f(x_1)\frac{x_0-x_1}{f(x_0)-f(x_1)}$



Secant Method

In general:

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

Secant Method

Computational example:

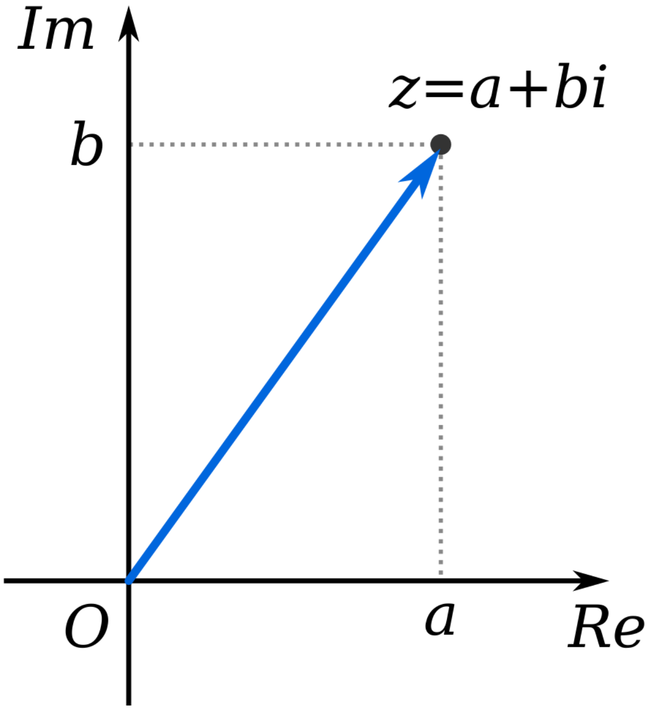
```r
secant <- function(ftn, x0, x1, tol = 1e-9, max.iter) {
  # initialize
  x_n0 <- x0
  x_n1 <- x1
  ftn_n0 <- ftn(x_n0)
  ftn_n1 <- ftn(x_n1)
  iter <- 0

  # continue iterating until stopping conditions are met
  while((abs(ftn_n1) > tol) && (iter < max.iter)) {
    x_n2 <- x_n1 - ftn_n1*(x_n1 - x_n0)/(ftn_n1 - ftn_n0)
    x_n0 <- x_n1
    ftn_n0 <- ftn(x_n0)
    x_n1 <- x_n2
    ftn_n1 <- ftn(x_n1)
    iter <- iter + 1
    cat("At iteration", iter, "value of x is:", x_n1, "\n")
  }

  return(c(x_n1, iter))
}
```

Fractals

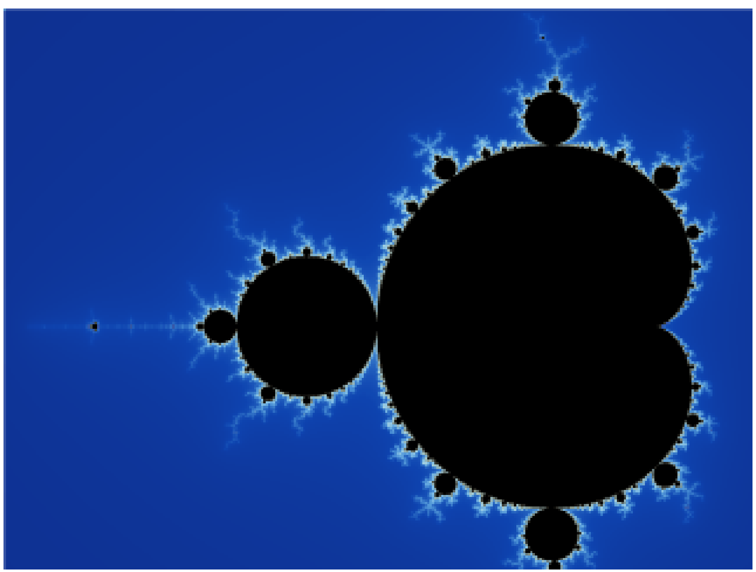
Root finding functions can also be applied to find roots for **complex functions**, which are functions of **complex numbers**.



Fractals

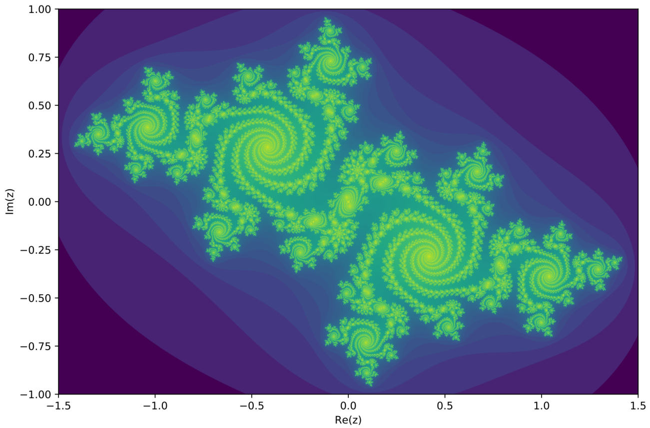
Each root has a **basin of attraction** in the complex plane, which is a set of all **initial guesses** that cause the method to converge to that particular root.

These **initial guesses** can be mapped into images. The boundaries of the basins of attraction are called **fractals**.



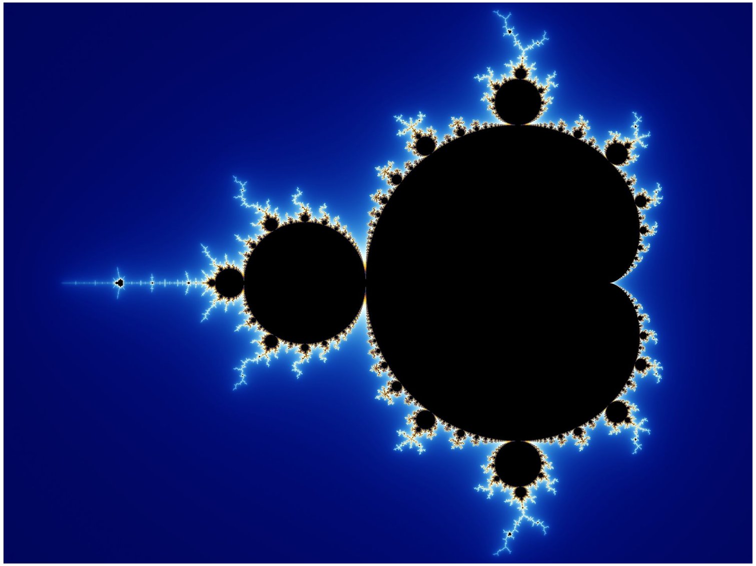
Fractals

Julia set



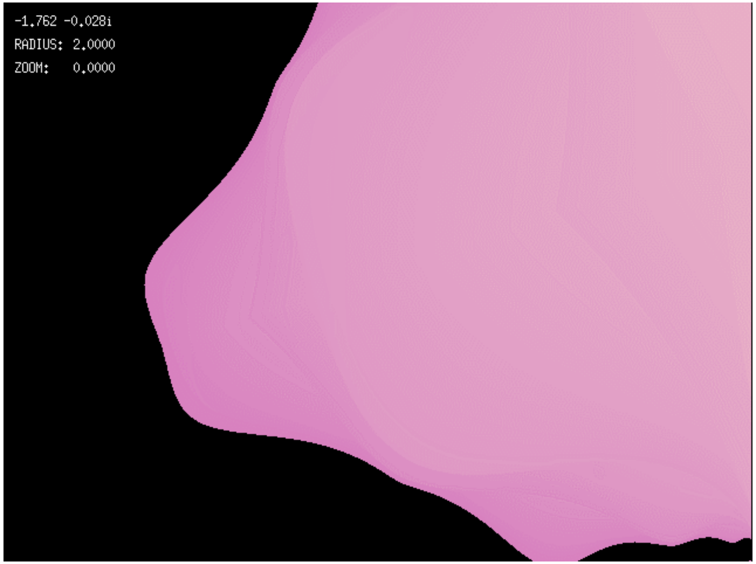
Fractals

Mandelbrot set



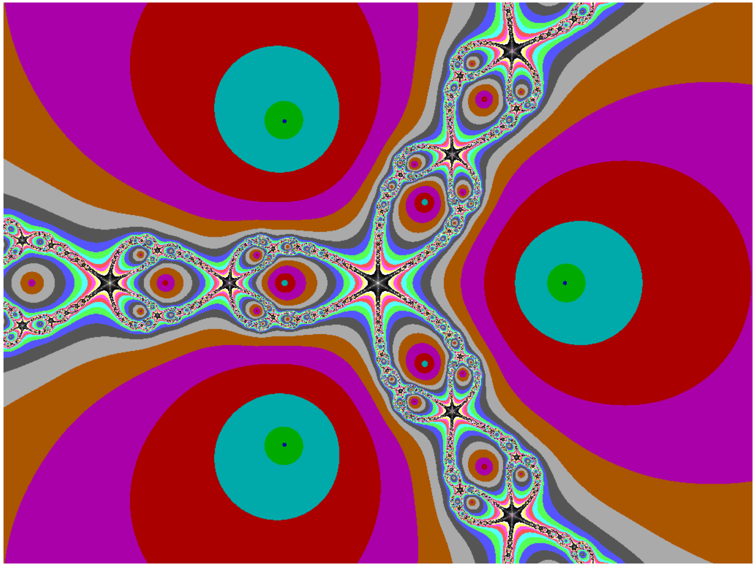
Fractals

Burning ship fractals



Fractals

Newton's fractals



A zoom in video with music 📽️🎹

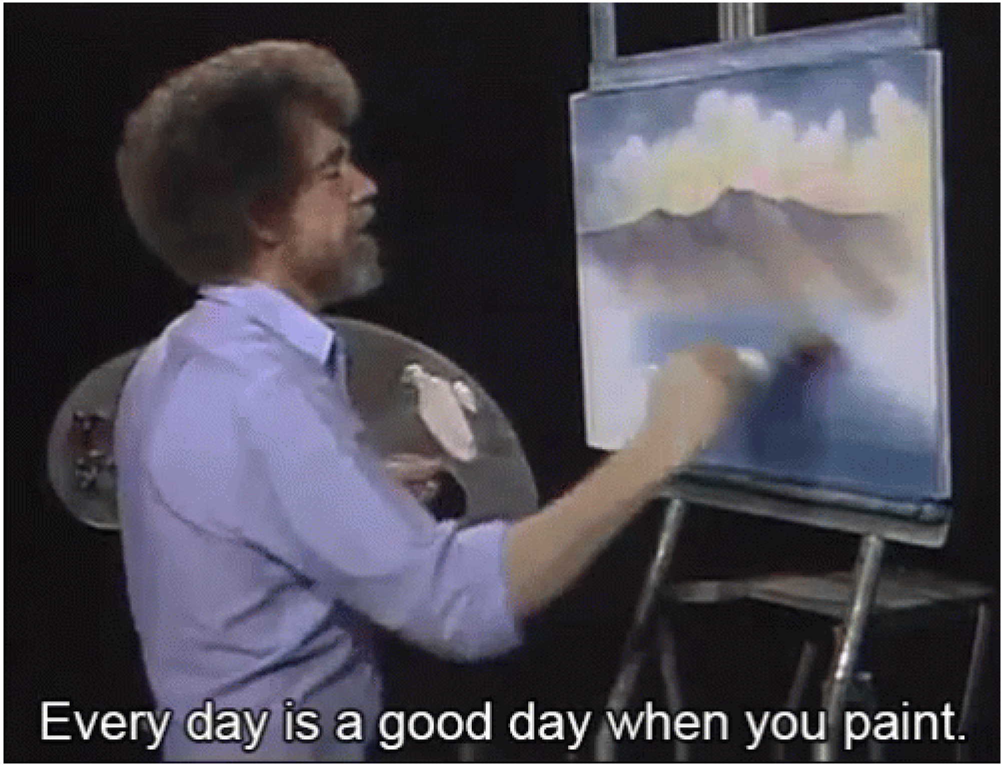
Summary

1. Root-finding algorithm: $h(x) = f(x) - g(x) = 0$ 📐

2. Most common: Newton's Method and Secant method 🍎

3. Complex numbers and fractals 👨‍🎨

Get Started - Creating your fractals



Recommended readings

- Introduction to scientific programming and simulation using r, by Andrew P. Robinson, Owen Jones, and Robert Maillardet [link]
- Newton fractal wiki page [link]
- Blog Fronkonstin [link]
- My repo [link]

Thanks for attending!

😄

Keep in touch twitter @kerenxuepi

Slides created via **RLadies xaringan** slide theme