Mandelbrot and Julia set plotting in R

Phileas Dazeley Gaist

2/26/2022

Setup

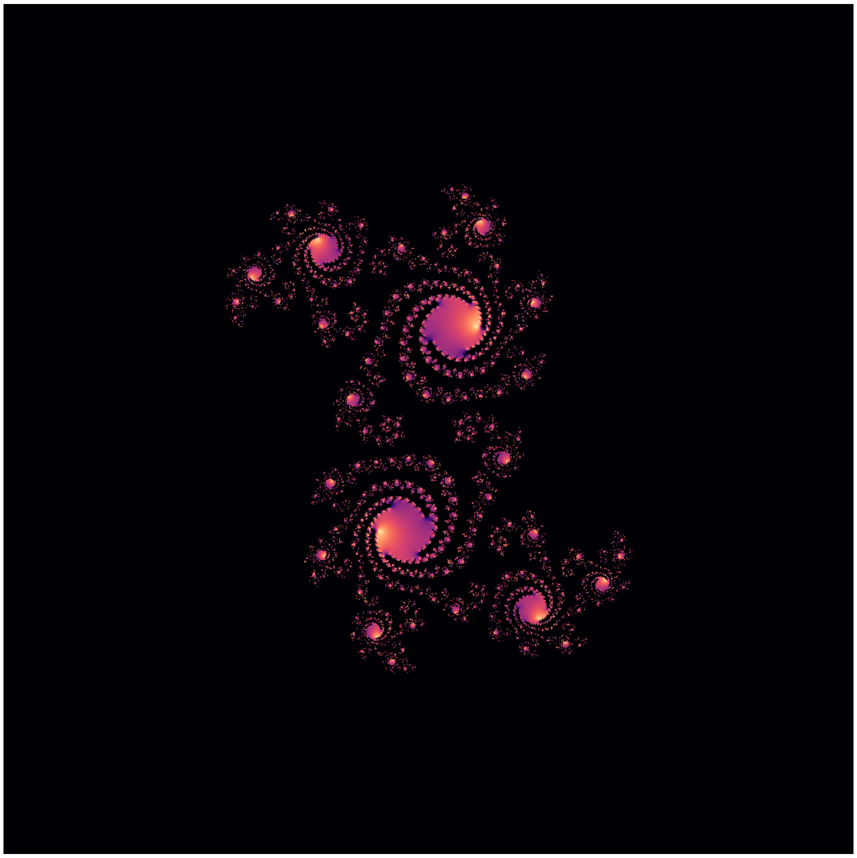
Julia sets

```
# let's define a sample value of c
c <-  0.355+0.355i
```

```
# let's define our Julia formula
julia <- function(z, c){
  return(z^2+c)
}
```

```
# let's define a maximum number of steps to decide if an orbit blows up to infinity or not
max_iter <- 100
# now let's define the limits and detail of our plot

# x and y limits (corresponding to real and imaginary)
x <- c(-2, 2); y <- c(-2, 2)
# horizontal and vertical resolution (the number of data points between limits)
h_res <- 1000; v_res <- 1000

real_line <- seq(x[1], x[2], length=h_res)
imaginary_line <- seq(y[1], y[2], length=v_res)*1i

space_grid <- outer(real_line,imaginary_line,"+") %>% c()

z <- space_grid
for(i in 1:max_iter){
  # apply the Julia function to every z_0 value
  z <- julia(z, c)
}

julia_data <- tibble(r=Re(space_grid),
                     i=Im(space_grid),
                     z=as.vector(exp(-Mod(z)))) %>% na.omit()
```

```
# plot

julia_data %>% ggplot(aes(r, i, fill=z)) +
  geom_raster(interpolate = T) +
  scale_x_continuous(expand=c(0,0)) +
  scale_y_continuous(expand=c(0,0)) +
  scale_fill_viridis(option="magma") +
  clear_theme +
  coord_equal()
```



Mandelbrot set

```
x <- c(-2.2, 1); y <- c(-1.1, 1.1)
h_res <- 1000; v_res <- 1000

max_iter <- 35

real_line <- seq(x[1], x[2], length=h_res)
imaginary_line <- seq(y[1], y[2], length=v_res)*1i
```

```
space_grid <- outer(real_line,imaginary_line,"+") %>% c()

z <- 0
cs <- space_grid # all c values
for(i in 1:max_iter){
  z <- z^2 + cs
}

mandelbrot_data <- tibble(r=Re(space_grid),
                          i=Im(space_grid),
                          z=as.vector(exp(-Mod(z)))) %>% na.omit()



# plot
mandelbrot_data %>% ggplot(aes(r, i, fill=z)) +
  geom_raster(interpolate = F) +
  scale_x_continuous(expand=c(0,0)) +
  scale_y_continuous(expand=c(0,0)) +
  scale_fill_viridis(option="magma") +
  clear_theme +
  geom_point(aes(Re(c),Im(c)), shape=8, colour="white", size=8) + # add a point for the julia
  coord_equal()
```