

# 1 Lecture 1: Complex analysis review

The first couple lectures will review the basics of complex analysis. We will use plots to help explain the material, and so this first lecture will focus on getting comfortable with plots of analytic functions.

## 1.1 Plotting functions in the complex plane

Consider a complex-valued function  $f : D \rightarrow \mathbb{C}$  where  $D \subset \mathbb{C}$ . To help understand such functions it is useful to plot them. But how?

### Method 1: real and imaginary parts

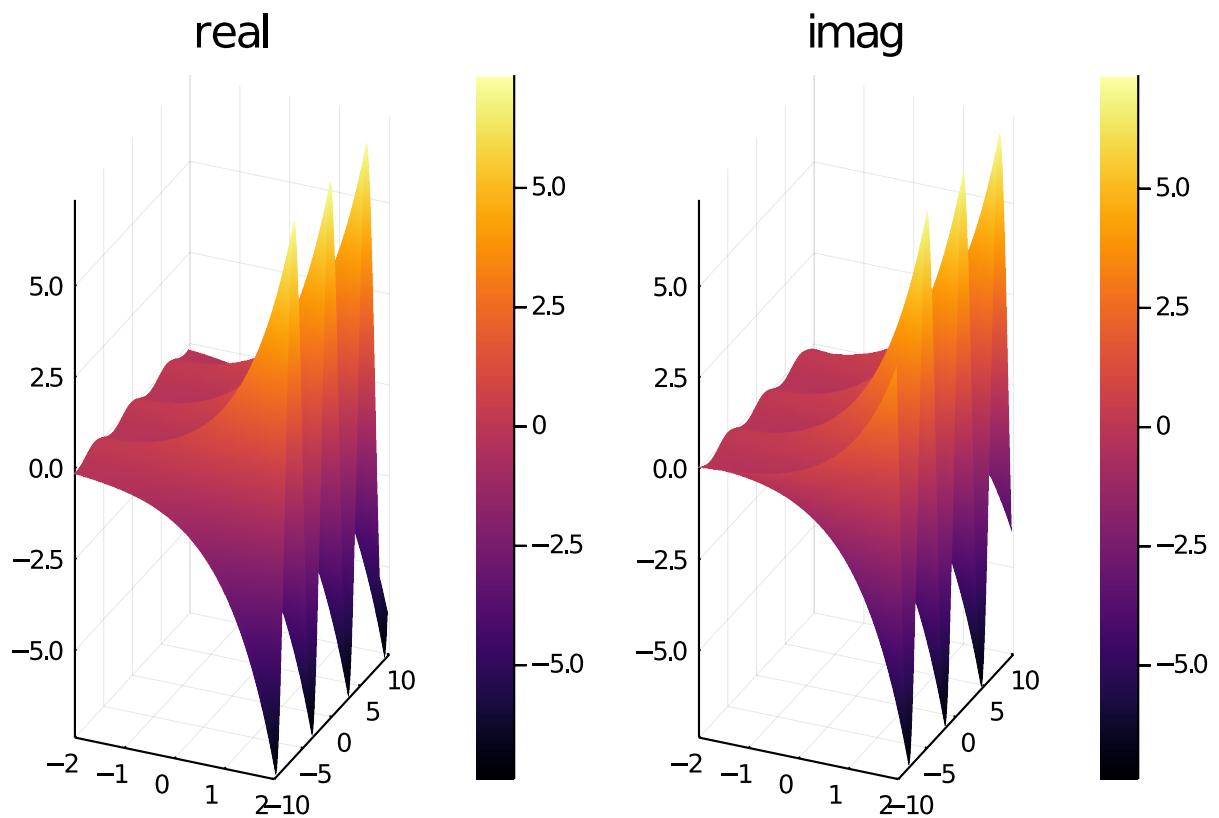
Every complex-valued function can be written as  $f(z) = u(z) + iv(z)$  where  $u : D \rightarrow \mathbb{R}$  and  $v : D \rightarrow \mathbb{R}$  are real-valued. These are easy to plot separately. We can do a surface plot:

```
using Plots, ComplexPhasePortrait, SpecialFunctions

f = z -> exp(z)
u = z -> real(f(z))
v = z -> imag(f(z))

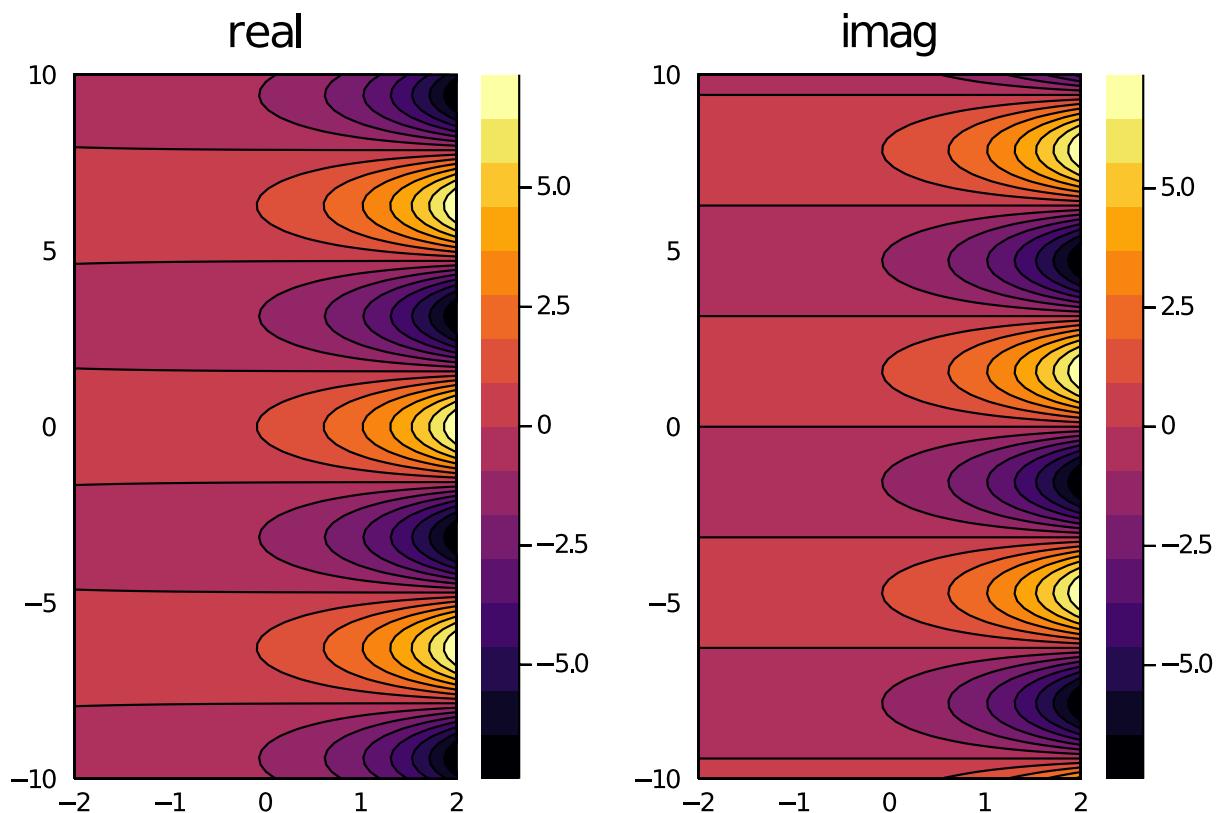
# set up plotting grid
xx = range(-2 ; stop=2, length=100)
yy = range(-10; stop=10, length=100)

plot(surface(xx, yy, u.(xx' .+ im.*yy); title="real"),
      surface(xx, yy, v.(xx' .+ im.*yy); title="imag"))
```



Or a heat plot:

```
plot(contourf(xx, yy, u.(xx' .+ im.*yy); title="real"),
     contourf(xx, yy, v.(xx' .+ im.*yy); title="imag"))
```



Method 2: absolute-value and angle

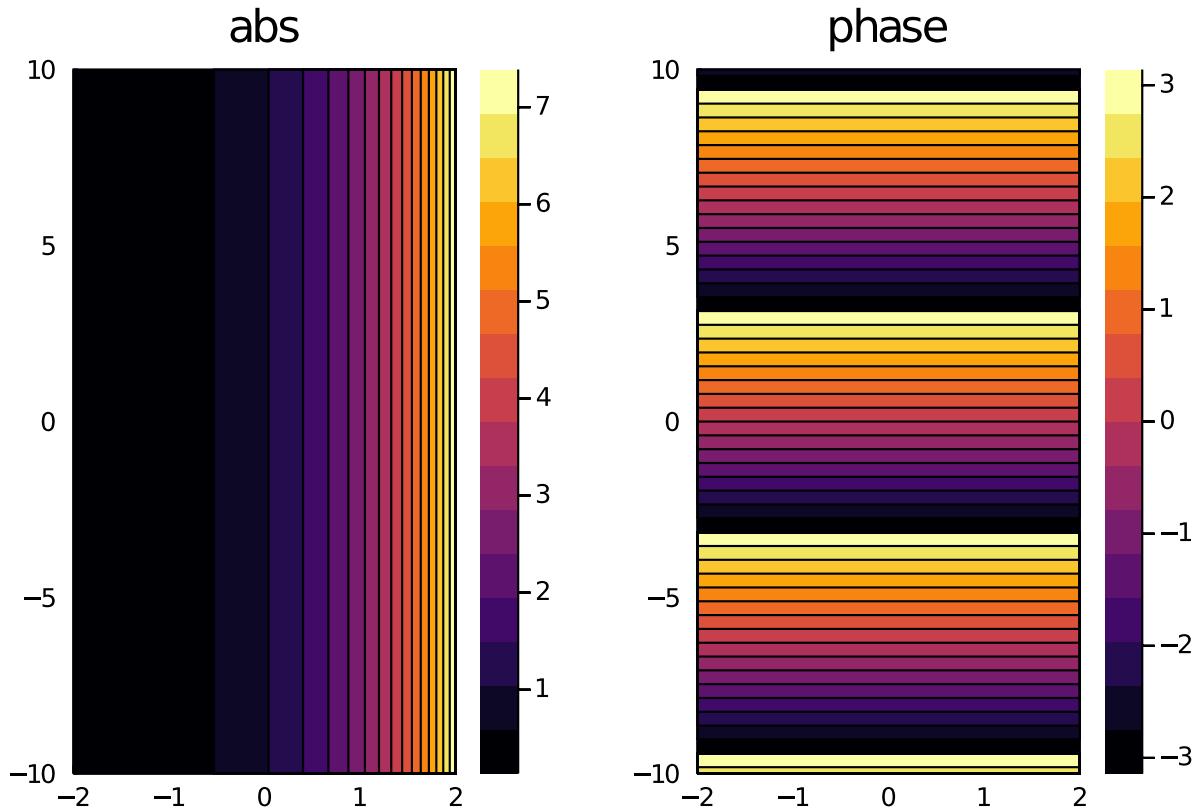
Every complex number  $z$  can be written as  $re^{i\theta}$  for  $0 \leq r$  and  $-\pi < \theta \leq \pi$ .  $r = |z|$  is called the *absolute value* and  $\theta$  is called the *phase* or *argument* (or in Julia, *angle*). We can plot these:

```
xx = -2:0.01:2
yy = -10:0.01:10

f = z -> exp(z)

r = z -> abs(f(z))
θ = z -> angle(f(z))

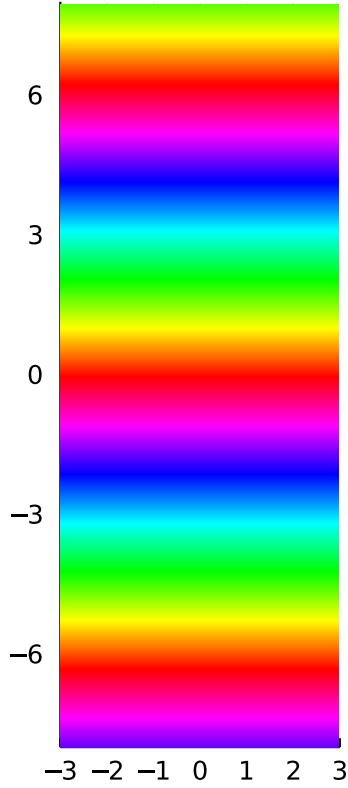
plot( contourf(xx, yy, r.(xx' .+ im.*yy); title="abs"),
      contourf(xx, yy, θ.(xx' .+ im.*yy); title="phase"))
```



### Method 3: Phase portrait

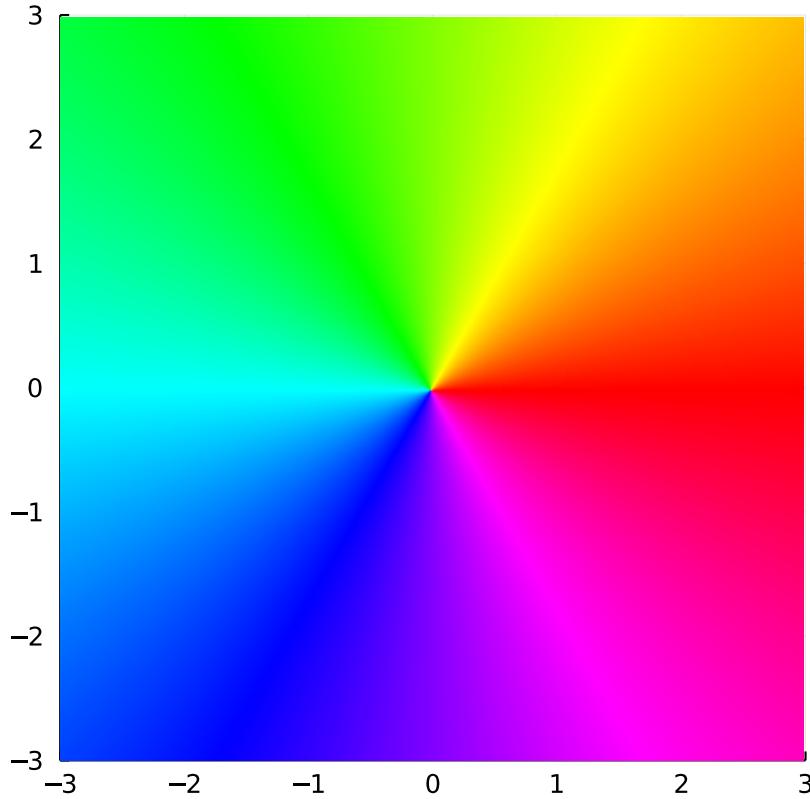
This method is essentially the same, as before, but it *only* plots the phase, and uses a "colour wheel" to reflect the topology of  $\theta$ .

```
phaseplot(-3..3, -8..8, z -> exp(z))
```



This is seen most clearly with  $f(z) = z$ : in this case, letting  $z = re^{i\theta}$  we are simply plotting  $\theta$ :

```
phaseplot(-3..3, -3..3, z -> z)
```

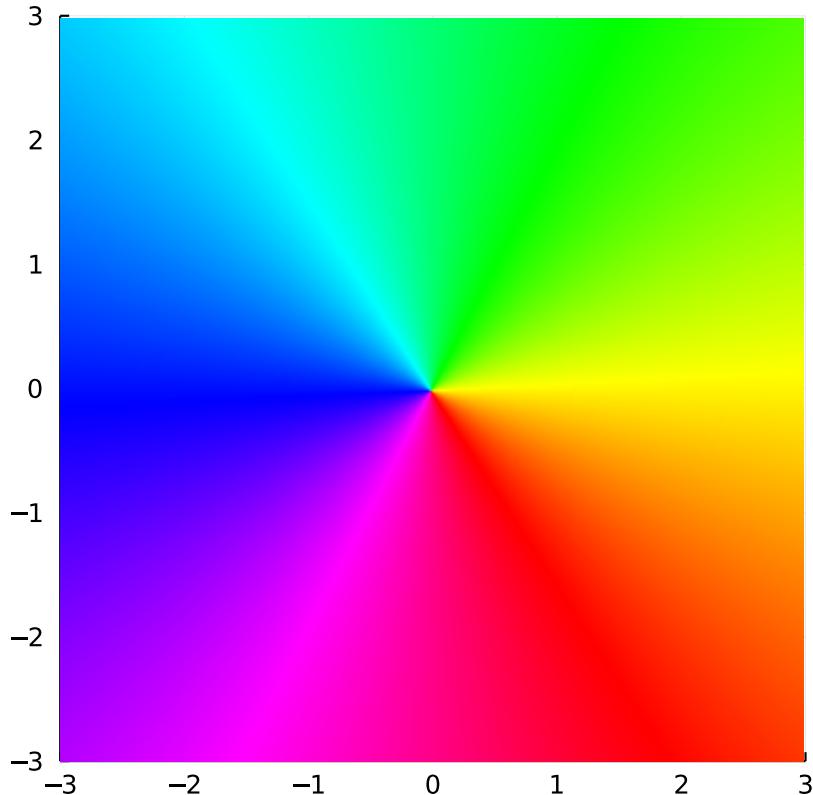


In other words, the colour red corresponds to  $\arg f(z) \approx 0$ , green to  $\arg f(z) \approx \frac{2\pi}{3}$  aqua to

$\arg f(z) \approx \pi$  and so on.

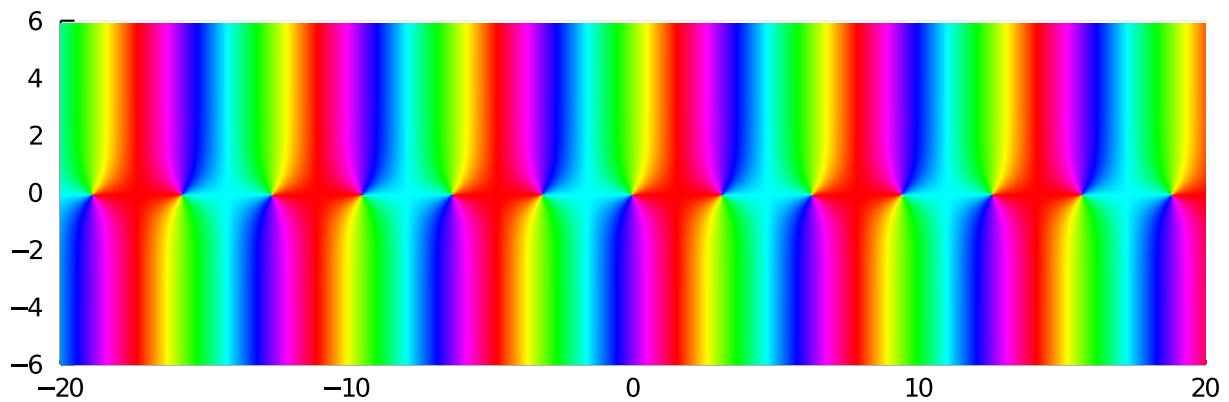
Note that multiplying  $z$  by a complex number  $Re^{i\varphi}$  will rotate the wheel, but the colours still appear in the same order when read anticlockwise:

```
phaseplot(-3..3, -3..3, z -> exp(1.0im)*z)
```



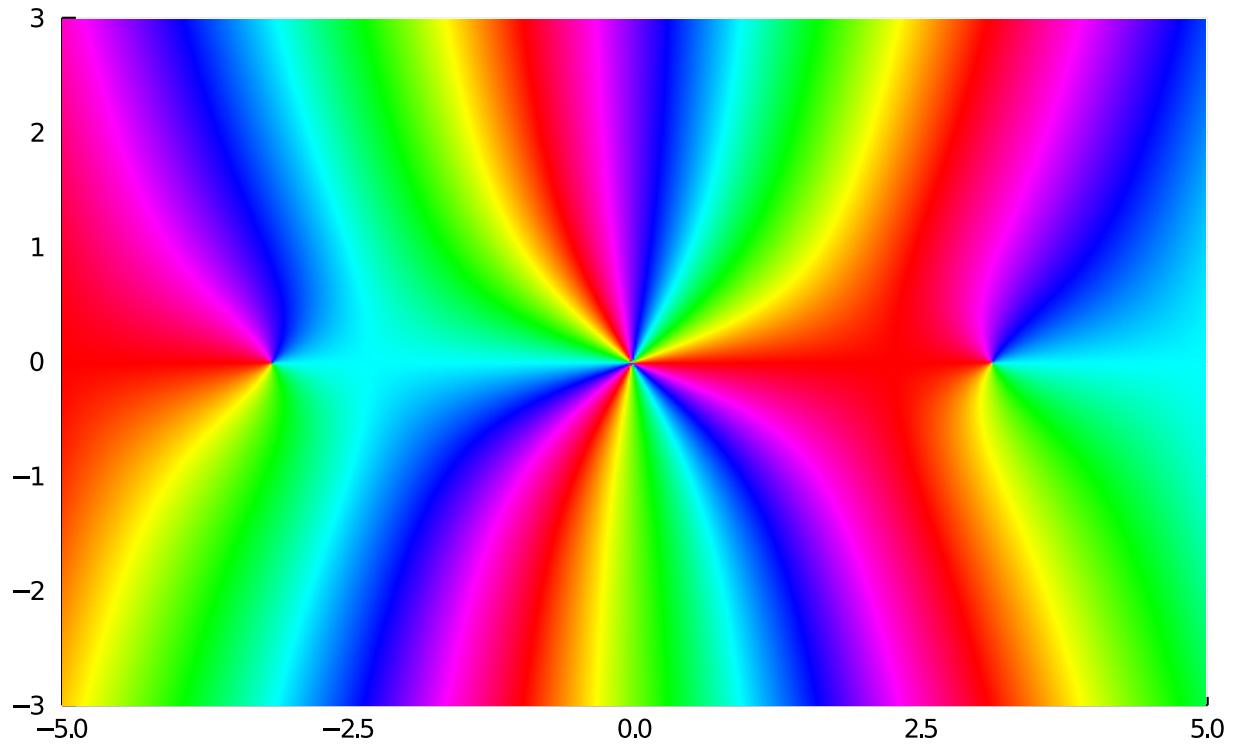
Therefore, if  $f(z)$  has a zero at  $z_0$ , since it behaves like  $f(z) = f'(z_0)(z - z_0)$ , we will have the full colour wheel always in the anticlockwise order red-green-blue-red:

```
phaseplot(-20..20, -6..6, z -> sin(z))
```



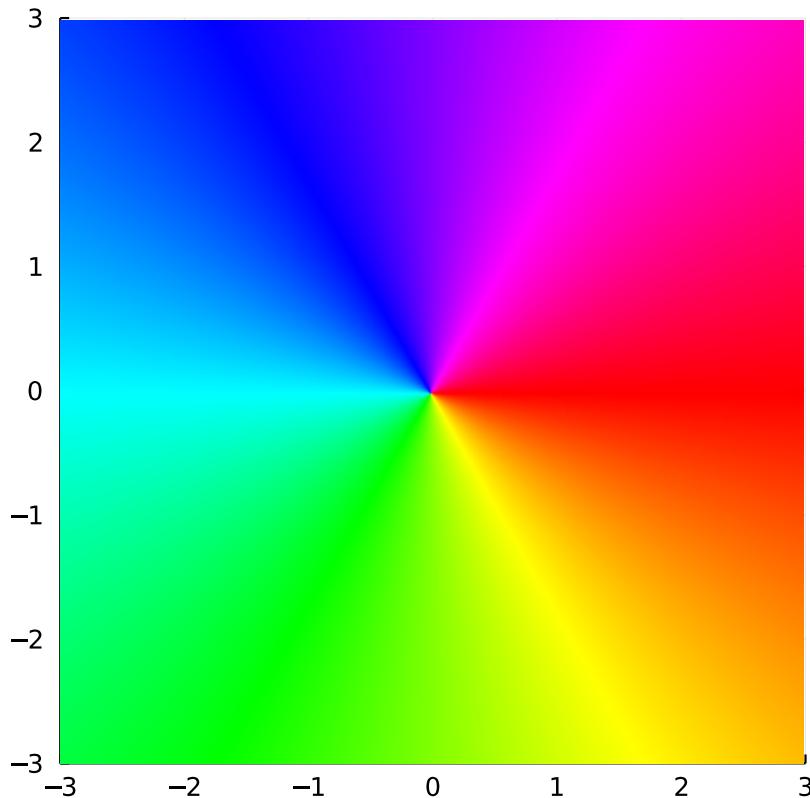
In the case of a double root like  $f(z) = z^2 = re^{2i\theta}$ , we have the wheel appearing twice, but still in the same order. Thus the order of a zero can be seen by the number of times we go around the colour wheel: here we see that the function has a triple root at zero since it goes red-green-blue-red-green-blue-red-green-blue-red:

```
phaseplot(-5..5, -3..3, z -> z^2*sin(z))
```



On the other hand, if we plot  $f(z) = z^{-1} = r^{-1}e^{-i\theta}$  the wheel is reversed to be red-blue-green-red:

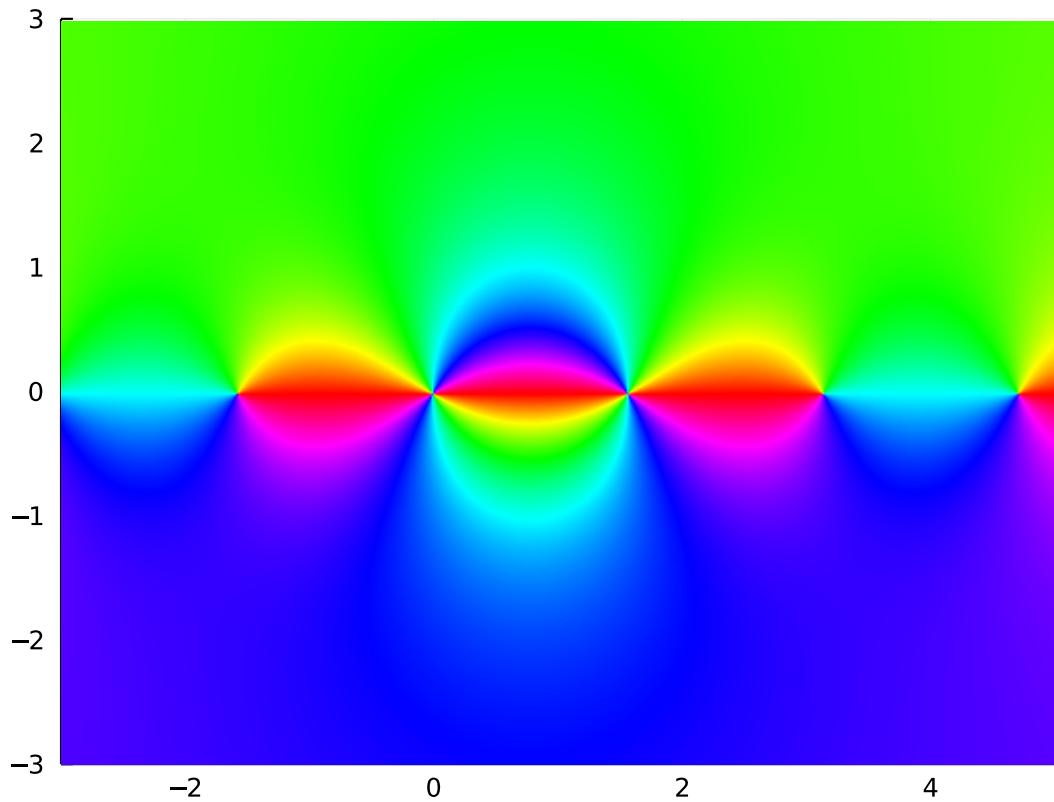
```
phaseplot(-3..3, -3..3, z -> 1/z)
```



Thus we can see from a phase plot where the poles and zeros are, and the order of the poles

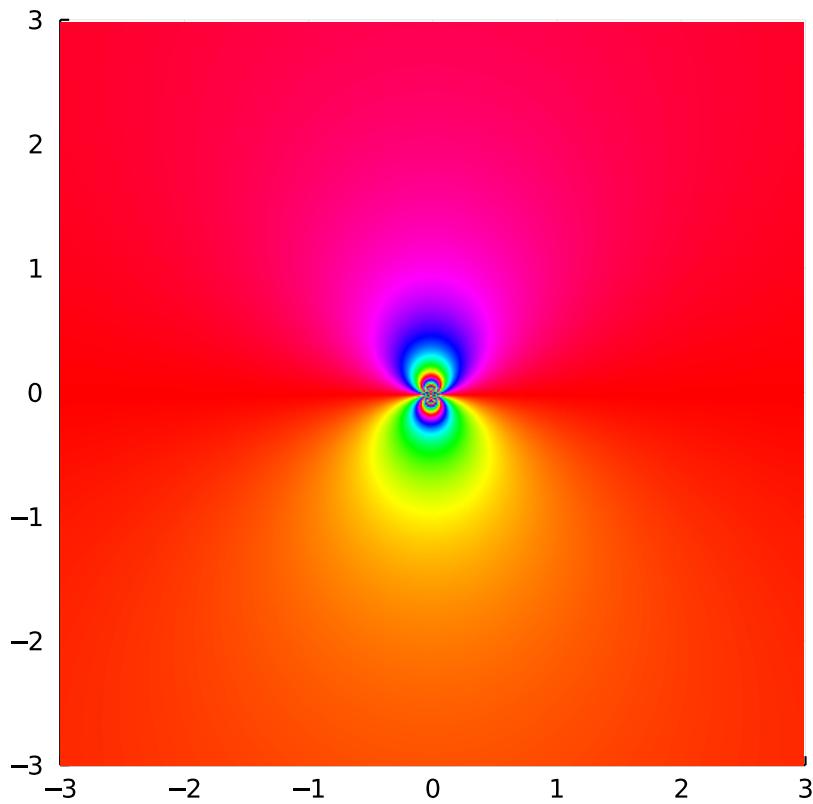
and roots: the following function has a double pole at 0 (red-blue-green-red-blue-green), a zero at  $-\frac{\pi}{2}$  (red-green-blue-red) and a double zero at  $\frac{\pi}{2}$  (red-green-blue-red-green-blue-red):

```
phaseplot( -3..5, -3..3, z -> cot(z)/z*(π/2-z))
```



Functions with more complicated singularities do not have such nice phase plots (these are what we will call *essential* singularities):

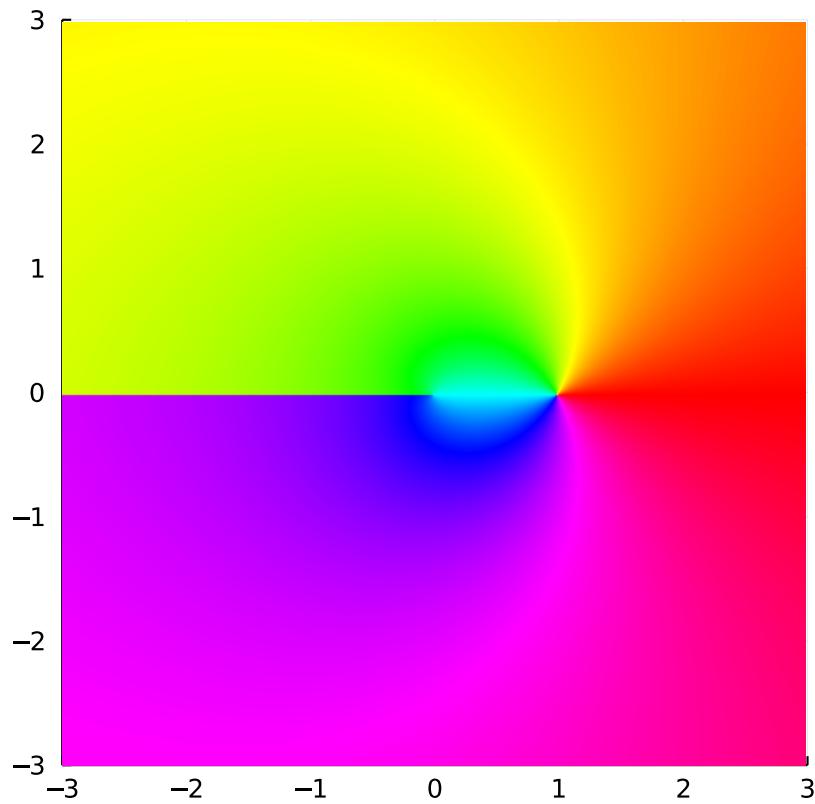
```
phaseplot(-3..3, -3..3, z -> exp(1/z))
```



Picard's great theorem: Within any arbitrarily small neighbourhood of an essential singularity of a function  $f(z)$ ,  $f(z)$  takes on every complex value, with at most one exception, infinitely often.

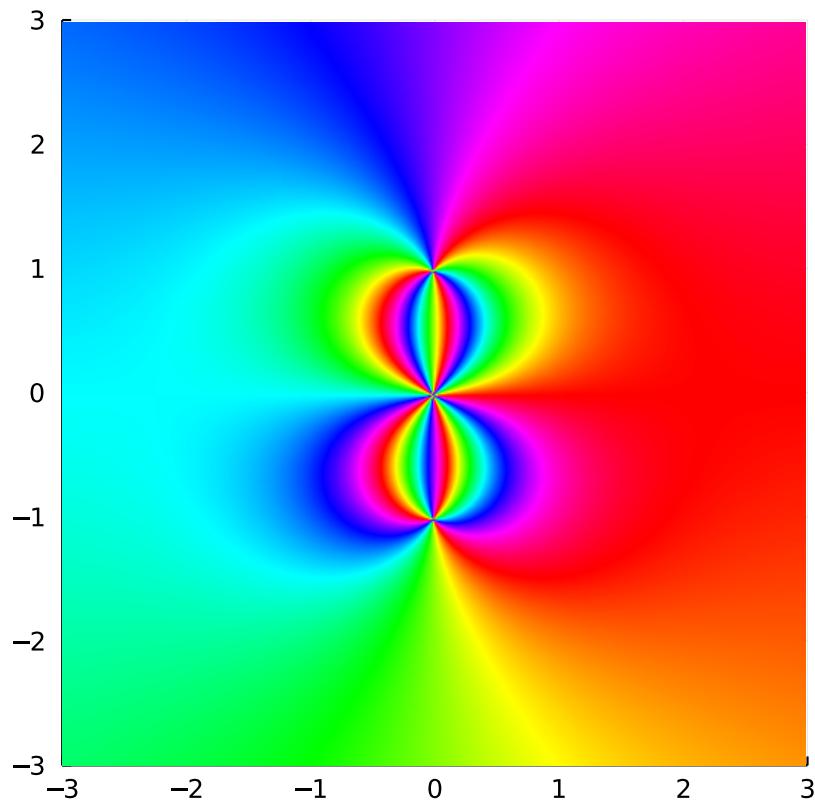
The jumps of functions like  $\log z$  or  $\sqrt{z}$  also appear naturally in the phase plot: here we see  $\log z$  has a zero at 1 (red-green-blue-red) and a jump along  $(-\infty, 0]$

```
phaseplot( -3..3, -3..3, z -> log(z))
```

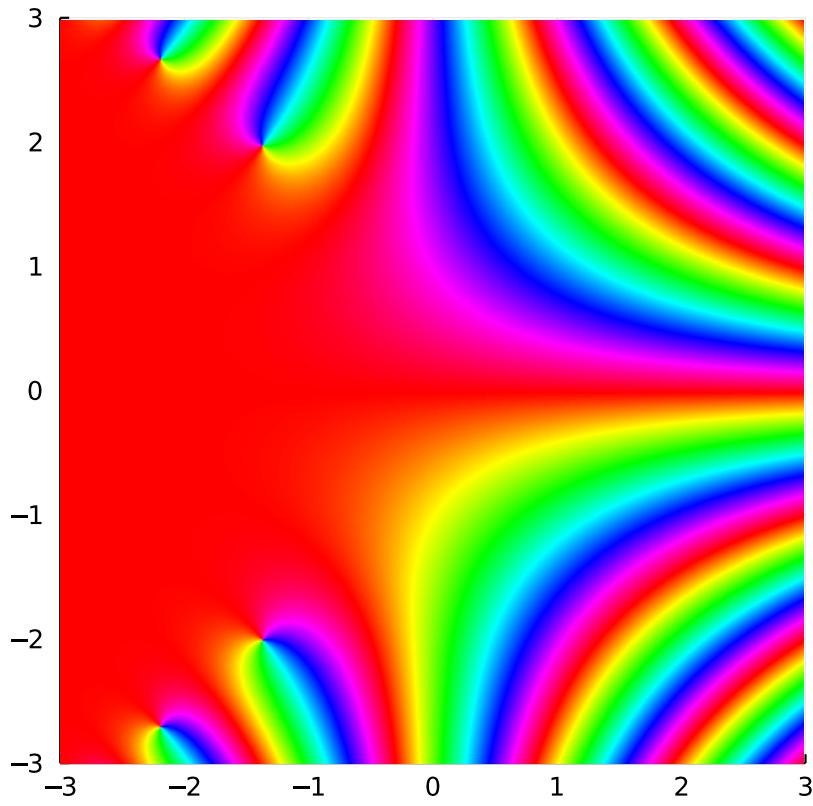


Often, singularities are in the complex plane. Can you determine where the zeros and roots, and their orders, of the following function are from the picture?

```
phaseplot(-3..3, -3..3, z -> z^5 / (1 + z^2)^3)
```

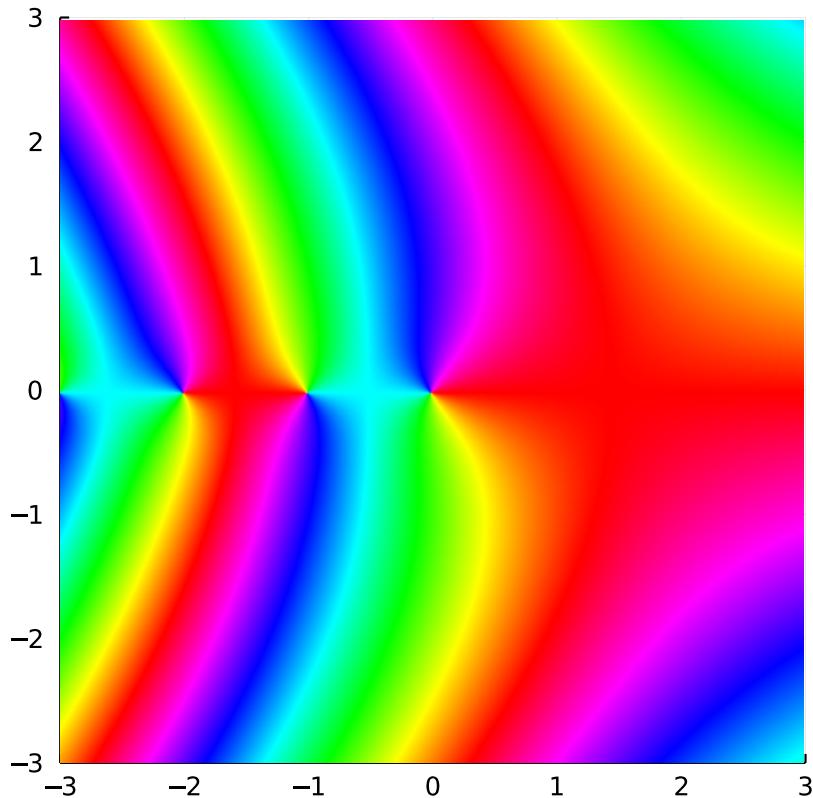


```
phaseplot(-3..3, -3..3, z -> erfc(z))
```



Here's the Gamma function. Are they poles or zeros at  $0, -1, -2, \dots$ ?

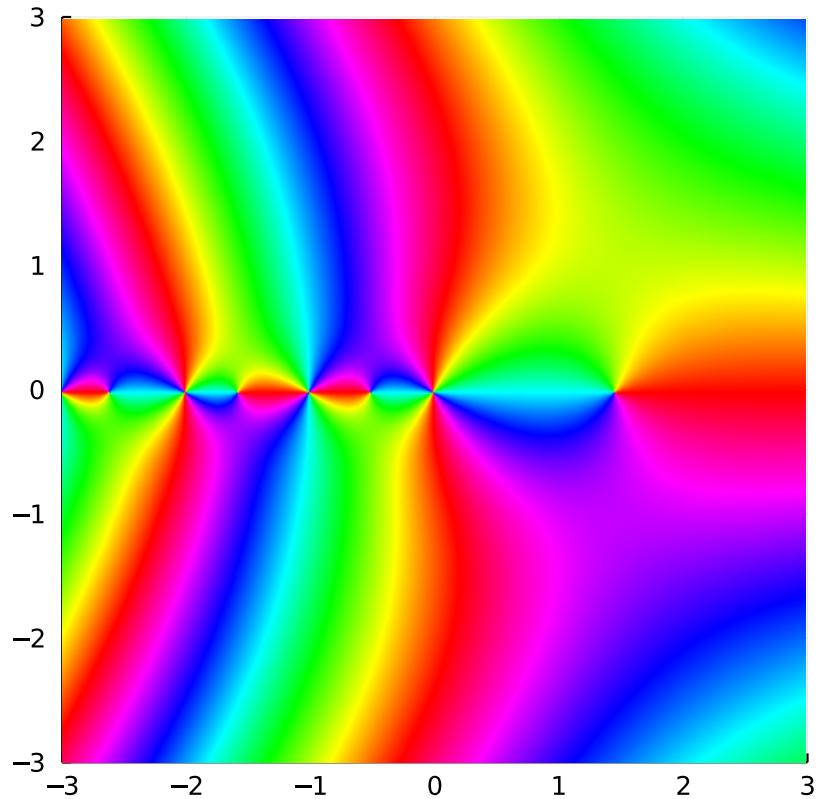
```
phaseplot(-3..3, -3..3, z -> gamma(z))
```



Something strange is happening at  $-1/2, -3/2, \dots$ . What if we differentiate? Here `dual(z, 1)` is a [dual number](#), which provides a convenient way to implement automatic differentiation:

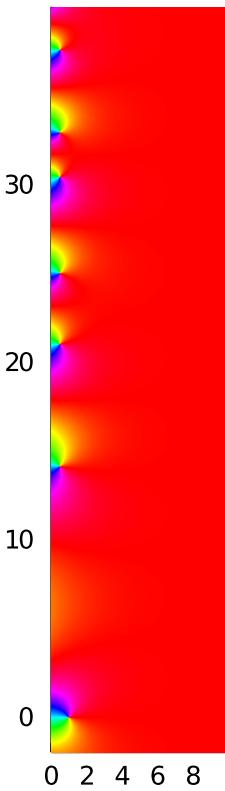
`gammap(z)` is derivative of `gamma(z)` w.r.t. `z`.

```
using DualNumbers  
gammap = z -> epsilon(gamma(dual(z,1)))  
phaseplot(-3..3, -3..3, gammap)
```



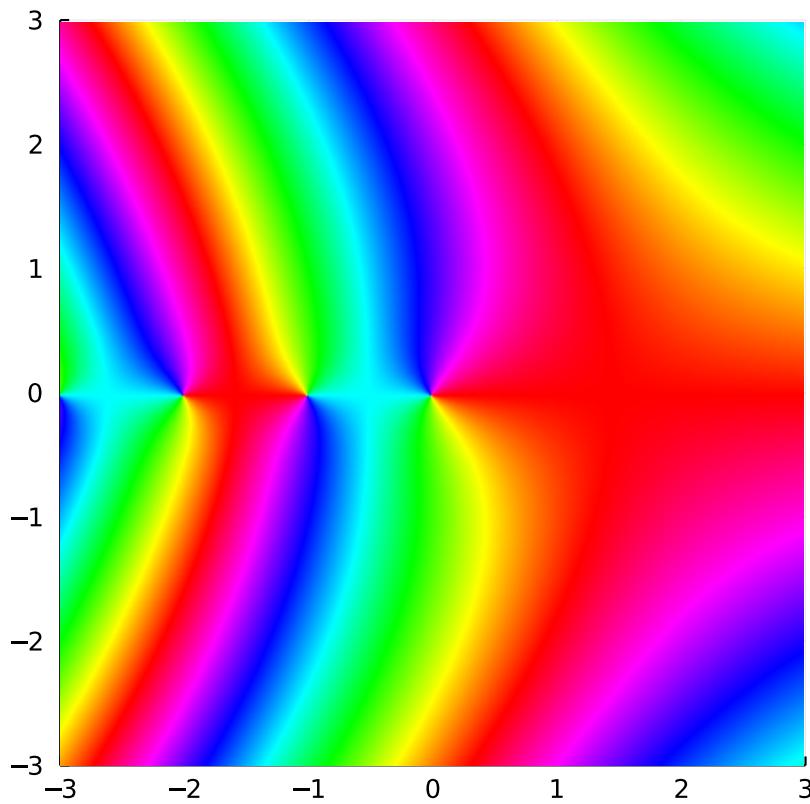
The Riemann Hypothesis: Here is a plot of the Riemann zeta function near the critical line  $z = 0.5 + it$ :

```
phaseplot(0:0.011:10, -2:0.011:40, zeta)
```



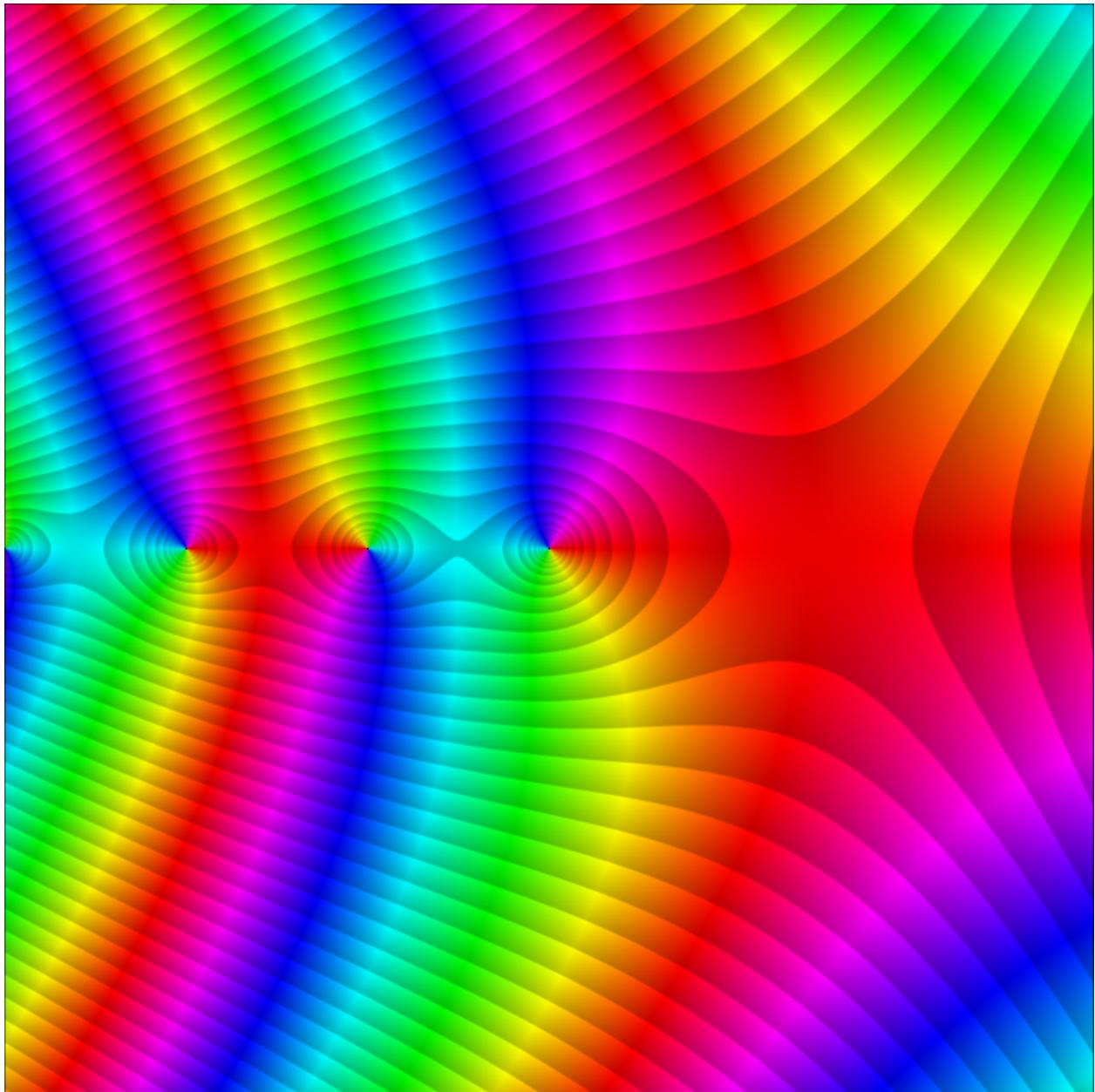
There are ways to include information about the modulus of a function in a phase portrait. Take again the example of the gamma function:

```
phaseplot(-3..3, -3..3, z -> gamma(z))
```



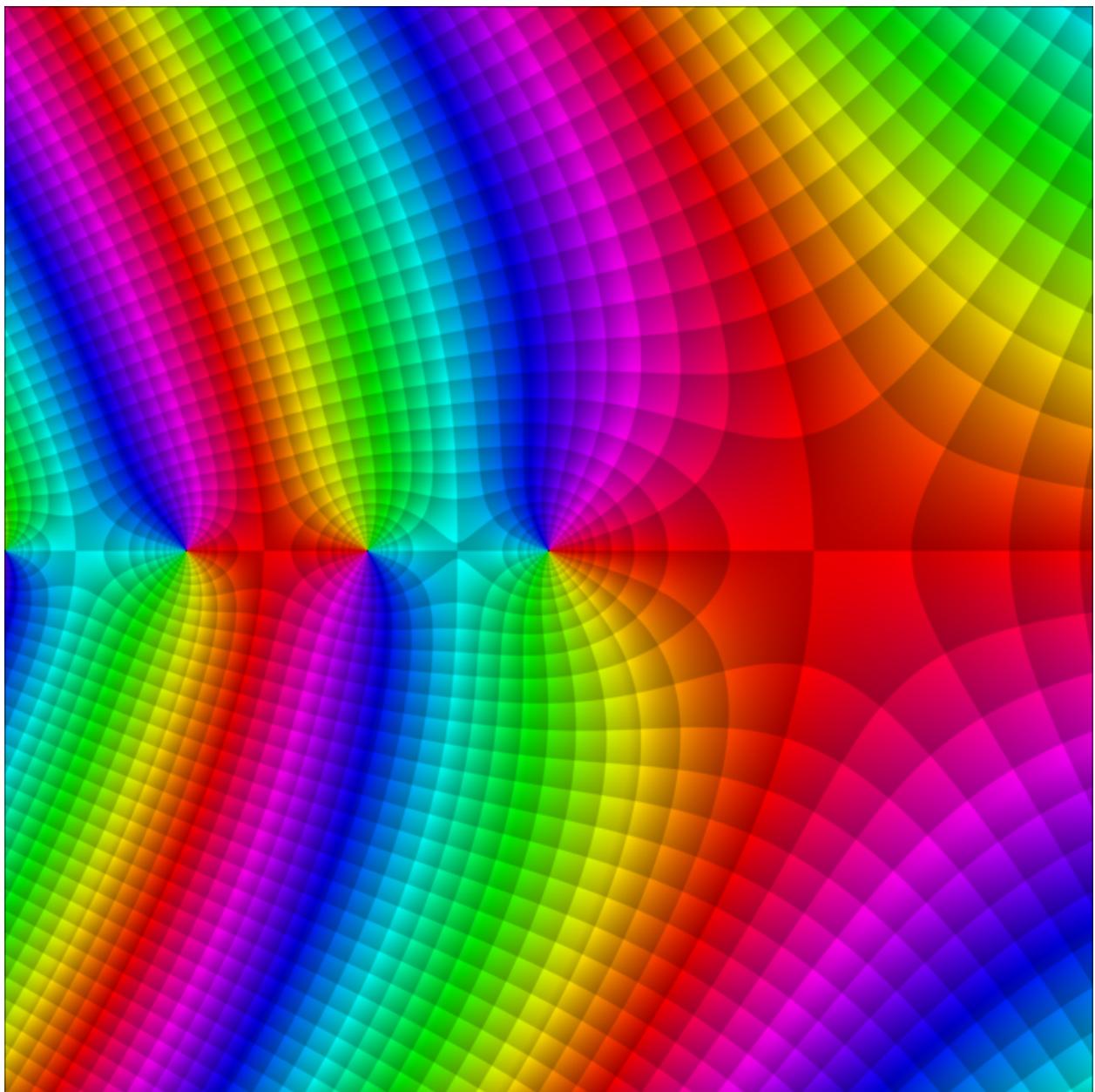
We can add contour lines/level curves of the modulus:

```
xx = -3:0.001:3
yy = xx
zz = xx' .+ im.*yy
fz = gamma.(zz)
portrait(fz, PTstepmod)
```



Next the contour lines of the phase are added:

```
portrait(fz, PTcgrid)
```



We can also shade the phase portrait or plot the modulus in the third dimension. At the moment, Julia does not provide support for this but it can be done in [MATLAB](#).