Visualizing Modular Forms

David Lowry-Duda

Bowdoin College Number Theory
22 November 2019

ICERM and Brown University

Acknowledgements

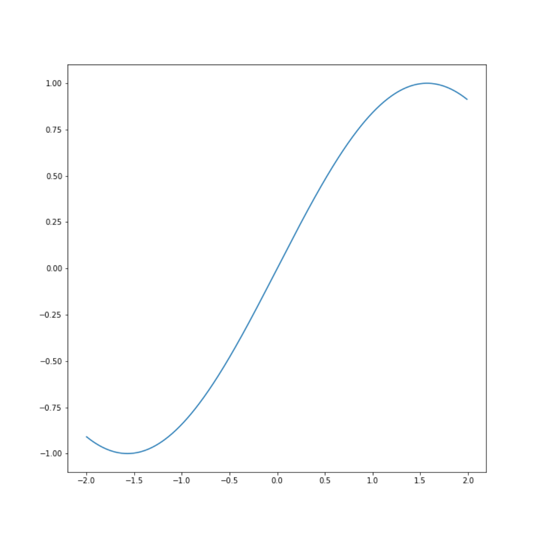
This day would not have happened without Naomi Tanabe. Bringing people together is a special gift.

I draw on inspiration from many sources in this talk. The underlying idea of considering plots of modular forms arose out of conversation with Edgar Costa while I was working on the LMFDB at the University of Warwick. The semester “Illustrating Mathematics” at ICERM has prompted me to revisit the topic — I’m only beginning to realize how I’ve taken for granted the enormous effort that goes into meaningful visualization.

I also thank my wife for encouraging me both mathematically and artistically.

Plotting functions is easy to do, and hard to do well. There are many plotting programs and it is pretty easy to produce a plot that represents *all* of the information about a function.

But many plots can misrepresent the underlying data and mislead the viewer. Different plots and different choices appeal to different aspects of the viewer's intuition — and it's important to recognize that these defaults are probably not always right.



The humble sine plot, according to python's matplotlib.

Plotting Complex Functions

Plotting complex-valued functions is even harder. The graph of a function

$$f : \mathbb{C} \longrightarrow \mathbb{C}$$

is fundamentally 4-dimensional. Representing 3-dimensional objects on a 2-dimensional medium is hard enough. 4 is outright challenging.

It is necessary to either throw away information to reduce the dimension, or to rely on non-spacial ways of representing data, or to use a mixture of both.

Today, we will experiment with a variety of visualizations.

This is the default plot of the identity function in sage. Given a point $z = re^{i\theta}$, we let the color be determined by the argument θ and the brightness be determined by the magnitude r.

The argument is periodic and the color wheel is circular, so this representation has some logic to it.

Brightness for magnitude isn't as effective.

Modular Forms

A modular form is a well-behaved complex-valued function f defined on the upper half-plane $\mathcal{H}$. These forms have symmetries of the form

$$f\left(\frac{az+b}{cz+d}\right) = (cz+d)^k f(z), \qquad \gamma = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma(N) < \mathrm{SL}(2, \mathbb{Z})$$

for an integer k which we call the *weight* of the form.

This implies that these forms will be periodic horizontally,

$$f(z+N) = f(z),$$

and satisfy a more complicated periodic condition in other directions. *Nice* (i.e. holomorphic cuspidal) forms quickly vanish to 0 as $\operatorname{Im} z \to \infty$, and we focus on these forms today.

How should we plot them?

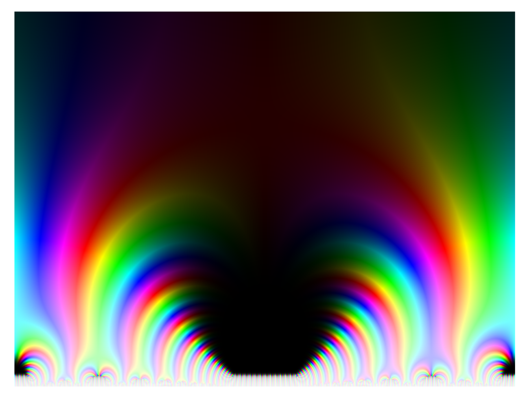
Plotting the whole upper-plane is not an option. But it might be reasonable to plot one horizontal period. Further, since $f(z) \to 0$ as $\operatorname{Im} z \to \infty$, we might hope that truncating our plot at some height Y omits little interesting behavior.

Take $f(z)$ to be the unique holomorphic cusp form of weight 12 on $\mathrm{SL}(2, \mathbb{Z})$. This is the Δ function, studied famously by Ramanujan.

Performing a default plot of f on a segment of $\mathcal{H}$ looks like the following.

This is plotted on $[-.5, .5] \times [0, 0.75]$. This is a plot of a modular form.

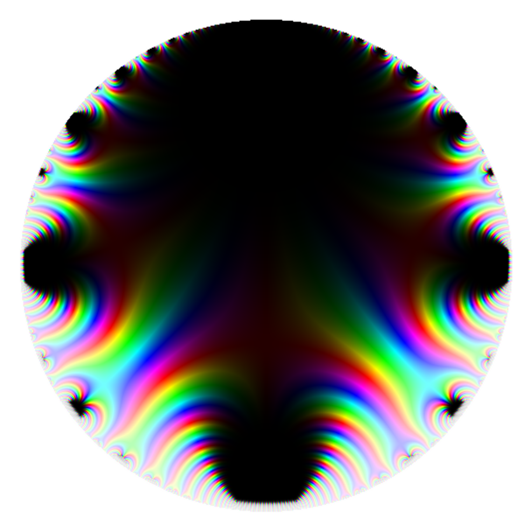
If you look carefully, you can see artifacts from numerical approximation near the horizontal boundary. But I don't get into the details of approximating values of modular forms today.



Alternately, we could choose a different representation of $\mathcal{H}$. The upper half-plane is conformal to the Poincaré disk, so it is possible to map $\mathcal{H}$ to the disk $\mathbb{D}$. I choose an orientation by choosing to map to the disk with

$$\begin{aligned} \mathcal{H} &\longrightarrow \mathbb{D} \\ 0 &\mapsto -i, \\ i &\mapsto 0, \\ \infty &\mapsto i. \end{aligned}$$

The effect here is to present a little *medallion* representing a modular form.



There are still artifacts around the boundary, but this is a very visually appealing representation. It contains “more” information than the part of $\mathcal{H}$, but it is perhaps challenging to reason about the behavior.

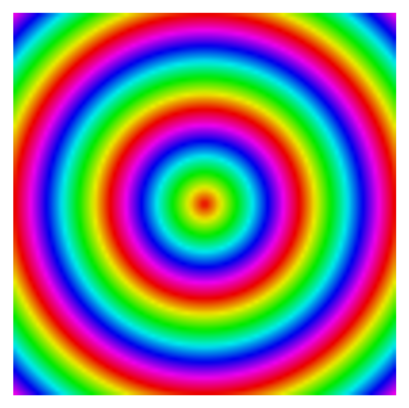
But on the other hand, it’s pretty dark. It turns out that vanishing as $\operatorname{Im} z \to \infty$ leads to lots of dark spots.

Throw away the argument

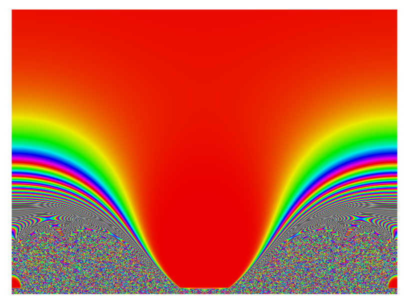
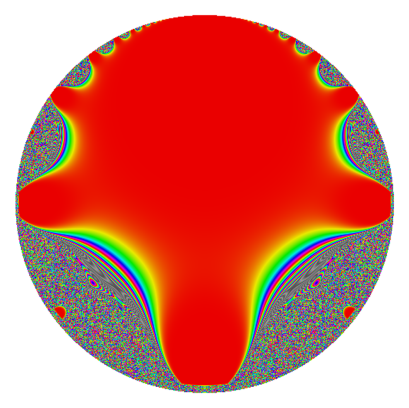
These pictures were interesting, but largely black. What do we care more about — the magnitude or the argument? (The answer is not obvious).

We can simplify the pictures by throwing away the argument entirely and plotting only the magnitude. Relying on brightness/grayscale leads to inexpressive plots. Instead, we might cycle through the color wheel.

On the next slide are two representations of the identity function. On the left is the one from before. On the right, we let the colors cycle. In particular, two consecutive red bands will denote points whose magnitude is 1 apart.



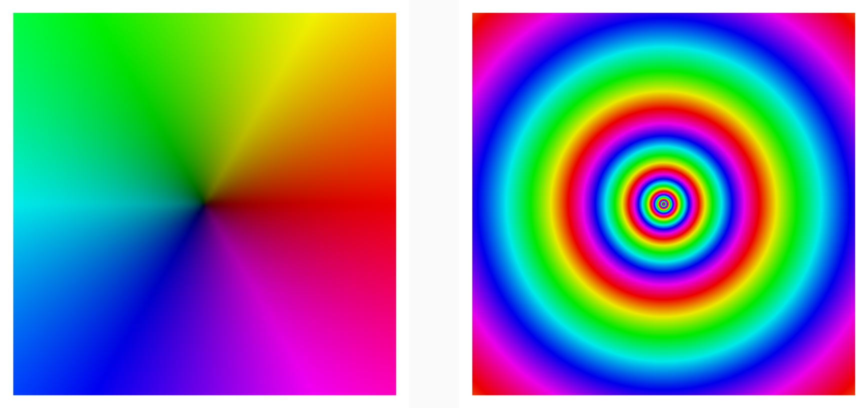
These each emphasize different things. On the left, it's pretty clear that the argument is smoothly varying. On the right, it's pretty clear that the magnitude is smoothly varying.

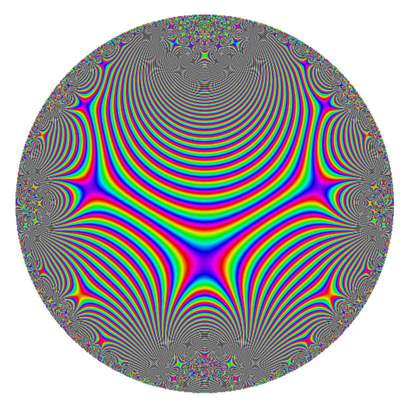
The large red blobs mean *small*. Unfortunately, when the form isn't small, the values change so rapidly that the plots look more like static and less meaningful.

A closely related idea would be to change how colors relate to the magnitude. Instead of consecutive bands of the same color corresponding to values that differ by 1 (or any constant), we could choose to allow consecutive bands of the same color to correspond to values that double or halve. That is, points in one red band would either be double or half the values in the next red band.

We again plot a pair of identity functions.



Given only the plot on the right, it's hard to know whether the function peaks or troughs at the origin. But applied to a modular form, we get the following plots.

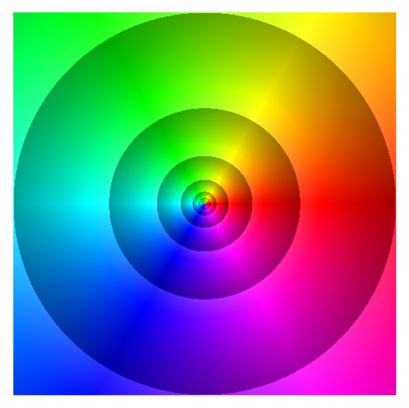
These are beautiful, and one can really see the fractaline nature of the magnitudes in the disk model.

Keep only some magnitude information

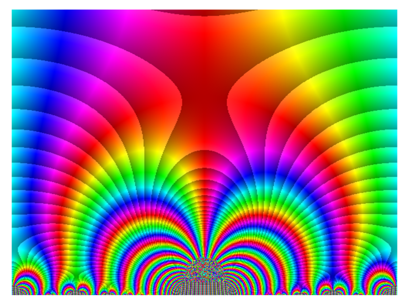
Elias Wegert suggested that I do a different sort of plot. In the following plots, we keep the argument (plotted as color again) and change how we plot the magnitude. To avoid dark or bright spots, we plot contours.

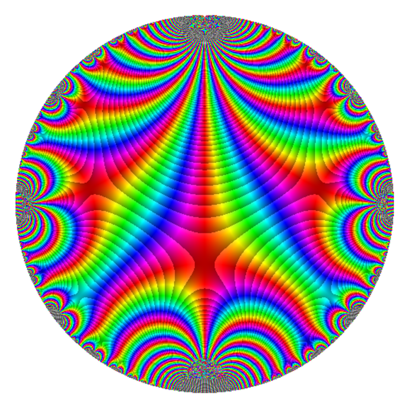
Actually, we don't compute contours — everything that follows is computed by domain coloring. Effectively one chooses the coloration of the identity function and pulls back the coloring to the modular form's values.

Further, the plots that follow are made by plotting functionality that I wrote. It's not perfect (as we'll see), but it's very nice.

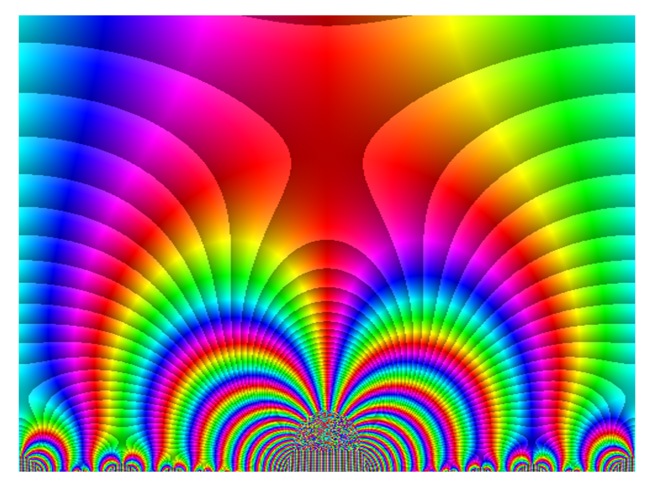


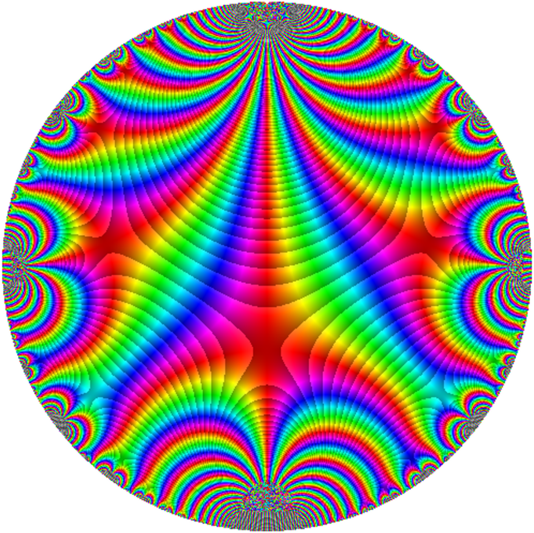
The plot on the right combines many of the good aspects of the default plot and the color-as-magnitude plot.

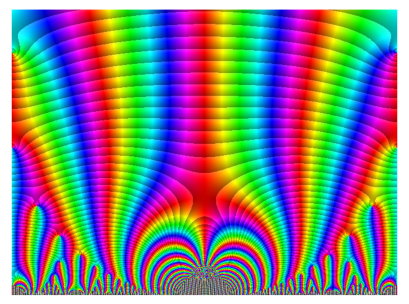




I like these plots, so I included them on separate slides for slightly bigger versions. (I don't dwell on them, but these slides are available on my website).







For comparison, I chose a different modular form (of weight 24) and show these plots here.

Color

I showed these pictures to Ed Harriss (who has made multiple mathematical coloring books). He was very supportive, but he also told me in no uncertain terms that I was using one of the worst possible colormaps, and I should revisit the notion of color.

This prompted me to investigate why the default colormap in sage's complex plot is bad. I learned a lot. I link to two very informative sources of information below.

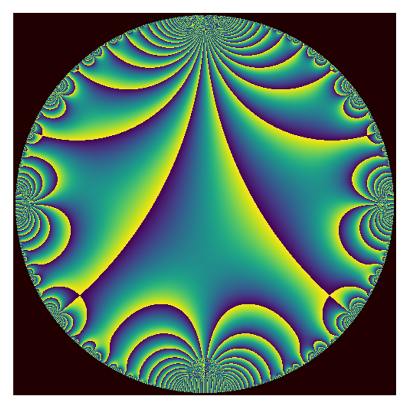
1. http://jakevdp.github.io/blog/2014/10/16/how-bad-is-your-colormap/
2. (SciPy 2015: A Better Default Colormap For Matplotlib) https://www.youtube.com/watch?v=xAoljeRJ3lU

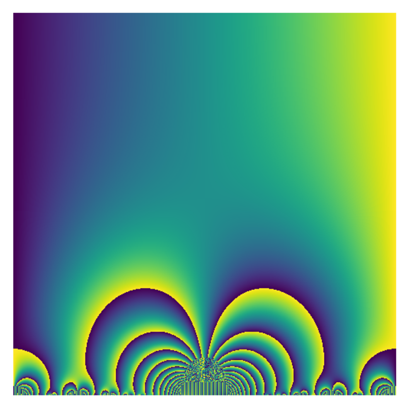
There are multiple problems. One problem is that “actual” brightness is not the same as “perceived” brightness. Colors are weird. The human eye is weird.

This is the standard coloring of the identity function again. Although this very reasonably goes through the colorwheel, we don’t see each color equally. Green is very dominant — it both appears to take up more space and appears brighter. Red is very small, and is eaten up by the brigher yellow-green and the pink-purple.

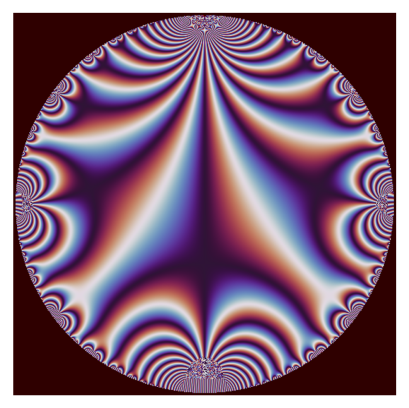
These are not features of the identity function — they are features of the colormap. These artificial features are fundamentally ways in which the plot is misleading us. Further, this colormap is not brightness-correct or colorblind friendly. We can do better.

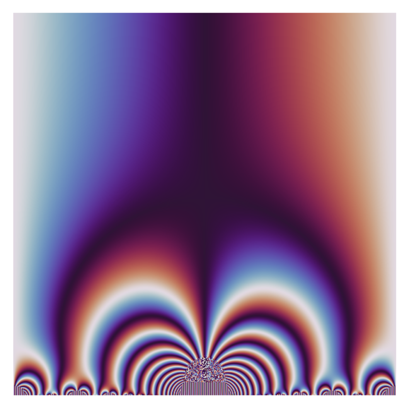
These are two "state-of-the-art" colormaps. They are both perceptually-uniform (color changes linearly with brightness on the left, and cyclically even on the right). The left is called *viridis*, and is now the matplotlib standard. The right is called *twilight*. Note that the left is not cyclic, while the right is.



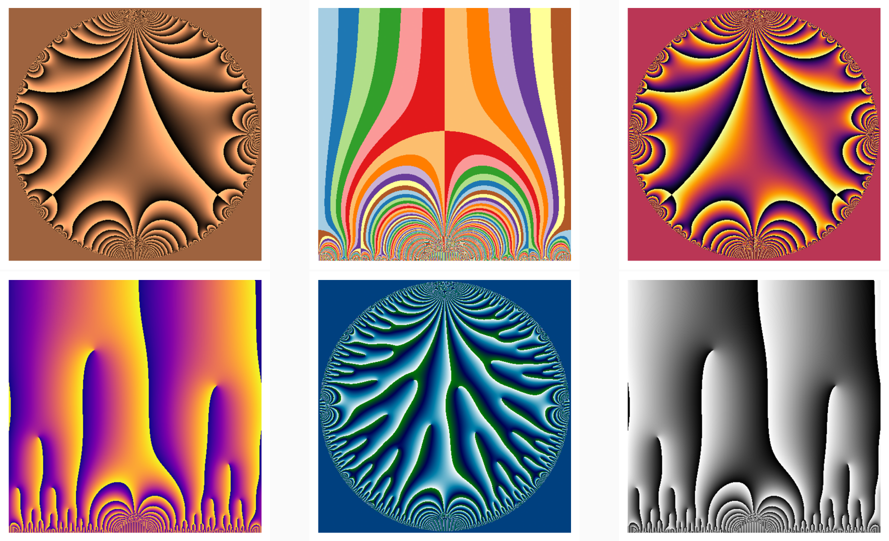


This is visually striking, but again the non-cyclic nature of the colorscheme introduces artificial features.





Arguably, this is the “most honest” representation. I would like to add the contour-like features to this plot, but I have not yet done this. (This is because I wrote the contour-like plot functionality is HSL colorspace, but these colormaps are naturally defined in RGB colorspace, and the transitional code I’m missing is thus confusing).



Other choices of colors yield slightly different views. (The bottom three are images of a the modular form of weight 24).

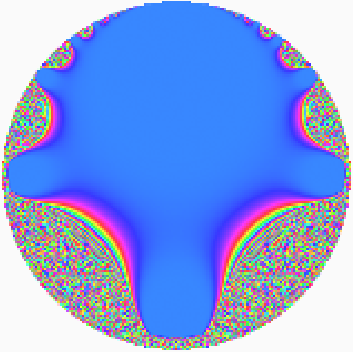
Conclusion

We've now seen many different visualizations of mostly two objects (the identity function and the Δ function, with a bit of a weight 24 cusp form here and there). I hope I've given the impression that there is no single obvious visualization; different visualizations highlight different aspects.

This is the currently used plot of the Δ function on the LMFDB.

There is lots of room for improvement!

What should we replace it with?



Thank you very much.

Please note that these slides are available on my website (davidlowryduda.com). The code used to generate these images is also available (or will soon be available).