
Subject: Re: Logarithmic Color Scaling

Posted by [Braedley](#) on Tue, 05 Dec 2006 20:06:31 GMT

[View Forum Message](#) <> [Reply to Message](#)

JD Smith wrote:

> On Tue, 05 Dec 2006 08:16:14 -0700, David Fanning wrote:

>

>> Folks,

>>

>> I'm embarrassed to admit this, but I spent the entire day yesterday
>> working on a logarithmic color scaling problem and got absolutely nowhere.
>> I was really counting on a breakthrough in the shower this morning, but no
>> joy there, either. :-(

>>

>> My dilemma is this. I can produce a log scaled image (using LOGSCL) and I
>> can create a log scaled color table (again using LOGSCL with the method
>> Lagos outlined yesterday). What I cannot do is associate a color on the
>> color bar with the actual image value.

>

> My personal opinion is to keep the scaling of the image data, and the
> mapping of image data over some min->max range to colors on the
> display as separate. The former can be quite flexible, log, sqrt,
> asinh, whatever. The latter should be linear, and reflect the mapping
> using axes which properly map original data values to colors. Why do
> I make the division this way? Presumably the data are floating point
> or double floats, and can take much more extreme scaling before they
> begin to suffer from roundoff and other numerical concerns. Not so
> with a 256 element byte color table.

>

> In that context, I think you are double-logging. I.e. you are scaling
> your data logarithmically, and then separately scaling your color map
> *and* the colorbar axis as well. This could explain why your values don't
> match up.

>

> You could either a) just display the linear color-bar (i.e. what you
> actually used, with logarithmic axes of course), or b) load a
> logarithmically mapped color bar as you do first thing, and run the
> *linear* image data through it and display with a linear axis, or c)
> use a linear color table with a log scaled image, display this
> log-scaled color bar, but then use a linear X axis values. You can't
> both map the colors *and* map the axis values, that's "double-logging".

>

> There are four places log could get applied, two each for data and
> colorbar. You must pick one on each side of the equation.

>

> 1) to the data themselves
> 2a) to the colormap indices (for displaying data)
> 2b) to the colormap indices (for displaying colorbar)

> 3) to the axis of the colorbar
>
> Here are the possibilities (3 of which I discussed above):
>
> a: 1 (data side) + 3 (colorbar side)
> b: 2a (data side) + 2b (colorbar side)
> c: 1 (data side) + 2b (colorbar side)
> d: 2a (data side) + 3 (colorbar side)
>
> you are now using:
>
> e: 1 (data side) + 2b (colorbar side) + 3 (colorbar side)
>
> My preference, which keeps things simple, is a). This would be
> especially true if you implemented one of the scaling functions used
> in the Spitzer community: LogLog. That could get confusing fast ;).
>
> One wrinkle is if you don't use a colorbar axis. Then a) doesn't work so
> well. In that case, you can use c), with an "implied" linear x-axis.
>
> JD

I would tend to agree with JD on this. I don't think it would be a good idea to scale the color table (and hence the colorbar). I've never had to worry about the problem you're facing because whenever I did anything remotely like this, I worked in dB, and so my colorbar was labeled in dB. I think it would be trivial for you to go from this thinking to labeling your colorbar logarithmically.

Braedley
