
Subject: Re: REGRESS and sky background
Posted by [Jeremy Bailin](#) on Tue, 06 Jul 2010 13:19:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Jul 5, 8:06 pm, Gray <grayliketheco...@gmail.com> wrote:

> Hi all,
>
> I'm baffled with how one of my programs is acting, and would love some
> insight, if there is any to be had.
>
> The routine is designed to fit the sky background image (in my case, a
> 128x128 subdivision of a larger astronomical image) to a plane ($Ax+By$
> $+C$) using REGRESS. My subdivisions are small enough that I think a
> plane is a pretty good approximation; the idea is to do a 3.5-sigma
> mean clip to remove sources, then regress the sky pixels to a plane
> and subtract the plane, and iterate until the fitted plane reaches 0.
> The problem is that it seems the slope of the background increases
> with increasing iterations, which it theoretically should not do.
>
> Here's my general algorithm; I actually use a different mean clipping
> routine, but astrolib's MEANCLIP gives the same (unwanted) results.
> Take a look and tell me what you think. Thanks!
>
> --Gray
>
> FUNCTION find_skybg, image, sigma
> img = image
> s = size(img,/dim)
> lx = rebin(indgen(s[0]),s[0],s[1]) ;x and y coordinates
> ly = rebin(indgen(1,s[1]),s[0],s[1]) ;to construct bg plane
> abc = fltarr(3)
> iter = 0
> repeat begin
> meanclip, img, m, subs=clips, clipsig=3.5 ;don't care about mean,
> just clips
> xy = array_indices(s,clips,/dim)
> ab = reform(regress(xy,img[clips],const=c))
> sigma = stddev(img[clips])
> abc += [ab,c]
> bg = ab[0]*lx+ab[1]*ly+c
> img -= bg
> iter++
> endrep until (iter ge 10 or total([ab,c]/abc le 0.02) eq 3)
> background = abc[0]*lx+abc[1]*ly+abc[2]
> return, background
> endfor

Not sure... I just tested it out on an image with stars and a

background gradient and it worked exactly as expected. What fraction of the image is making it through the sigma clipping? I could see it being unstable if that fraction is sufficiently small. Is that fraction reasonably stable from iteration to iteration? Maybe there are an unusual number of pixels right around 3.5σ , whose inclusion or exclusion makes a big change to the solution?

-Jeremy.
