
Subject: regression with poisson models

Posted by [matzke](#) on Mon, 18 Jun 2001 00:59:31 GMT

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

Hi,

...hope that you smart astronomy types can take a minute to answer a question from a humble geographer.

I have been using the function REGRESS (IDL 5.4) to test correlations between images derived from two different sensors (simply using each pixel-pixel comparison as an observation). This works fine, but both datasets appear to have poisson distributions rather than gaussian distributions.

In S-plus 2000, it is a fairly simple matter to use the 'Statistics --> Regression --> Generalized Linear Model' menu option, and change the model for the data from Gaussian to Poisson. But there appears to be no similar easy option for IDL REGRESS or similar functions. The only relevant line that I could find in the IDL help files was this (in the help entry for REGRESS):

```
=====
MEASURE_ERRORS
```

Set this keyword to a vector containing standard measurement errors for each point Y[i]. This vector must be the same length as X and Y.

Note - For Gaussian errors (e.g., instrumental uncertainties), MEASURE_ERRORS should be set to the standard deviations of each point in Y. For Poisson or statistical weighting, MEASURE_ERRORS should be set to SQRT(Y).

```
=====
```

...but if I just set the keyword MEASURE_ERRORS to SQRT(Y), I get errors because of zeros in the array Y -- even once I got the dimensions etc right.

Here's my workaround:

```
errorarray1[0:points2-1]=sqrt(abs(y[0:points2-1,0]))
errorarray1=errorarray1+0.0000001
...and then I use MEASURE_ERRORS=errorarray1. This code doesn't crash
(it does if the 0.0000001 isn't added), but this seems like a pretty
half-baked solution; plus I don't really have any idea if this is the
equivalent of assuming a poisson distribution rather than a normal
one.
```

Is there a simpler way to do a regression in IDL assuming poisson-distributed data rather than normal distributions? Is what I've got right now even doing this?

To Craig, who is I think the resident IDL genius:

I found the newsgroup from your webpage, & downloaded the programs for image regression...I'll see if I can get them to work. Is there perhaps a solution to my problem within one of the functions you've created?

Back to the group:

Anyway, I hope this isn't too silly a question. Both my stats & IDL skills are limited. I suppose I could learn the s-plus scripting language, but I've spent the last few months learning IDL, & love it for image analysis (along with ENVI), & would hate to have to learn a whole 'nuther language just to solve this problem. I have a zillion variations on these images to compare, so doing things manually in S-plus doesn't seem like a likely option.

Thanks, Nick

PS:

Why am I doing this? See my webpage:

<http://www.geog.ucsb.edu/~matzke/mad/home.html>

...And if you want to st-borrow some (perhaps primitive, but functional) IDL 5.4 scripts, check out the discussion page at my dept:

<http://pinon.geog.ucsb.edu/cgi-bin/forumdisplay.cgi?action=topics&forum=IDL,+ENVI,+other+RS+software&number=5&am p;DaysPrune=45&LastLogin=>

(Do 'show all topics' to see some of the older stuff)

Thanks, Nick

Subject: Re: regression with poisson models
Posted by [Craig Markwardt](#) on Tue, 19 Jun 2001 05:39:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Nick--

matzke@geog.ucsb.edu (Nick Matzke) writes:

>
> I have been using the function REGRESS (IDL 5.4) to test correlations
> between images derived from two different sensors (simply using each
> pixel-pixel comparison as an observation). This works fine, but both
> datasets appear to have poisson distributions rather than gaussian
> distributions.
... long description deleted ...

First, I wanted to note that the image fitting code you will find on my web page is not particularly suited to your task. As you note for yourself, what you are really doing is a linear regression between two datasets, and the fact that the data came from images is largely incidental.

Nick, I think you are generally on the right track. I believe you will want to substitute the Poisson error in place of the usual Gaussian error. This is done all the time for counting statistics, which I assume is likely to be your case too.

Now, closer to your specifics. You are describing REGRESS in IDL 5.4, which is still "new" in my book [i.e. I don't have regular access to it yet :-]. It appears that major changes have occurred with all the regression codes in IDL 5.4, and to be honest I do not think all the changes are good. The big change is a shift to the MEASURE_ERROR-style keywords instead of weights. The problem with this is exactly what you found: you can't assign a *zero* weight without crashing the program!

My mind actually boggles at this change. I hope they correct course somewhat and allow both error and weight options. The MPFIT family of routines does.

Workarounds, in order of difficulty:

- * use the WEIGHTS parameter instead of MEASURE_ERROR. It is obsolete, but it still works.

- * remove the zero-valued entries from your data set. They get a zero weight anyway, so removing them cannot hurt.
wh = where(y GT 0)
result = regress(x(wh), y(wh), measure_error=sqrt(y(wh)), ...)

- * use MPFITFUN with a linear function, for which WEIGHTS will always work, I promise. However you don't get all those clever automatic statistics that you get with REGRESS. :-)

Good luck,
Craig

--

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
