Poisson errors, fitting positions (due 1 Mar 2013)

Marten van Kerkwijk

14 February 2013

Numerical simulations

In this problem set, you will use numerical simulations to test some ideas presented in class. You can use any numerical package you happen to know, but should consider writing your own code, especially if you have never done this. To help you get started, get inspired by numpy.random.poisson or look at testpoisson.py, a short python programme using numpy and matplotlib I wrote for the first question. For the other questions, you may want to look at scipy.optimize.leastsq. In general, if you need help with writing code, do ask! An important goal of this problem set is to get you to a point where you don't hesitate to do a Monte-Carlo simulation if it seems useful.

Poisson distribution

In class, I mentioned that one has to be careful assigning weights $\sigma = \sqrt{N}$ for Poisson-distributed data. Verify that statement, by doing a simulation in which you do $n = 100$ trials of $m = 100$ points each for Poisson-distributed data with mean μ=100, and calculate both the straight average and a weighted average. Make a histogram of these averages, and discuss in a few words why the two give inconsistent results. Also try one or two different mean values μ and look for consistencies in the deviations. (*For more discussion, see Mighell 1999, ApJ, 518, 380.*)

Fitting gaussians

Now write a routine that simulates a one-dimensional image of a star, with model data $M(x)$ given by,

$$M(x) = C + A\frac{1}{\sqrt{2\pi}\sigma_w}\exp\left[-\frac{1}{2}\left(\frac{x - x_c}{\sigma_w}\right)^2\right].$$

For definiteness, use pixels $x = 0\ldots99$, sky $C = 100$ photons, amplitude $A = 5000$ photons, and width $\sigma_w = 2$ pixels. Now add Poisson noise to all these data points, and do a non-linear least-squares fit with a Gaussian (assuming uncertainties of $\sqrt{N}$, i.e., ignoring the minor deficiency arising from that assumption that you found above). Repeat 100 times and compare the typical uncertainty on x_c inferred from the covariance matrix with the scatter between different trials. How does the scatter relate to the width σ_w and the signal-to-noise ratio?

Gaussian fit to non-Gaussian PSF

On purpose, make stars that do not conform to a simple Gaussian, e.g., by replacing the above model with (feel free to do something else),

$$\begin{aligned} M(x) &= C + A(1 - \alpha)\frac{1}{\sqrt{2\pi}\sigma_w}\exp\left[-\frac{1}{2}\left(\frac{x - x_c}{\sigma_w}\right)^2\right] \\ &+ A\alpha\frac{1}{\sqrt{2\pi}\sigma_{w,2}}\exp\left[-\frac{1}{2}\left(\frac{x - x_c - \delta_x}{\sigma_{w,2}}\right)^2\right]. \end{aligned} \tag{1}$$

This mimics a poor PSF, in which a fraction $\alpha = 0.2$ of the light is in, say, a wider PSF with width $\sigma_{w,2} = 4$ pixels, which is offset from the good "core" by $\delta_x = 1$ pixel. Again, add Poisson noise and fit the model star, but fit it with the simple model from the previous question (i.e., ignore the second Gaussian in the fit). Compared to the good fits, what happens to χ^2? What happens to the uncertainty on x_c inferred from the covariance matter? How does the latter compare to the offset from the true position? And how does it compare to the scatter around the mean from the different trials? If you wanted to measure differences in positions on an image, what would you suggest is a good estimate for the uncertainties on these position differences?