Physics 266. Fall 2016. Homework 6

Due on 23 Nov, 2016

SHOW ALL WORK AND HAND IN ANY CODE YOU WROTE TO DO THE CALCULATIONS.

1. Simple MCMC with emcee

Here we are going to do some wavelength calibratioin, but with MCMC to estimate the parameter PDFs. On canvas under Files → Data is wavecal.txt, which contains (x, λ) pairs (x is the first column and λ is the second column). Assume that the error on each of the x points is 0.4 pixel. We will be finding $x = x(\lambda)$ using a polynomial fit. To help you with this problem, you will find on canvas (Files → python) an ipython notebook called emcee_example.ipynb. Look it over to help you with this problem.

(a) Use `emcee` to fit a polynomial of the following form to the data.

$$x = a_0 + a_1\lambda + a_2\lambda^2 + a_3\lambda^3$$

Plot the results using the corner.py code, as at the end of the example.

Note: In your earlier homework, you fit a 2nd order polynomial to this data set. Use your best-fit values of a_0, a_1, and a_2 to set the initial guesses for those parameters in this fit. Center your initial guess for a_3 at 0.

(b) Give the 68% confidence regions for each of the four parameters. The corner plot may report these, but for some of the parameters, it won't show enough significant digits. You can follow the example text to see how to get these values.

2. A first data reduction pipeline

In this problem, you will combine some steps that you have developed in earlier problem sets to make a simple data reduction pipeline that you will then apply to real data. The data for this problem were obtained by a 10-inch telescope that was set up on the roof of the Physics Building. For this telescope, the CCD was not cooled with liquid nitrogen, so there is a noise term called dark current depends on the exposure time. To correct for it, we take exposures of the same length as the science exposures, but with the shutter closed, so that the only contribution is the dark current (plus noise). This is then subtracted from the science data.

(a) On canvas, under Files→Data you will find three files called `Dark_T10_120s_*.fits`. Take the median of the data in these three files to create a dark-frame image. Now subtract that new image from the data in the file called `NGC4874_T10_2010_V_1.fits` to create a dark-subtracted science image. Display the dark-subtracted image using the clipped-σ ranges from HW3, i.e., from $-1\sigma_{\rm clip}$ to $+10\sigma_{\rm clip}$. Also don't use the standard imshow color map (which is called "jet"). Instead, use the "YlOrBr_r" color map from the matplotlib.pyplot.cm library.

(b) On a previous homework (HW3) you created a flat-field file. If you did things correctly, the mean / median of that file should be around 1. Take the dark-subtracted image that you obtained in part (a) and do the flat-field correction by dividing that image by the flat-field image that you previously made. Display the result using the same display parameters as in part (a).

3. Graphical Fourier Transforms and the Convolution Theorem

Use your knowledge of the convolution theorem and the graphical Fourier transforms that we covered in class to draw graphs of the Fourier transforms of the following two functions. **NOTE:** I do NOT want you to solve these using integrals. Just draw the plots as we did for several examples in class.

(a) $F(\omega) = \cos(\omega T_0)\,\mathrm{sinc}(\omega T_0)$

(b) $f(t) = e^{-t^2}\cos(\omega_0 t)$

Helpful note: Although we did not discuss this in class, the Fourier transform of a Gaussian is just another Gaussian, with the usual FT convention of broader things transforming into narrower ones, and vice versa. That information alone is enough for you to do this question, but just to be complete, here is the transform:

$$f(t) = e^{-at^2} \Longleftrightarrow F(\omega) = \sqrt{\frac{\pi}{a}} e^{-(\omega^2/4a)}$$