UCF Physics: AST 5765/4762: (Advanced) Astronomical Data Analysis

Fall 2019 Homework 9

Due Tuesday 29 October 2019

Work:
Become sufficiently familiar with image corrections to:

1. Know what calibration data to acquire given the observing objectives and the sky conditions,
2. Create calibration frames appropriate to a given dataset,
3. Remove dark current, bias, background, and flat-field variations from an object frame.

Resources:

1. (no new reading)

Hand in:

This assignment continues the correction of the data in `WebCourses/Files/data/hw7`. Be careful to follow the instructions above the questions in HW7.

1. (10 points) Copy the routines and homework files from the previous homework (including any inclusions of prior assignments) into the directory for this assignment. Correct any errors you may have had, making a comment that says "`# FIXED:` " and giving the date. You may refer to the posted solution, but if you do so you must state what you used from it. The only non-comment for this problem should be the prior homework file run as a batch job (e.g., `from hw8_sol import *`). This will run all the homework files as batch files, each calling the prior one as its first thing, back to HW7. Of course, make sure they all still run without errors. It is a good idea to compare your results to those in the solutions.

2. (10 points) Write a series of Python commands that creates a flat field frame from the `k_lampon` and `k_lampoff` data in `WebCourses/Files/data/hw7`. Combine the lamp-on and lamp-off sets separately with median combination. Subtract the lamp-off median frame from the lamp-on median frame. Normalize the result, using the normalization region of HW8. Write the flat field to a file named `flat.fits`. Print the value of pixel index `[217, 184]`.

3. (10 points) Apply the flat field to all the sky-subtracted object data (can you figure out how to do it without a loop, using broadcasting?). Print the value of pixel index `[217, 184]` in each frame, in one command if you can. Write the last resulting frame (with modified header) to a file named `stars_13s_9_flat.fits`.

4. (10 points) Write a routine called `disk` that produces a 2D mask array with the image of a disk in it. Pixels in the image should have a value of `True` for each pixel whose center is within a certain radius of a given (possibly fractional) position, and `False` elsewhere.

The inputs should be, in order: the radius, a 2-element array or tuple giving the center of the disk (y, x order), and a 2-element array or tuple giving the image shape to make (y, x order). ORDER IS VERY IMPORTANT! Store the result of `disk(6.2, (12.3, 14.5), (25, 30))` in a file named `disktest.fits` (save as 8-bit unsigned integer data). Print the values of pixel indices `[14, 14]` and `[2, 1]`.

5. Calculate each of the following without reference to notes. Then look at your notes or Bevington but do not do the problems. Then later, try again. Repeat until you can do them and similar problems without notes.

   (a) (3 points) $\frac{6\pm3}{3\pm1}$

   (b) (3 points) $\frac{5\pm1}{2\pm0.01}$

   (c) (2 points) $(6 \pm 3) \times (3 \pm 1)$

   (d) (2 points) $(5 \pm 1) \times (2 \pm 0.01)$

   (e) (1 points) $(6 \pm 3) + (3 \pm 1)$

   (f) (1 points) $(5 \pm 1) + (2 \pm 0.01)$

   (g) (1 points) $(6 \pm 3) - (3 \pm 1)$

   (h) (1 points) $(5 \pm 1) - (2 \pm 0.01)$

   (i) (1+2+3 points) Find the mean, standard deviation, and standard deviation of the mean for the set (part of a larger population):
   4, 5, 4, 3, 2, 6, 7, 4, 5, 3, 2, 4, 5, 4, 3, 5.

   (j) (3+3+4 points) Find the error-weighted mean, standard deviation, and standard deviation of the mean for the set:
   (5 ±1), (4 ±2), (5 ±2), (6 ±1), (5 ±2), (5 ±1), (3 ±1), (4 ±2), (7 ±2), (5 ±1).

6. (10 points) Include a copy of your class log file in your handin. Print the Git log for your main homework file.