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

Fall 2019 Homework 7

Due Tuesday 15 October 2019

Work:
Become sufficiently familiar with astronomical imaging arrays to:

1. Know the different types of imaging arrays and how they work.
2. Understand the systematic and random effects relevant to array data, including photon noise, dark current, bias, read noise, sky background, and flat field variations.
3. Know when and how to apply the various corrections to the systematic errors in array data.
4. Be able to calculate signal-to-noise ratios for given sets of observing parameters.

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. Chapters 4.6 through 5 of Howell (**DUE** before class Tuesday 15 October 2019)

Introductory notes:

1. In this and the next several assignments, you will correct and make measurements from the data in `Files/data/hw7` on WebCourses. **They are not quite normal FITS images!** See below for where to put these on your machine, how to refer to them, and how to read them.
2. Expect assignments to take 5–7 hours/week each if you understand basically what to do, and much more if you don't. Please come to office hours and **ask questions** rather than wasting long periods being stumped. Start them early, and work ahead to the next assignment if you want to.
3. The solutions to these assignments are very detailed, and often contain commentary on methods and the thought processes that go into making analysis decisions. Be sure to read them **and run them**, even if you got 100% on your assignment.

4. **Important!** Download files from the Files area of WebCourses into the directory `/home/`<username>`/ast5765/WebCourses/Files/`, mirroring WebCourses' structure. We will refer to this directory as "`Files`" from now on. You can hover your cursor on each folder in the Files area, click the wheel icon that appears, and click Download. This will download a Zip file to your computer, which you can put in your new `Files` directory and `unzip`. In the future, as new items appear in Files on WebCourses, download them and put them in the right place under your `Files` directory, duplicating the WebCourses folder structure in your directories exactly. If you copy announcements, text from assignments, etc. out of WebCourses, you can put them in like-named directories under `WebCourses`.

5. You should already be putting your homework in `/home/`<username>`/ast5765/handin/hw7_`<username>`/`. In your homework, use the relative path `../../WebCourses/Files/data/hw7/` as your data directory. If you use another path, your homework will not run when we correct it, which costs points.

6. Hard-coded values should be in your main homework file, as requested. Do not hard-code any values into any routines you write; query the headers or pass the values as arguments instead. Your routines should work on any dataset, not just this one.

7. The data are from the WIRC camera at the Palomar 5-meter telescope. **Read the data files with** `Files/python/rdpharo.py`, which corrects their quirks and makes them good FITS data. You can manipulate the returned header with the Astropy FITS routines and methods (see the Python help pages); examples also appear in `Files/python/doc/pydatatut/pydatatut.pdf`).

8. During the analysis, you will read FITS headers and data into Python arrays. When correcting images, work on these arrays in place rather than copying them. **ONLY** when requested, write FITS files in your homework directory. Do this in your main homework file, not in any routine. The purpose of writing out FITS files is so that there is something to grade even if your code does not run in a fresh Python session. Normally, you would not write them out. Do **NOT** read any FITS file that you have written, nor write one that was not requested.

9. Before handing in any of the assignments, quit Python, delete all your FITS files, and run your homework file as a Unix command like this:
   `./hw7_sol.py > hw7_sol.out 2>&1`
   Check the output carefully for buried error messages.

10. Finally, if you are stuck, or if you see a problem and don't know how to fix it, and it is too late to get help, please describe the problem in comments in your homework file. Identifying a problem, even if you can't fix it, nets you partial credit.

Hand in:

1. (10 points) A hypothetical CCD has no electronic noise source except for read noise $\sigma_r = 3$, which occurs in each pixel. The CCD is in space, so the sky is quite black and can be neglected for the purposes of this problem. For a constant stellar flux of $F = 900$ photoelectrons per second arriving at your detector, contained within 13 pixels, **find an expression** for

the signal-to-noise ratio (S/N) for summing all of this signal in multiple exposures. What is S/N if you take ten 5-second exposures? How about for taking just one 50-second integration? Which is better and why? What are some other practical considerations (pro or con) besides just the hypothetical S/N? Note that the CCD Equation in Howell is for calculating one frame's S/N. How would you modify it for multiple exposures?

2. (10 points) Write a series of commands that:

   (a) Assign the data directory name (including the final "/") to the variable `datadir` and the FITS extension string (including the dot) to `fext`. Define NumPy string arrays `objfile` and `darkfile` that give the file names. Do not include the `.fits` extension or the directory name in these. Each array assignment should be a single Python command, one line per file, with the names lined up. If you need more hard-coded values, enter them in this section of the assignment. Print `datadir, fext,` and the last elements of `objfile` and `darkfile`.

   (b) Read one data file and determine the data array sizes. Assign these to `ny` and `nx`. Also define `nobj` and `ndark` to contain the numbers of files by querying the sizes of the string arrays. Print `ny, nx, nobj,` and `ndark`.

3. (10 points) Write a series of commands that:

   (a) Allocate two 3D float arrays, one for the objects and one for the darks, with sizes `nobj, ny, nx` and `ndark, ny, nx` respectively. Print the shapes of the object and dark arrays.

   (b) Read the data into the arrays. Do this in two loops, one each over the object and dark filename arrays. Save the last header from each set in `objhead` and `darkhead`. Read and print the date of observation (`DATE-OBS`) from each of these.

   (c) **Extra credit** (5 points) Why not print `TIME-OBS`?

4. (10 points) Find a function that allows you to apply the median combination method in a single function call, without loops. It should accept a 3D data array and return the median-combined 2D array. Run the routine on your dark data. Print the value of pixel index `[217, 184]`.

5. (10 points) Add HISTORY to the representative dark header, telling that it is a median combination dark frame. See rdpharo.py for an example of adding HISTORY. Write the result of the previous problem (with modified header) to a file named `dark_13s_med.fits`. Write as 32-bit floating-point data. Do not write more than one file.

6. (10 points) Subtract the median-combined dark frame from each object frame, working on the object data array in place. If you're very good, you can do it without loops! Write the first frame to a file named `hw7_(username)_prob6_image1.fits`. Print the value of pixel index `[217, 184]` in the first object frame before and after subtraction.

7. (10 points) Include a copy of your class log file in your handin. Print the Git log for your main homework file.