ASTR 8060
Homework 4

Learning goals: Learn to quantitatively assess digital images using Python, including making line and column plots, computing RMS from selected regions; read and write FITS files in Python; compute gain and readnoise from data; remove bias levels by fitting overscan regions and trim images.

You will find a link to the `Imaging/` data directory in the image_combination notebook. Copy this over to your local machine. logfile.txt contains a description of what each exposure is because the header information is incomplete. These observations were taken with WIROPrime, the prime focus camera on the 2.3m telescope owned by the University of Wyoming, and I will give you its information as needed. For the following analysis, I ask you to perform some computations on your own images using Python and using Python's `CCDPROC` package so that you become familiar with both sets of tools. The reason for doing some of this by hand is so that you realize `CCDPROC` is not a magical black box! I will always accept code that you write yourself in lieu of adopting `CCDPROC` tools. Where applicable, show me your plots that back up your conclusions.

The process of data reduction is basically one of recovering the real (I_{real}) source brightness in an image in the presence of other sources of signal and noise. Each pixel has a relative sensitivity to incoming light, F (for flat field). There is sky background signal, S. There is dark current in each pixel, D. There is the DC bias level in each pixel, B. All of these effects act to give you the observed counts in each pixel, I_{obs}, when what you want to know is I_{real}. Mathematically,

$$I_{obs} = (I_{real} + S)F + D + B$$

The process of image reduction is one of inverting this equation to recover I_{real} from I_{obs}. Uncertainty on the real source, I_{real}, comes from propagating uncertainties in the other terms.

For the following tasks, document each step as best you can with text and by graphical representation in a Jupyter notebook. Turn in plots and quantitative notes, along with your reasoning for each step. Think of these as guidelines for assessing astronomical data, rather than an exact to-do list. Be creative and careful in your in-depth analysis of the data.

Note: In this and the homeworks to follow, it is impossible (and un-wise) for me to simply give you a step-by-step recipe to follow. Rather, this is a semi-free-format exercise where I expect you to encounter questions and difficulties as you get to each step. When you are unsure, come ask for advice. I don't expect anyone to simply go off and do this without interaction with me or other students on a regular basis.

1. Start a Jupyter notebook that will contain all sort of things you will learn in the course of your data analysis. Keep here your procedures that you run so that you can quickly re-run them with a few keystrokes. Also keep here anything you'd keep in a paper logbook regarding notes about your reduction. Markdown in your Jupyter notebook will take the place of formal reduction notes. The quality of the this file will make up one component of the grade for this homework.

2. Examine the images in DS9 to become familiar with where the overscan region of the chip is (on both sides of the data section; make note of where this is). Look at examples

of bias images, dark images, and flats so that you can know by inspection what kind of image you are viewing.

3. Assess the RMS and mean levels of an image. Use Python to practice plotting lines and columns of data from one of the bias images. Use Python to compute the mean and RMS of a region near the center of the chip.

4. Open Phillip Massey's users's guide to ccd reductions with IRAF, linked on the class website. Use this as a rough guide, but we won't follow everything he recommends (there are just too many ways to do the same thing!). An example of these steps in action can be found at the CCD reduction and photometry guide Github tutorial also linked from the class website.

5. Examine the all bias exposures (or overscan regions throughout the night) and quantify how much the bias level changes throughout the night. Compare the mean level of the biases to the mean levels of the overscan regions from other files throughout the night. How much variation do you find? Make an argument for whether it would be a good or bad idea to combine all of the biases to make a master bias and subtract that from all the images versus using the overscan region of each image as an estimate of the bias level.

6. Using `CCDPROC.SUBTRACT_OVERSCAN` and `CCDPROC.TRIM_IMAGE`, fit and subtract the overscan region of each image and trim the image to remove the overscan region. In the overscan fitting, try out 'chebyshev', 'legendre', and 'hermite', and 'polynomial' of various orders for the fitting function. Qualitatively try to summarize the differences between the functions (you can also look up and summarize the properties of these types of polynomials, but this is not required). Also try orders 1 through about 8, and make an argument for how large an order is necessary to fit the overscan region.

7. Combine the bias frames into a master bias frame using `CCDPROC.COMBINE`. By visual inspection or using image statistics, decide whether it is appropriate to combine all your bias exposures to make 1 master bias, or whether you need to restrict the input frames because of variations among your bias frames. Are there trends in your sequence of bias frames? Does either the level, or the pattern of the bias change throughout the night? If so, by how much? How much noise would you be adding to your data if you decide to do a bias subtraction in addition to your overscan subtraction. Do the subtraction of your master bias from all other science frames if you can justify that it is warranted.

8. Now examine the 'dark' images quantitatively and estimate the range of dark current (electrons per pixel per second) you see in the darks. WIRO Prime's gain is 2.5 e^-/ADU. Use Python to inspect the header and see the exposure time in each of the darks. Pixel values will vary! Be careful to avoid being fooled by cosmic rays, which will appear as large bright spots usually spread over a few pixels, whereas high dark current is usually isolated to just individual pixels.

9. Use `CCDPROC.COMBINE` to combine darks together to make a master dark. You will need to play with different combination parameters to see which one does the best job of eliminating cosmic rays. At a minimum, do

   - A straight average of all dark frames.

- A straight median of all dark frames.
- An average where outlier pixels are rejected if they are more than 3σ away from the mean.
- A median where outlier pixels are rejected if they are more than 3σ away from the mean.

Use the resulting RMS in the master image as an indicator of which combination procedure is best. Compare your master dark to your master bias to estimate the significance of the highest dark count pixels. How many times larger is the typical dark current than the noise (σ) in the master image?

The aim of this assignment is to characterize and subtract the overscan region, trim the frames so that a 2048x2048 data section remains, create a master bias, subtract it from all of the frames that you will need, and save the processed images with "otz" added to their filenames to indicate that these steps have been conducted. Your submission should be a Jupyter notebook that executes the tasks above for at least the following frames: biases, flats in all bands, darks, science frames for PG1633+099 and NGC6823.