Galfit Tips and Tricks

Written by Alexander Navarre - 12/11/2023

Galfit Software Home Page - Link

- Main hub for Galfit info. Software files contained here
- Example files contained here
- GALFIT Paper linked here (look at version 3.0)
- Instruction File linked here (similar to paper)
- Advisory and Top 10 Rules of Thumb linked here. Use these for troubleshooting
- FAQ used here. Use for troubleshooting
- There is a galfit facebook page, and I hear that the community still responds here if you need help.

The Galfit website has a *lot* of information, troubleshooting tips, and documentation. However, a lot of this will not be relevant to the average user (or at least it wasn't for me.) Below I list the stages of installing and using Galfit, with notes about the most important things to keep in mind.

1. Installation
    a. Look at the galfit installtion document I made
2. Understanding The Input File
    a. Keep a copy of the example input file found on the Galfit home page. It has all of the proper formatting and can easily be copied and edited for new projects.
    b. The top block (signified by the lettered rows), contains project info. The comments are self-expanatory.
        i. For C.) do not trust galfit to make a sigma image by itself. It doesn't do a good job. Either use a premade file (like .wht file), or leave it as none.
        ii. D.) is very important, as having a PSF model will drastically improve your results. There are premade HST PSFs found on STSCI, but these likely don't match your HST data perfectly due to steps in the data reduction pipeline. They can be a good first start though.
        iii. E.) This is the relative resolution (pixel size) of the PSF data to the science data.
        iv. H.) This must be in image coordinates. You can supply the entire image with 1 (XMAX) 1 (YMAX), but then Galfit will take forever to run. Use DS9 region with a box shape to find these numbers.
        v. I.) Make this the same size as your image region in H.
        vi. J.) For HST this is calculated from header properties here. Use ABmag.
        vii. Knan.) Found in headers

c. Each code block below the project info corresponds with 1 model. You can have as many as you like, but more models mean longer compute times and potentially overfitting.
    i. Skipping this model in the output image still runs the model computations. If you want a particular model to not change, you have to set all of the 1's to 0's.
d. Each model is defined by a function. These functions are described in the paper and the instruction file. The example input file has all of the examples written out explicitly.
    i. Sersics and Gaussians are the most common models.
e. At the end of your models, you need to have a sky model. Without this, Galfit runs into problems and can crash.
f. Everything past the sky model in the example input file is advanced and does not need to be understood for most applications.

3. Understanding the Constraints File

a. While not explicitly needed, the constraint file can help keep a model in check. Consider using if parameters keep going super high or super low.
b. Keep track of what order you have your models in the input file, because that numbering determines the component number in the constraint file.
c. See the example constraints file on the Galfit home page for the different constraint types.

4. Preparing your files

a. Any fits files that you input to galfit (science, psf) need to be in units of electrons (or counts). Galfit wont create good models if the scale of the data is too high or too low. Why? ¯_(ツ)_/¯ Check your data's header to see what units it is in.
    i. e-/s * header['EXPTIME'] = e-
    ii. c * header['GAIN'] = e-
    iii. Useful page for more complicated transformations [here](#)
b. If your source is close to the edge of the frame, or has data artifacts, consider making a bad pixel mask. This is a mask (of the same size as the image) of 1's and 0's, where bad pixels are marked and not used in the computations.
c. If you can't find a premade PSF that works well, you can try finding an isolated star in your data and using that. You will have to manually crop the image and make a new file though.
d. Mike Gladders has some legacy IDL code that takes multiple stars in frame and tries to combine them into a good PSF model for your specific data, but accessing IDL is a huge pain. Someone should try and re-create that code in python.

5. Running Galfit

a. Once your input file is set up with your initial guesses on parameter values, "galfit yourinputfile.input" runs galfit. You will see all of the iterations as Galfit tries to find the best solution.
b. Galfit will output a file (galfit.01) after running. This contains the last iteration before either 1.) It found a good enough solution and quit, or 2.) It reached 100 iterations before finding a good enough fit.
    i. In case 2, about 80% of the time more iterations (running galfit on the output file) won't get you a better solution. Galfit is stuck in a parameter space that loops or hardly moves. In this case, manually changing the parameters (giving them a kick) is required to get to a satisfactory solution.
    ii. Once you run "galfit galfit.01", galfit will output galfit.02, and every successive run will create a file with the next number in the sequence. It is important to write down which files contain your current best fit as you experiment.
c. Galfit will sometimes crash with a segmentation fault. Sometimes this is because parameters get too low or too high. Sometimes it's because galfit is black magic and incomprehensible to the human mind. In this case, it is best to use a constraint file, kick your parameters away from their current values, or change your model structure.

6. Analyzing the Results
    a. Galfit outputs a fits file with the name you specified in the output field. To open this in ds9, you have to go file -> open as -> multiple extension cube or multiple extension frames.
    b. The first extension is always blank. The second is the cutout of the input data specified by "image region to fit". The third is just the combination of components. The fourth is the second minus the third, and called the residual.
    c. By looking at the residual, you can tell where your models currently over or underfit the data.
        i. If your galaxies have small but non-negligible residuals in their cores, it is likely your PSF model isn't great.