nProFit user's manual

Bolivia Cuevas Otahola
Divakara Mayya

nProFit is a software tool developed in Python, Pyraf and Fortran aimed to fit dynamical models of observed surface brigthness profiles. We describe in this manual the requirements to install nProFit, as well as the operation and modules of the code

Pre-requisites

nProFit requires Python 2x in order to be executed. The following libraries are required to run nProFit:

- NumPy
- SciPy
- Matplotlib
- sys
- os
- glob
- Astropy
- fits
- Pyraf

A complete Iraf installation is required to be linked with the Pyraf library. We recommend the use of conda environment to install Python and the required library dependencies.

nProFit is suitable for Unix-like systems (MacOS and Linux).

A Fortran compiler is also required to run the nProFit dynamical models libraries.

A DS9 installation is also desirable, to plot the surface brightness profiles along with RGB snapshots of the analyzed objects.

Installation

1. Choose a directory where the nProFit code and its dependencies will be stored.
2. Copy the compressed nprofit.tar.gz file into the selected directory.
3. Extract the files using tar -xvf nprofit.tar.gz.
4. From the Unix-terminal run the install.sh script by using the command "sh install.sh". This scripts compiles the codes to create and fit dynamical models. If the compilation is successful, you will find a directory called "nprofit_library" and two fortran binary files named gen_iso and isothermal, respectively.
5. The script.sh execution may take a while, depending on your system architecture. Once the message "Execution completed" appears, the required dependencies will be created.
6. Create an alias to run nProFit from any directory. To do this, it is necessary in the first place to determine the type of shell you are currently running on. Type in the terminal "echo $SHELL". Follow the instructions below depending on your output:
   (a) For BASH:
      i. Find the absolute path to your installation directory by typing "pwd" in the terminal.
      ii. Copy such path and open your shell config file by typing "vi ∼/.bashrc" (use super user priviledges to do this, by adding "sudo" at the beginning of the command, if you cannot modify the file).
      iii. Insert the copied path by typing "i" in add a line as follows "export PATH=copied-path/:$PATH"
      iv. Insert an additional line by typing "Esc and "o". Add the following line "alias nprofit='python copied-path/nprofit.py'"
      v. Type ":wq" to save the changes and quit.
   (b) For Cshell:
      i. Find the absolute path to your installation directory by typing "pwd" in the terminal.
      ii. Copy such path and open your shell config file by typing "vi ∼/.cshrc" (use super user priviledges to do this, by adding "sudo" at the beginning of the command, if you cannot modify the file).
      iii. Insert the copied path by typing "i" in add a line as follows "setenv PATH ${PATH}:copied-path/"
      iv. Insert an additional line by typing "Esc and "o". Add the following line "alias nprofit python copied-path/nprofit.py'"

v. Type ":wq" to save the changes and quit.

(c) If the installation path and aliases declared in the previous step are correct, you will be able to run nprofit in each subsequent terminals you open, by typing nprofit

Running nProfit

nProFit requires a set of input files, including a configuration file, and a file describing some image properties, for a given number of filters (depending on the user's needs). If additional actions such masking contaminant sources are required, additional files will be required by nProFit to carry out such a procedure (See sec. Masking contaminant sources). The most important file is the configuration file. We include sample files to perform a test run with mock data in the folder "example" in the installation path. You can modify the file named "input" to run the code according to your needs. The name "input" can be changed as well, as required by the user. The test run, can be executed by typing in the linux terminal "nprofit input". If the latter command results in an error, return to the installation section and check that the path and alias are declared properly.

If the execution of the test sample in the "example" folder is complete, files with the preffix "derived_parameters_" will be generated, along with the directory "plots_profiles" and the auxiliar directory "nprofit_data" storing the trimmed individual images of the objects along with its extracted background-subtracted profiles.

Image properties

In the "input" file, the user defines the name of file containing the image properties, in the following columns:

1. Filter
2. Zero point magnitude for each filter
3. Image name for each filter
4. DS9 color code (which can be R, G or B, selected by the user, this option is used for DS9 to plot the RGB snapshots)
5. Sky file name per filter (if the user chose to measure the background value, this option can be ignored, and remain as in the example file. In case the user wants to include a given file with these values, the name of such a file should be written in this column per each listed filter)
6. Ellipticity and PA file (as in the previous option)

7. PSF files names for each filter
8. Image pixel scale
9. Image scale in arc/seconds
10. Mass-to-light ratio (this parameter is used by nProFit to compute the model-derived parameters. If this value is unknown, assume one)
11. Solar absolute magnitude in each filter.

The number of rows of this file will be given by the number of filters for analysis. For example, if 3 filters are used, this file should contain 3 rows.

Trimming the images

One of the most salient nProFit features is its capability to extract the surface brightness profiles of several objects in the same image. To do this, nProFit trims the provided image in user given sizes, which can be of constant size for each object (as in the "input" file, with value of 100 in the third row, referring to 100 pixels box sizes) or different sizes for each object (defined in a separate file with the IDs in the first column and box sizes in pixels in the second column). For constant sizes choose option 1, and for customized different sizes option 2 in the second row. The boxes for trimming are defined by nProFit, with centers in the coordinates given by the user in row 4. We include a sample in the file coords.dat, containing 3 columns, in the first one the IDs and the second and third one the x and y coordinates of the center of each object. The sample file is in image (units), however WCS coordinates system is also accepted by nProFit. The former system is selected in row 5, selecting option 1, and for the latter option 2.

Considering that, in some cases, the user would like to carry out some modifications, option 6 can be set to "no" if the trimming was already performed, avoiding to trim the images unnecessarily.

At this point, the outputs will be saved to the newly generated "nprofit_data" directory, sharing the same preffix as the original image, with a suffix containing the ID and filter of each object.

Background values estimation and subtraction

Options in rows 7 to 10, are devoted to the background values estimation and subtraction. The user can choose to measure the background value for each object by using nProFit, setting row 8 to "yes" and row 10 to "no", or provide a file containing the background and

rms values for each object for each filter, setting row 8 to "no" and row 10 to "yes". In the latter option, a list of the files containing the background values information will be required, which will be given in column 5 in the image properties file ("filters.dat" in the test run). The background values will be subtracted to the previously trimmed images if row 7 is set to "yes" and will not be subtracted otherwise.

If the user choose to measure the background values, nProFit offers to methods, based on the statistical median estimator. The first one is based on a complete scan of the image based on a σ-clipping technique. This method, *instat* is available as an Iraf task. To choose this option, you should install the Inaoe Iraf package. To do this, go to Inaoe IRAF Package. The second option, is the default one, which consists of estimating the background value in boxes with sizes 10% the complete image, computing the median value. Once the background value are estimated, nProFit generates one file per filter, containing the background median and dispersion values for each object. The image properties file ("filters.dat" in the test run), contains the names by default, used by nProFit, to store the background data. This ensures that subsequent runs can be performed without measuring the background values, several times, and resuming the runs in further steps.

nProFit saves the subtracted images in the "nprofit_data" with the suffix "_sub.dat".

Masking contaminant sources

Rows 11 and 12 are devoted to set the properties for masking contaminants. This used can be either used in a first run or in subsequent runs. If the objects under analysis are in crowded regions, the user may benefit from the procedure performed by nProFit to compute a fitting radius, which is the optimal radius used by nProFit to remove the contribution by contaminant sources. However, in some cases, a masking procedure will be required to improve the results. The user can choose to carry out such a procedure by setting row 11 to "yes". In this case, the user need to provide a file with the centers and sizes of the contaminant sources in each trimmed image, as the one provided in Tab. 1, for 3 objects, the first one with 4 contaminant sources, the second one with 2, and the third with none. The first columns, contains the objects identifiers, whereas the second and third one a fitting radius (which can be left by default) and the number of contaminants to mask. The latter value is used by nProFit, to determine the number of columns to read, to generate the mask images. The subsequent columns contain as many values, as 3*Nmask, arranged as (x,y,r), i.e., the x and y coordinates of each contaminant source and its radius. This file can be easily generated by the user, using DS9, through a visual inspection of the sources, in the individual trimmed images, previously generated by nProFit in the folder nprofit_data. The centers and sizes in the mask files are given in pixel units.

nProFit creates mask pixel images with .pl extension and stores them into the "nprofit_data"

Table 1: Sample masks information file for three objects.

ID	RAD	N_{MASK}	X_{M1}	Y_{M1}	R_{M1}	X_{M2}	Y_{M2}	R_{M2}	X_{M3}	Y_{M3}	R_{M3}	X_{M4}	Y_{M4}	R_{M4}
1	51	4	33.1	93	6.7	98.1	84	12.9	40.8	61	12.8	17	3	33.1
2	51	2	53.9	89	14.3	62.1	36	10.4						
3	51	0												

directory if the user chose to mask the contaminating sources.

Position angle and ellipticity estimation

Rows 13, 14, 16 and 17 are devoted to set the properties of the ellipticity and P.A. estimation for isophotal fitting. The user can choose either to supply a file with ellipticies and position angles by setting row 13 to "yes" and 14 to "no", or compute such geometrical parameters by fitting elliptical isophotes using ellipse by setting row 14 to "yes" and 13 to "no". nProFit, computes such values, determining the radius at which, the obtained ellipticity value stabilizes.

The ellipticity values are subsequently fixed to perform an isophotal fitting to extract the surface brightness profiles. The user can choose to constrain the ellipticity values, setting an upper limit by setting row 16 to "yes". The upper limit should be expressed in row 17.

Files with the suffix "_sky_sub_calc_ellip.tab" are stored in the "nprofit_data" directory and used by nProFit to compute the ellipticity.

Surface brightness profiles extraction and dynamical models fitting

The user can choose whether to perform an isophotal fitting to extract the surface brightness profiles (SBPs) of each object, by setting row 15 to "yes". From the extracted background-subtracted SBPs, nProFit, computes the fitting radius, choosing the minimum value between the inflexion point radius and the radius at which the intensity value is above 3 times the background dispersion value. A summary of the computed fitting radius, as the corresponding plots are contained in the folder rfit_plots inside the nprofit_data.

Once the SBPs are extracted, nProFit creates input files for the isothermal code, devoted to fit dynamical models to the SBPs. To proceed to fit models to the observed SBPs, nProFit requires PSF data for each filter. By setting row 18 to "yes", the user indicates nProFit that the models will be convolved with a PSF. In the next release we will include predefined PSF functions. At the time, nProFit only has the option to read PSF data

provided by the user (row 9 set to 1). nProFit performs an isophotal fitting for each PSF image, and prepares such data for fitting.

The SBPs prepared for the fitting procedure are saved into the files preffix "object_" followed by the filter and id. Files for Moffat-EFF, King and Wilson models fitting contain the preffix moffat_input, king_input and wilson_input, respectively. The files with suffix "_psf_file.dat" contain the PSF data files prepared for model fitting. Some addition list files, are created for nProFit purposes to be able to identify the files corresponding to each object and filter.

Considering that the most time-consuming module of nProFit is the one corresponding to model fitting, the user can choose to prepared the data and fit it automatically (row 20 option 1, and row 25 option 1), or carry out the fits later (row 20 option 1, and row 25 option 2). nProFit also provides the option to compute model-derived parameters from previously performed fittings (row 20 option 2 and row 25 option 3). The user can also prepare a script to fit the models using high-computing facilities (row 20 option 2 and row 25 and option 2).

The user can choose Moffat-EFF, King and Wilson simultaneously, by setting rows 21, 22 and 23 to "yes". It is also possible to choose one model only or two. The basic structural parameters, are summarized in the files with preffix "moffat_pars_", "king_pars_" and "wilson_pars_" for Moffat-EFF, King and Wilson, respectively, along with their corresponding errors, computed from the confidence intervals.

Given that in some cases, the nProFit library could be generated outside the installation path (if the user moves it away from the installation path), it is necessary to provide nProFit with the absolute path containing the dynamical models libraries generated with the install.sh script. The path should be written in row 24.

The user can choose to plot the resulting fits along with RGB snaphots by setting row 26 (upon previous declaration of the DS9 path in row 27). The resulting plots will be saved into the directory plots_profiles on the profile_plots.pdf file.

Finally, from the basic structural parameters, the user can choose to compute model-derived parameters by setting row 28 to "yes". The user can choose to perform this task alone, from previously fitted dynamical models by setting row 28 to "yes" along with row 25 to 3, row 20 to 3, row 7,8 14,15 to "no". The derived parameters are summarized in files with the preffix "derived_parameters_moffat_", "derived_parameters_king_", and "derived_parameters_wilson_" for for Moffat-EFF, King and Wilson, respectively, along with their corresponding errors obtained by propagating the errors on the basic structural parameters.

Creating a mock sample to test the code

Along with nProFit, we provide mksample, a simple code to simulate a mock sample of objects following Moffat-EFF, King or Wilson profiles, centered at user-defined coordinates, ellipticities and given by pre-defined structural parameters. mksample is contained in the mksample folder, along with a test sample, to generate 9 objects.

Troubleshooting

We encourage the users to carefully select the row options as explained in this manual to avoid execution errors. Some more options can be explored. In case of error, check whether two options cannot be simultaneously set to "yes" or "no" as described previously.

In case of any Iraf-related errors during the execution, remove the "nprofit_data" folder and execute nProFit again.

For further inquiries please e-mail to bolyumbra@gmail.com