
Calculation of Luminosity Profiles for a Sample of Galaxies

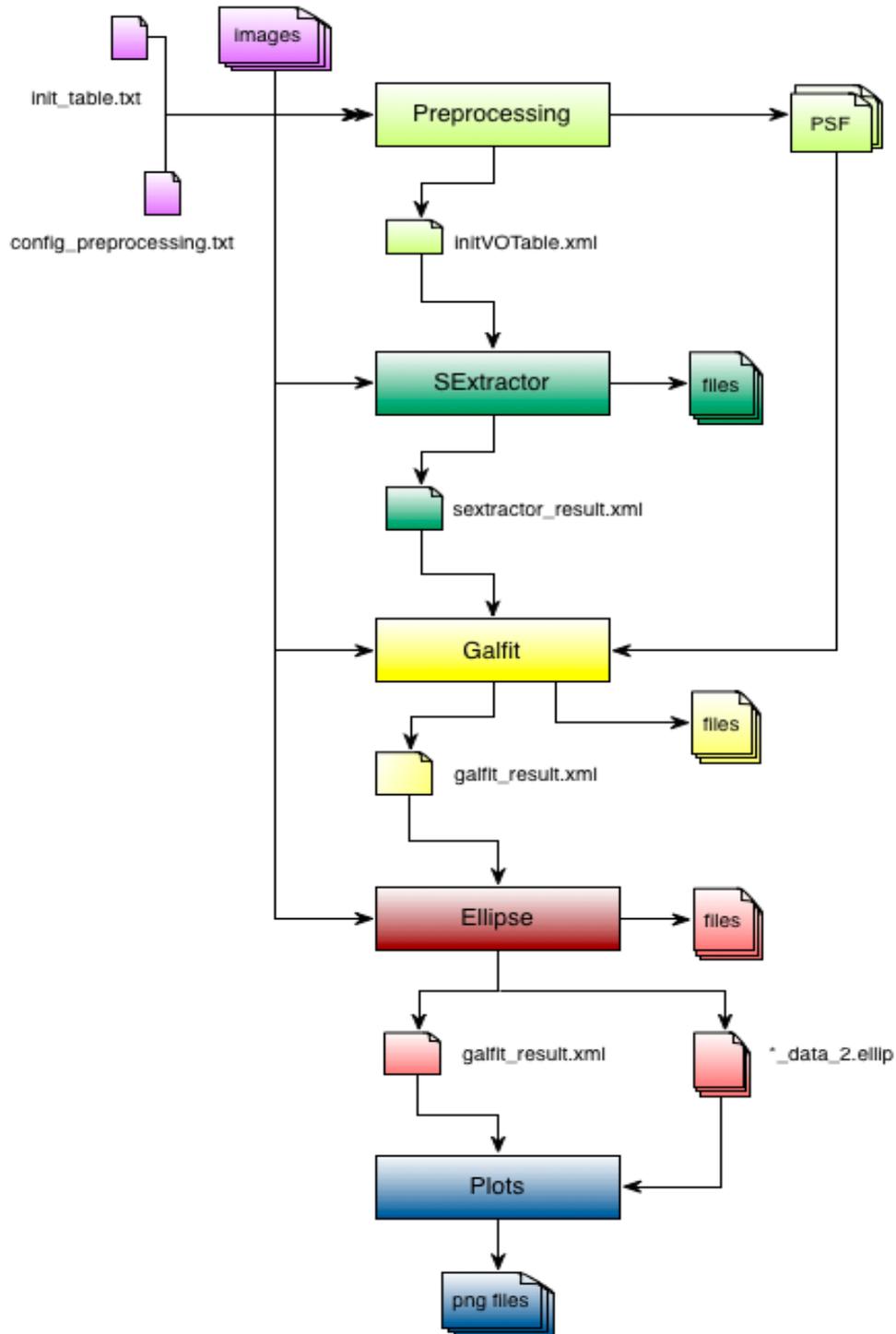


Table of contents

1. Materials and methods3

 1.1 Calculation of luminosity profiles3

 1.2 Photometry modelling software3

 1.2.1 SExtractor3

 1.2.2 GALFIT4

 1.2.3 ELLIPSE IRAF4

 1.3 Execution environment.....4

 1.4 Workflows.....5

 1.4.1 Preprocessing5

 1.4.2 SExtractor6

 1.4.3 Galfit.....7

 1.4.4 Ellipse7

 1.4.5 Plots8

2. Results9

 2.1 Luminosity profiles9

Appendix A – Graphical representation of workflows10

Appendix B – Luminosity Profiles RO structure and content12

Appendix C – Results of Luminosity Profiles RO.....24

List of Figures

Figure 1: Luminosity profiles for galaxy CIG 33 (anti-truncation)..... 24

Figure 2: Luminosity profiles for galaxy CIG 281 (truncation)..... 25

Figure 3: Luminosity profiles for galaxy CIG 520 (normal)..... 26

1. Materials and methods

1.1 Calculation of luminosity profiles

The study of galaxy properties is deeply related with the electromagnetic energy that is emitted and absorbed by their different components. Thus, looking at different regions of the spectrum, it is possible to study each one of them. In the optical region, we identify the stellar component, so using optical images and spectra we can study the stellar population age, the chemical composition of the stars, their distribution within the galaxy, and even their internal motions.

To study the distribution of stars within the galaxy, we use the luminosity profiles, which give information on how tight are the stars as we move away from the centre of the galaxy. In order to extract these profiles, an optical image of the galaxy (calibrated in physical units) is needed. From this image, we measure the quantity of luminosity per unit area (surface brightness) that the galaxy emits at different distances to the galaxy centre. This luminosity is related with the quantity of stars that is enclosed in that area. The distribution of surface brightness versus radius is different for each part of the galaxy, and therefore we can identify its components, such as a bulge, a disk, a bar, etc. Moreover, the shape of the function that fits each of these structural components provides information about how the galaxy formed and evolved.

Different tools can be used to obtain the information described above. In this second RO developed we have used two complementary software packs: GALFIT makes a 2-dimensional modelling of each galaxy component and calculates the luminosity profiles of the modelled components, ELLIPSE extracts the luminosity profile of the observed galaxy. The comparison between modelled and observed luminosity profiles will allow us to extract the information needed to infer the galaxy evolution and the formation of its components. The luminosity profiles also provide information about the outer parts of galaxies, which is a very hot topic.

1.2 Photometry modelling software

1.2.1 SExtractor

SExtractor¹ (Source-Extractor) is a software pack that builds a catalogue of objects from an astronomical image. In order to produce this catalogue of structural and photometric values for each object identified in the image, SExtractor needs a configuration file with some information about the image and its calibration parameters. This configuration file also contains a large number of control parameters that will be used for the detection of objects, including a minimum area for the objects detected, a sigma level above the background, and a deblending parameter for separating overlapping objects. In our specific case, each galaxy image only contains one big object, and all these parameters are left by default.

¹ <http://www.astromatic.net/software/sextractor>

1.2.2 GALFIT

GALFIT² is a data analysis algorithm that fits 2-D analytic functions to brightness density distribution of galaxies registered in digital images. The functions used are some of the most commonly found in astronomy literature, such as exponential, Sérsic or de Vaucouleurs. One of its main features is that it allows for simultaneous fitting of an arbitrary number of components, and combination of the above functional forms.

One of the scientific applications of GALFIT is to model big nearby galaxies, in order to probe the detailed structures of these galaxies. With the aim to study the nearby isolated galaxies of the AMIGA³ sample, we have designed a workflow to make a bulge-disk-bar decomposition using GALFIT on images provided by SDSS DR7⁴. Since GALFIT needs some initial parameters, the workflow is designed to take them from the results produced in a previous one using SExtractor. The results of GALFIT include a tabular file with the fitted parameters of each structural component as well as a multi-layer FITS file composed of 2-D images including *observed galaxy*, *modelled*, and *residuals* images, which can also reveal evidence of other fine structures.

1.2.3 ELLIPSE IRAF

The ELLIPSE task works under the IRAF⁵ environment and is used to fit elliptical isophotes in optical images of galaxies, i.e. looks for elliptical curves in the galaxy image fulfilling all points having the same luminosity per unit area. The task reads one 2-dimensional image section and produces as main output one tabular file that contains parameters for each fitted isophote. The task starts from a first guess elliptical isophote defined by user provided approximate values for the X and Y centre coordinates, ellipticity and position angle of the galaxy. We have designed a workflow that takes these initial values from the results produced in a previous one using SExtractor, and fits the isophotes to modelled images provided by GALFIT as well as to the observed images. After fitting the ellipse that corresponds to a given value of the semi-major axis, the axis length is incremented/decremented following a pre-defined rule. At each step, the starting ellipse parameters are taken from the previously fitted ellipse that has the closest semi-major axis length to the current one. From the table given by ELLIPSE, we subsequently plot the values representing the surface brightness versus radius of each elliptical isophote. i.e. the luminosity profile.

1.3 Execution environment

The first requirement to run the workflows provided by both ROs is Taverna Workbench⁶ 2.4 or higher. AstroTaverna⁷ (Taverna plugin) is also needed in order to get functionalities related with Virtual

² <http://users.obs.carnegiescience.edu/peng/work/galfit/galfit.html>

³ <http://amiga.iaa.es>

⁴ <http://www.sdss.org/dr7>

⁵ <http://iraf.noao.edu>

⁶ <http://www.taverna.org.uk/download/workbench/2-4/>

Observatory web services queries and management of standard VOTable data formats. In general, the execution environment is a Linux distribution including Python⁸ 2.x and a bash shell. The following list shows other specific dependencies for the main workflows:

- Preprocessing
 - Python
 - read_PSF⁹
 - AstroTaverna
- SExtractor
 - SExtractor 2.8.6 or higher
 - AstroTaverna
- Galfit
 - Galfit
 - AstroTaverna
- Ellipse
 - IRAF¹⁰ including *tables*, *stdas* and *ellipse* modules
 - Python with *pyraf*¹¹ package
 - AstroTaverna
- Plots
 - Python including *matplotlib*¹² package
 - AstroTaverna

1.4 Workflows

1.4.1 Preprocessing

This workflow compiles the information needed by subsequent workflows in a VOTable and edits metadata information comprised in image files so that they be analysed afterwards. The user needs to provide a configuration file with the absolute path of the folder where the RO is located in the local file system, and a set of paths relatives to it where the original image files may be found, and where the files generated by the workflows will be stored. Another input is a tabular ASCII file where each row contains information of an image: the coordinates, the image filename, SDSS metadata uniquely identifying the

⁷ <http://wf4ever.github.com/astrotaverna/>

⁸ <http://www.python.org/>

⁹ http://www.sdss.org/dr7/products/images/read_psf.html

¹⁰ http://www.astro.uson.mx/~favilac/downloads/ubuntu-iraf/iso/IRAF_Ubuntu.iso

¹¹ http://www.stsci.edu/institute/software_hardware/pyraf

¹² <http://matplotlib.org/>

image, a flag indicating if the galaxy is barred or not, and the filenames for the PSFIELD and PSF files created in this workflow.

The workflow is composed by two nested workflows. One makes up the absolute paths of the files, extracts the band and the resolution of the images from the images headers and queries the *sdssdr7-field*¹³ service to obtain the properties of the fields. It also downloads the PSFIELD file from the SDSS web archive, runs the READ_PSF program on the PSFIELD file in order to build the PSF file, and edits its header image to add needed information in a later analysis process. Finally, it stores those data and other provided by the user in VOTables. The second one compiles those VOTables into a single one together with the coordinates and barred flag provided by the user, and it adds three more columns, which are calculated using the values of other columns.

Taverna runs out of memory when the workflow is preparing the sample of 282 images of this use case. This problem can be solved increasing the maximum of the 400MB default memory in the Taverna configuration file, or changing the preferences so that it does not keep the provenance in memory. In order to avoid changing the default settings of Taverna, the initial table with 282 rows may be also split in four parts, the workflow executed four times, and concatenate those four VOTables into a single one with a short shim workflow.

1.4.2 SExtractor

The main idea of this workflow consists of using SExtractor to estimate a set of parameters that are needed in later stages of the experiment. The parameters to be inferred are initially declared in a configuration file located in the user file system. Other configuration files are produced at run time during the execution of the workflow.

The workflow receives as inputs a VOTable that encapsulates the information needed for its execution, a template to create SExtractor configuration files and vocabulary for them, as well as several string values designating the column in the VOTable that provide filenames for images, configuration files and result files.

Although the configuration files of SExtractor have a defined structure, the values of its parameters depend on the galaxy. Therefore, a template pattern to create the files is chosen (key values in the template are replaced by values from the VOTable).

The workflow is composed of four nested workflows that are sequentially executed:

- Creation of configuration files: It fills the template for each galaxy present in the input VOTable and it checks that the terms used in the template and in the VOTable match the terms in the vocabulary.
- SExtractor execution: It runs SExtractor on each image/galaxy declared in the VOTable.
- Cat SExtractor results: SExtractor results are organized in different files. This workflow gathers all the information into a single VOTable that is joined to the original one.

¹³ http://vo.astronet.ru/sai_cas/conesearch?cat=sdssdr7&tab=field

- Addition of new information: It uses SExtractor results to insert new calculated columns in the VOTable. The resulting VOTable is the final result of the main workflow.

1.4.3 Galfit

This workflow uses GALFIT to fit Sérsic and exponential-disk functions to the brightness distribution registered in the images of galaxies, in order to build a model of each galaxy. The execution of GALFIT requires a configuration file with information related to the image, the observation and the adjusting functions. It returns a text file with new values for the adjusted parameters and a multi-layer FITS file composed of 2-D images (*observed galaxy, model and residuals*).

The main inputs of this workflow are a set of template files and vocabulary for them, partition criteria to process barred and non-barred galaxies in a different way, and a VOTable that contains data and filenames. It also requires the specification of the VOTable column names that provide the information on where to store GALFIT results in the file system. The output consists of a VOTable as well as the standard products and intermediate files delivered by GALFIT executions.

Depending on the image and the configuration parameters, GALFIT may fail or crash. The workflow manages these errors by filling the VOTable with error values (-99 and -999) in order to make explicit in the VOTable that GALFIT failed.

The workflow is composed of six nested workflows that are executed sequentially:

- Creation of GALFIT configuration files: It fills the template for each galaxy found in the input VOTable and checks that the terms used in the template and in the VOTable match those in the vocabulary. The template is defined to make a good estimation of the centre.
- GALFIT execution: It runs GALFIT for each galaxy found in the VOTable.
- Addition of new information: It retrieves GALFIT results in order to create a VOTable that is joined to the original one.
- Creation of GALFIT configuration files for a second iteration: It fills a new template for each galaxy. The template is different depending on whether the galaxy has a bar (partition criteria). The template is defined to make a fine grain fitting by using the results of the first execution.
- GALFIT execution: It runs GALFIT using the second configuration files.
- Addition of new information: It retrieves GALFIT results in order to create a VOTable that is joined to the previous one and it adds new calculated columns that are needed in later stages of the experiment.

1.4.4 Ellipse

This workflow calculates the ellipses that better match the brightness distribution of a galaxy by doing two iterations. In the first one the centre and the outer ellipse are approximated since there might be bars that affect to the inner ellipses. In the second one, a fine grain fitting is performed and output files with the data defining the ellipses are produced.

ELLIPSE execution depends on IRAF. In particular, the workflow uses *pyraf* as an intermediate python package to call IRAF functionalities. IRAF requires setting up an IRAF folder, where tasks and procedures have to be called from this folder. For this reason, IRAF path is one of the inputs of this workflow. The remaining inputs are a VOTable, two templates, a vocabulary and several column names to point out the VOTable columns that provide the required information.

Depending on the image and the SExtractor results ELLIPSE may fail or ask for new centres. The workflow detects this situation if ELLIPSE result files are not created and then it fills the VOTable with error values (-999) that allow filtering the table.

The workflow consists of the following steps:

- Filter VOTable: Although this workflow only needs information from SExtractor execution, the RO protocol is designed to add workflow results incrementally into the VOTables. For this reason, the VOTable used as input in this workflow is the one coming from *galfit* workflow, removing the rows (galaxies) that failed during GALFIT execution will improve the performance.
- Create ELLIPSE scripts: It uses the template pattern to create a *pyraf* script to run the first ELLIPSE iteration.
- Run first ELLIPSE iteration. It executes the previous *pyraf* scripts.
- Fill VOTable with ELLIPSE results: It retrieves information from ELLIPSE results and creates a VOTable, add new calculated columns, join the VOTable to the original one and filter the rows where ELLIPSE has failed.
- Create second new ELLIPSE scripts to obtain more refined results.
- Run second ELLIPSE iteration.
- Get VOTable without errors: It detects the galaxies where second iteration failed and returns a VOTable without these galaxies.

Although several intermediate files are created during the execution of this workflow, the final outcome is the last VOTable and ELLIPSE data files produced in the second iteration.

1.4.5 Plots

This workflow represents the luminosity profile for each galaxy found in the input VOTable. The generated plots are based on GALFIT and ELLIPSE results. In particular, the plots contain:

- Modelled bulge luminosity profile
- Modelled disk luminosity profile
- Modelled bar luminosity profile (if the galaxy has a bar)
- Combined modelled luminosity profile
- Observed galaxy luminosity profile profile (including error bars)
- Residual values for the profiles between the model and the image

Some of these luminosity profiles may be found in Appendix D.

2. Results

2.1 Luminosity profiles

The Luminosity Profiles RO developed for obtaining the luminosity profiles was applied to a subsample of 90 isolated galaxies present in the AMIGA Catalogue. We compared the results obtained for all the galaxies in three different photometric bands from the SDSS: g, r, and i-bands. The 2-D models of bulges obtained with GALFIT confirm that most of our galaxies have pseudobulges according to the Sérsic index, which is indicative of secular evolution processes. The same result is obtained in all the three bands, making the result more robust.

The analysis of the luminosity profiles reveals that some of the galaxies present a truncation in the outer parts, which is related with a change in the disk luminosity profile. We found also anti-truncations and normal profiles for other galaxies in our sample, which rejects the interaction with other galaxies as the mechanism responsible for the formation of some of these profiles. Some of these luminosity profiles may be found in Appendix C.

Appendix A – Graphical representation of workflows

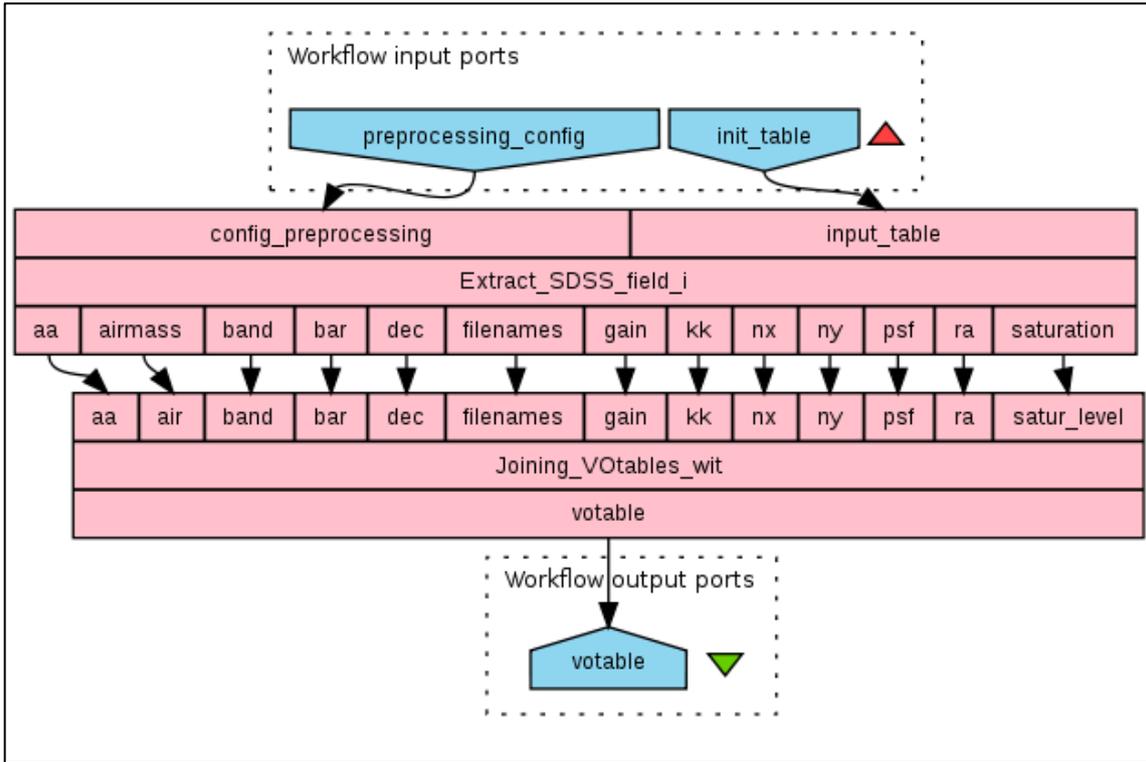


Figure 1: Preprocessing Workflow

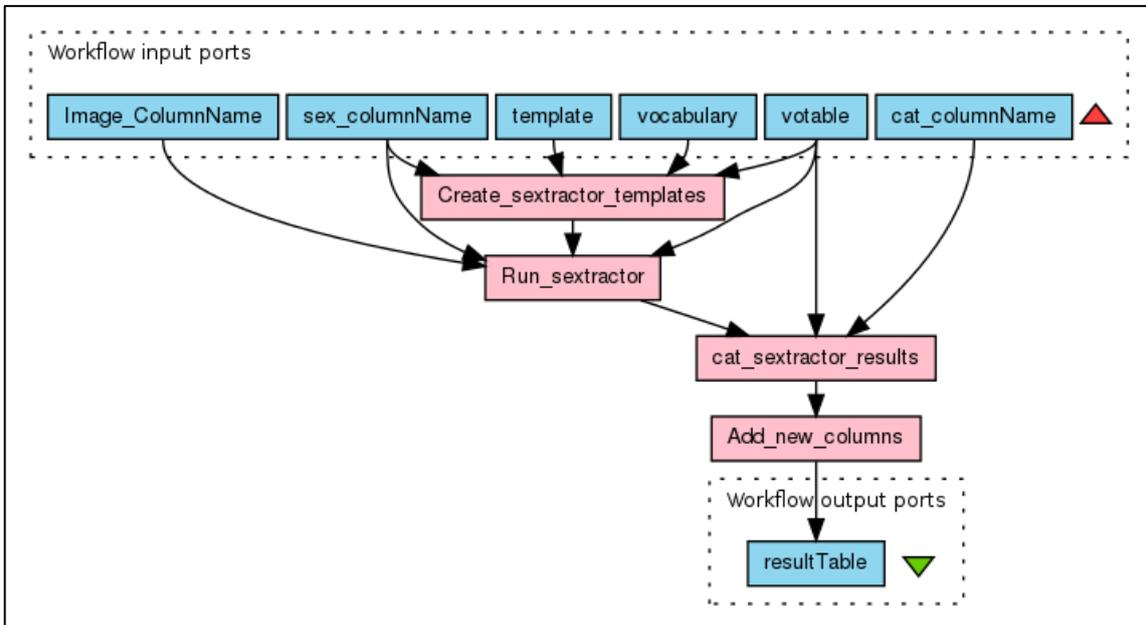


Figure 2: SExtractor Workflow

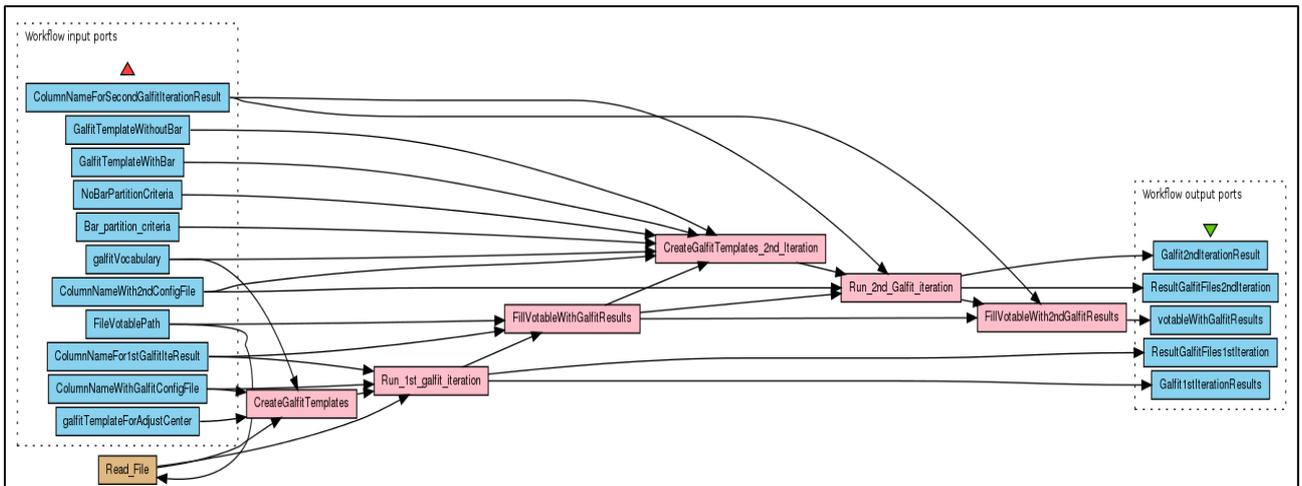


Figure 3: Galfit Workflow

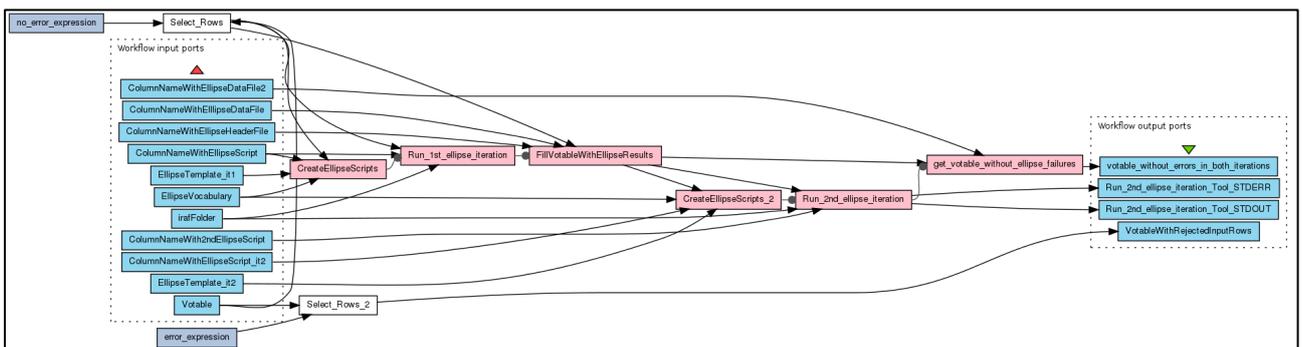


Figure 4: Ellipse Workflow

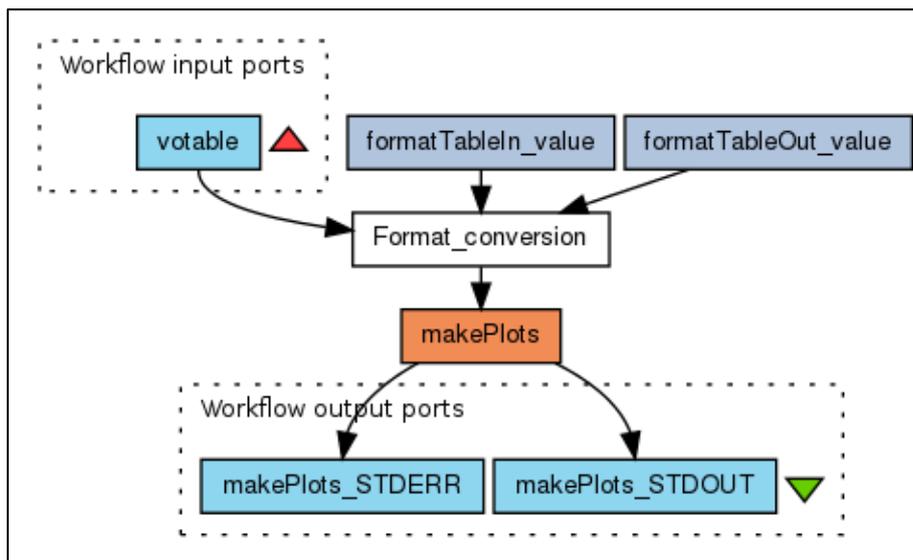
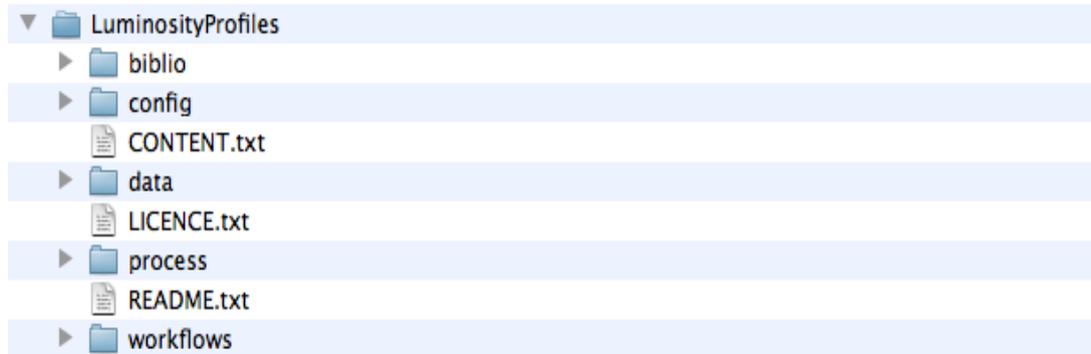


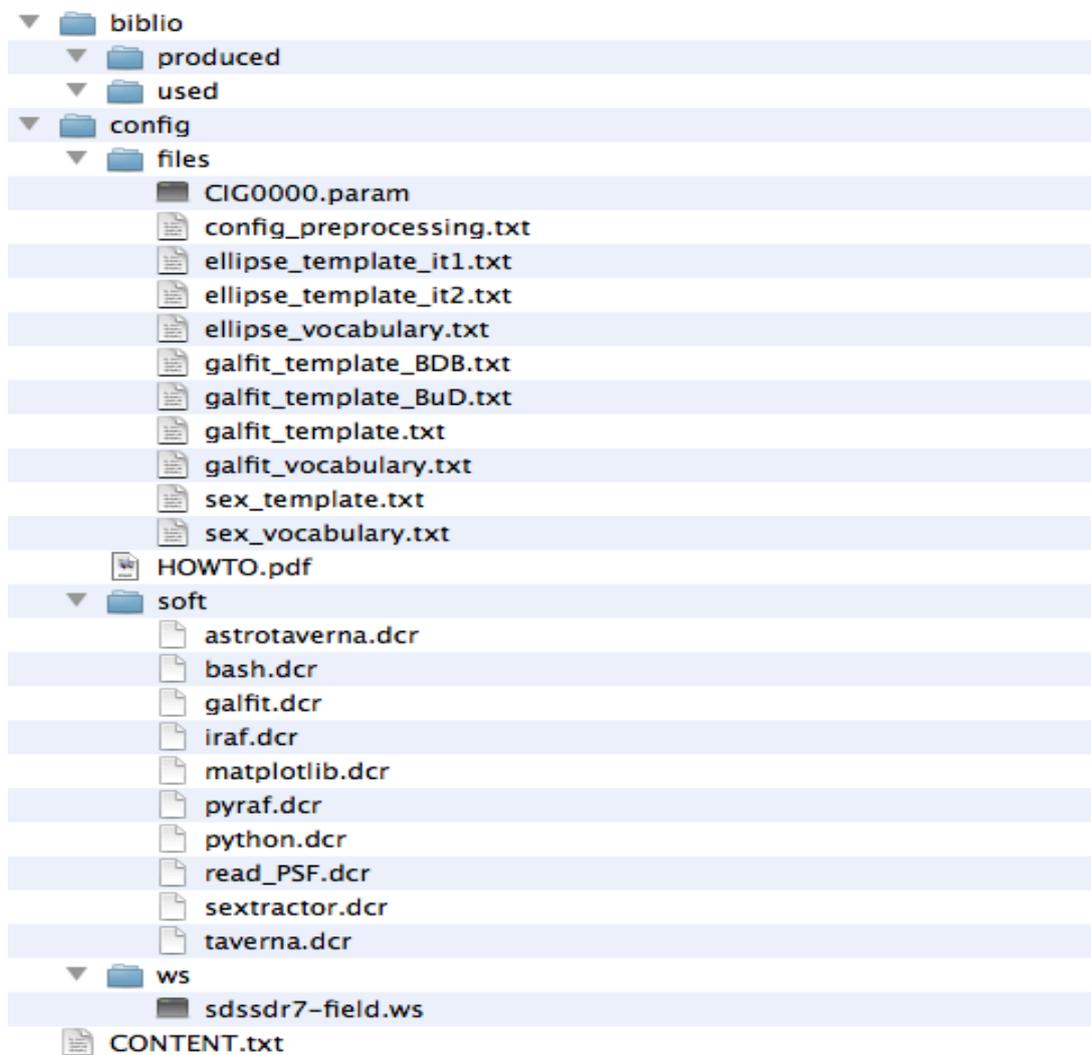
Figure 5: Plots Workflow

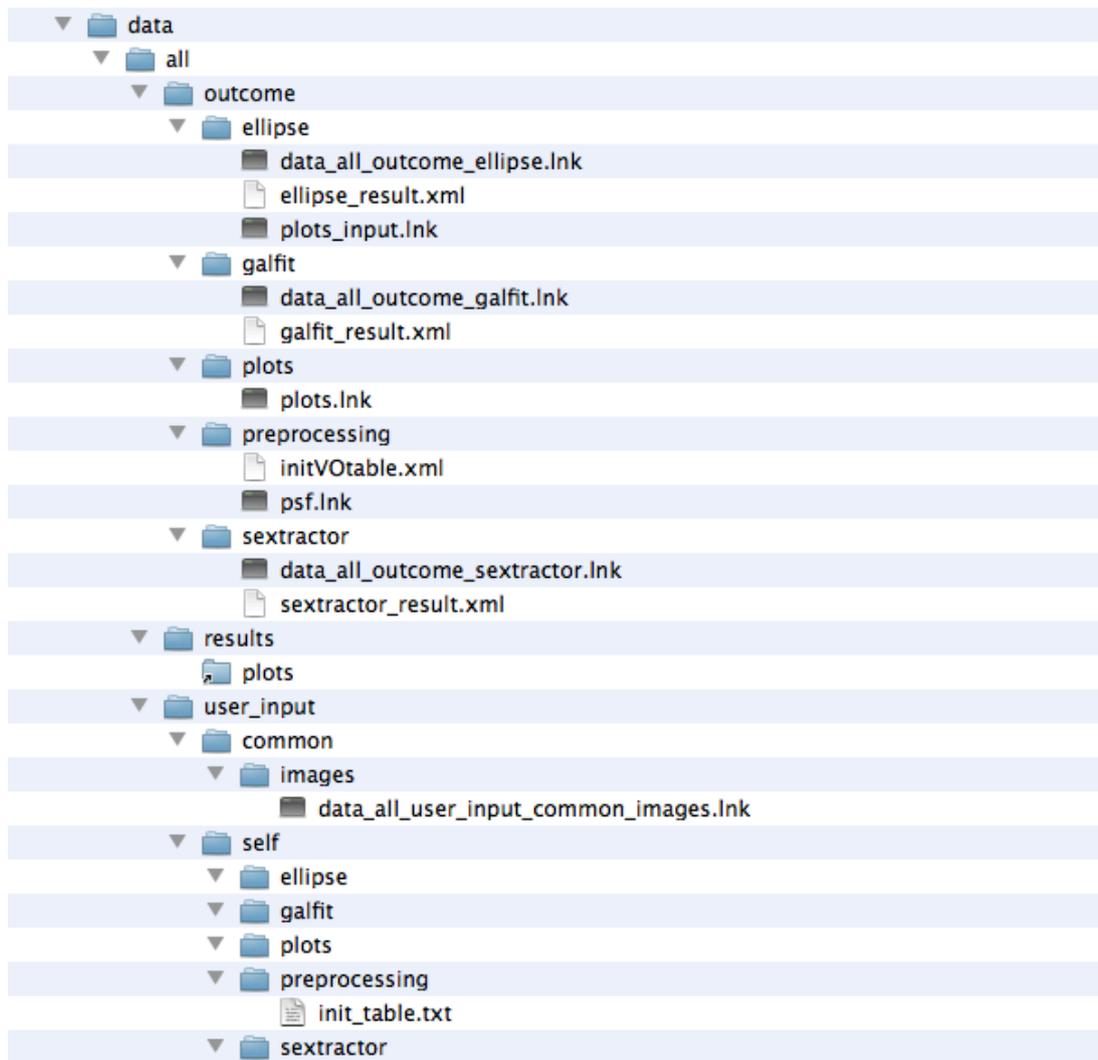
Appendix B – Luminosity Profiles RO structure and content

ROOT



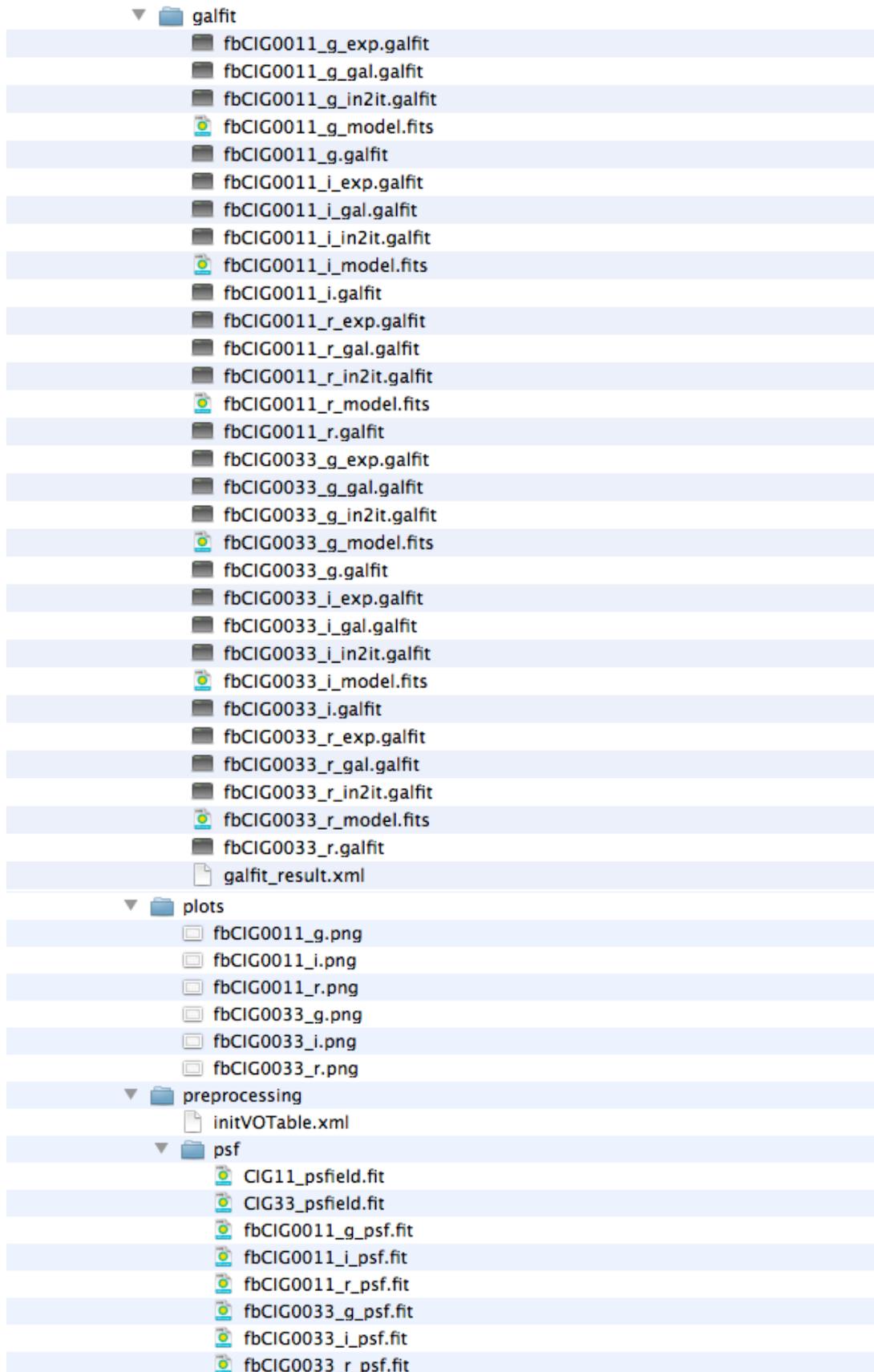
CONFIG



DATA

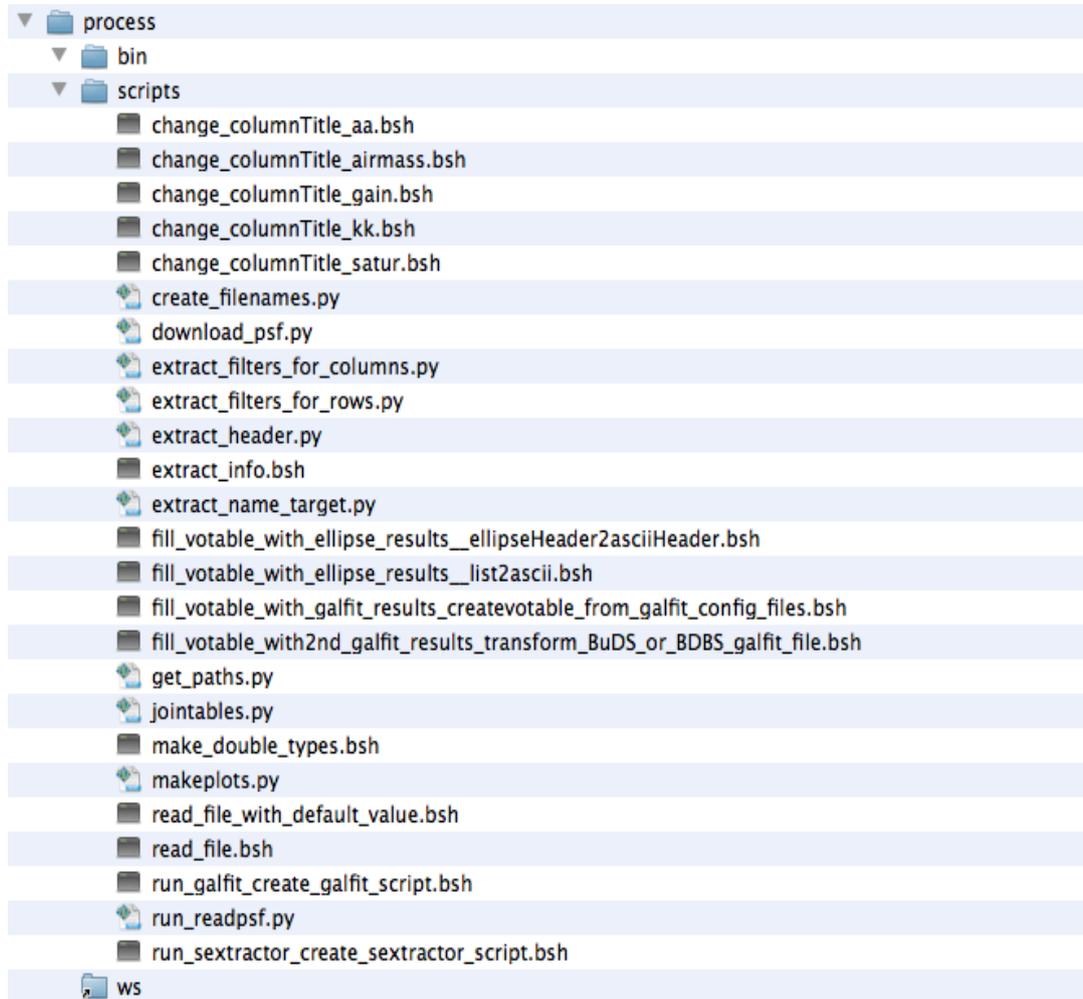
SUBSETS

- ▼ data
 - ▶ all
 - ▼ subset
 - ▼ outcome
 - ▼ ellipse
 - ellipse_result.xml
 - fbCIG0011_g_2.tab
 - fbCIG0011_g_cd_2.ellip
 - fbCIG0011_g_cd.ellip
 - fbCIG0011_g_data_2.ellip
 - fbCIG0011_g_data.ellip
 - fbCIG0011_g_ellip1.py
 - fbCIG0011_g_ellip2.py
 - fbCIG0011_g.tab
 - fbCIG0011_i_2.tab
 - fbCIG0011_i_cd_2.ellip
 - fbCIG0011_i_cd.ellip
 - fbCIG0011_i_data_2.ellip
 - fbCIG0011_i_data.ellip
 - fbCIG0011_i_ellip1.py
 - fbCIG0011_i_ellip2.py
 - fbCIG0011_i.tab
 - fbCIG0011_r_2.tab
 - fbCIG0011_r_cd_2.ellip
 - fbCIG0011_r_cd.ellip
 - fbCIG0011_r_data_2.ellip
 - fbCIG0011_r_data.ellip
 - fbCIG0011_r_ellip1.py
 - fbCIG0011_r_ellip2.py
 - fbCIG0011_r.tab
 - fbCIG0033_g_2.tab
 - fbCIG0033_g_cd_2.ellip
 - fbCIG0033_g_cd.ellip
 - fbCIG0033_g_data_2.ellip
 - fbCIG0033_g_data.ellip
 - fbCIG0033_g_ellip1.py
 - fbCIG0033_g_ellip2.py
 - fbCIG0033_g.tab
 - fbCIG0033_i_2.tab
 - fbCIG0033_i_cd_2.ellip
 - fbCIG0033_i_cd.ellip
 - fbCIG0033_i_data_2.ellip
 - fbCIG0033_i_data.ellip
 - fbCIG0033_i_ellip1.py
 - fbCIG0033_i_ellip2.py
 - fbCIG0033_i.tab
 - fbCIG0033_r_2.tab
 - fbCIG0033_r_cd_2.ellip
 - fbCIG0033_r_cd.ellip
 - fbCIG0033_r_data_2.ellip
 - fbCIG0033_r_data.ellip
 - fbCIG0033_r_ellip1.py
 - fbCIG0033_r_ellip2.py
 - fbCIG0033_r.tab



- ▼ folder sextractor
 - fbCIG0011_g_seg.fits
 - fbCIG0011_g.cat
 - fbCIG0011_g.sex
 - fbCIG0011_i_seg.fits
 - fbCIG0011_i.cat
 - fbCIG0011_i.sex
 - fbCIG0011_r_seg.fits
 - fbCIG0011_r.cat
 - fbCIG0011_r.sex
 - fbCIG0033_g_seg.fits
 - fbCIG0033_g.cat
 - fbCIG0033_g.sex
 - fbCIG0033_i_seg.fits
 - fbCIG0033_i.cat
 - fbCIG0033_i.sex
 - fbCIG0033_r_seg.fits
 - fbCIG0033_r.cat
 - fbCIG0033_r.sex
 - sextractor_result.xml
- ▼ folder results
 - plots
- ▼ folder user_input
 - ▼ folder common
 - ▼ folder images
 - fbCIG0011_g.fit
 - fbCIG0011_i.fit
 - fbCIG0011_r.fit
 - fbCIG0033_g.fit
 - fbCIG0033_i.fit
 - fbCIG0033_r.fit
 - ▼ folder self
 - ▼ folder ellipse
 - ▼ folder galfit
 - ▼ folder plots
 - ▼ folder preprocessing
 - init_table.txt
 - ▼ folder sextractor

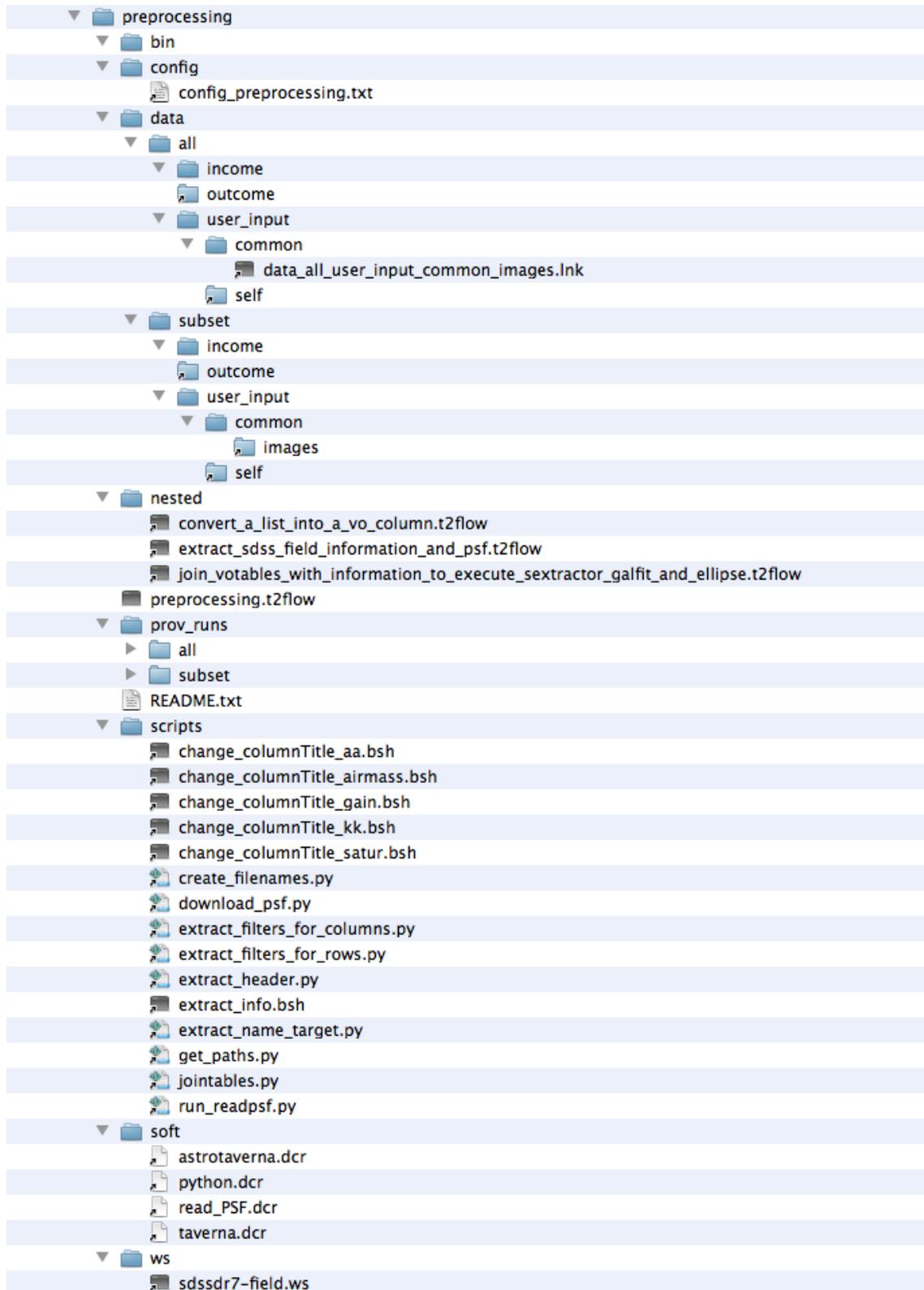
PROCESS



WORKFLOWS

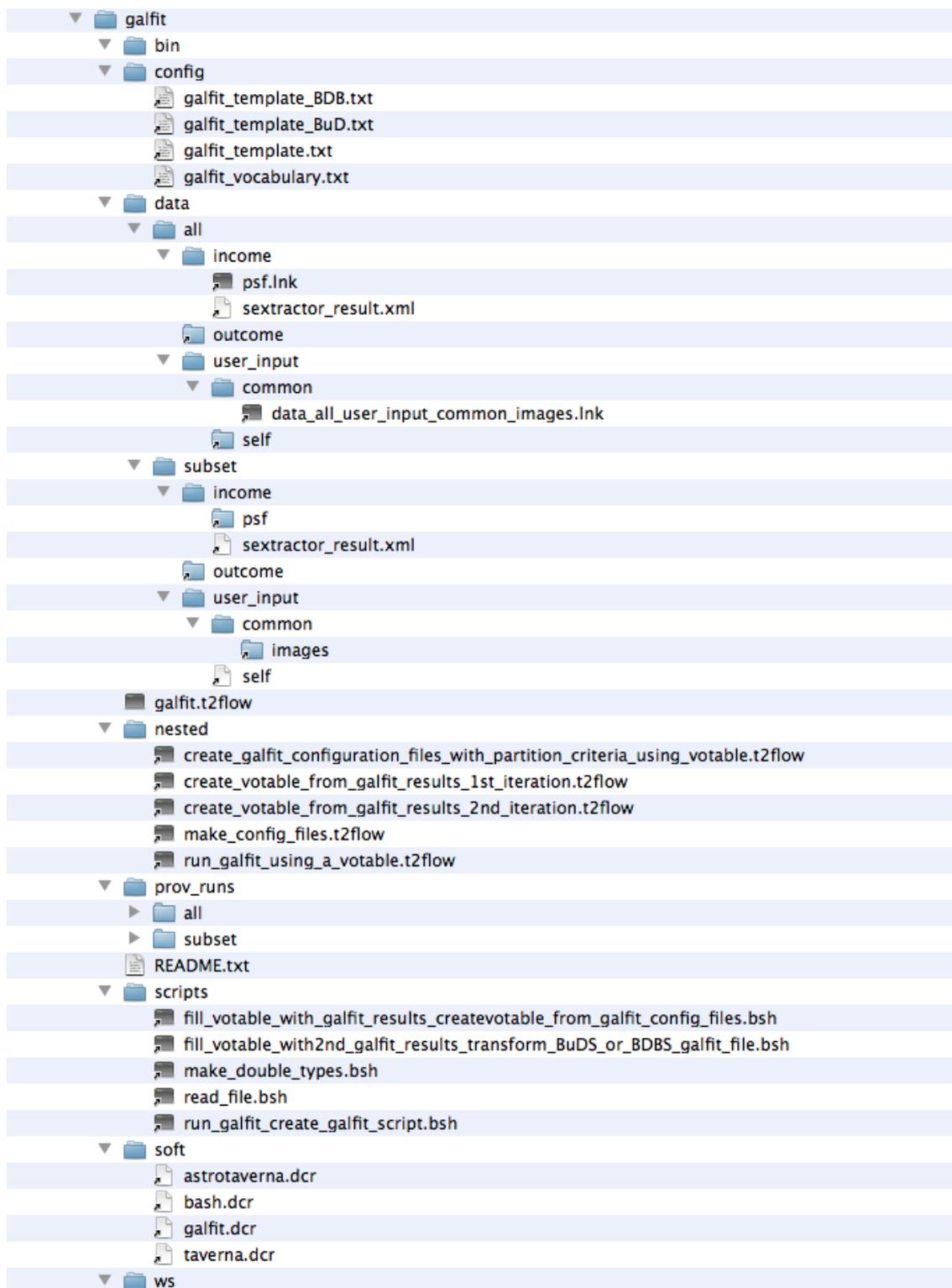
- ▼ workflows
 - ▼ main
 - ▶ ellipse
 - ▶ galfit
 - ▶ plots
 - ▶ preprocessing
 - ▶ sextractor
 - ▼ nested
 - add_columns_to_a_votable_resulting_from_executing_sextractor.t2flow
 - convert_a_list_into_a_vo_column.t2flow
 - create_galfit_configuration_files_with_partition_criteria_using_votable.t2flow
 - create_votable_from_ellipse_results_using_a_votable.t2flow
 - create_votable_from_galfit_results_1st_iteration.t2flow
 - create_votable_from_galfit_results_2nd_iteration.t2flow
 - create_votable_from_sextractor_results.t2flow
 - detect_ellipse_failures_and_get_votable_without_ellipse_failures.t2flow
 - extract_sdss_field_information_and_psf.t2flow
 - join_votables_with_information_to_execute_sextractor_galfit_and_ellipse.t2flow
 - make_config_files.t2flow
 - run_galfit_using_a_votable.t2flow
 - run_scripts_from_a_column_in_a_votable.t2flow
 - run_sextractor_using_votables.t2flow

PREPROCESSING

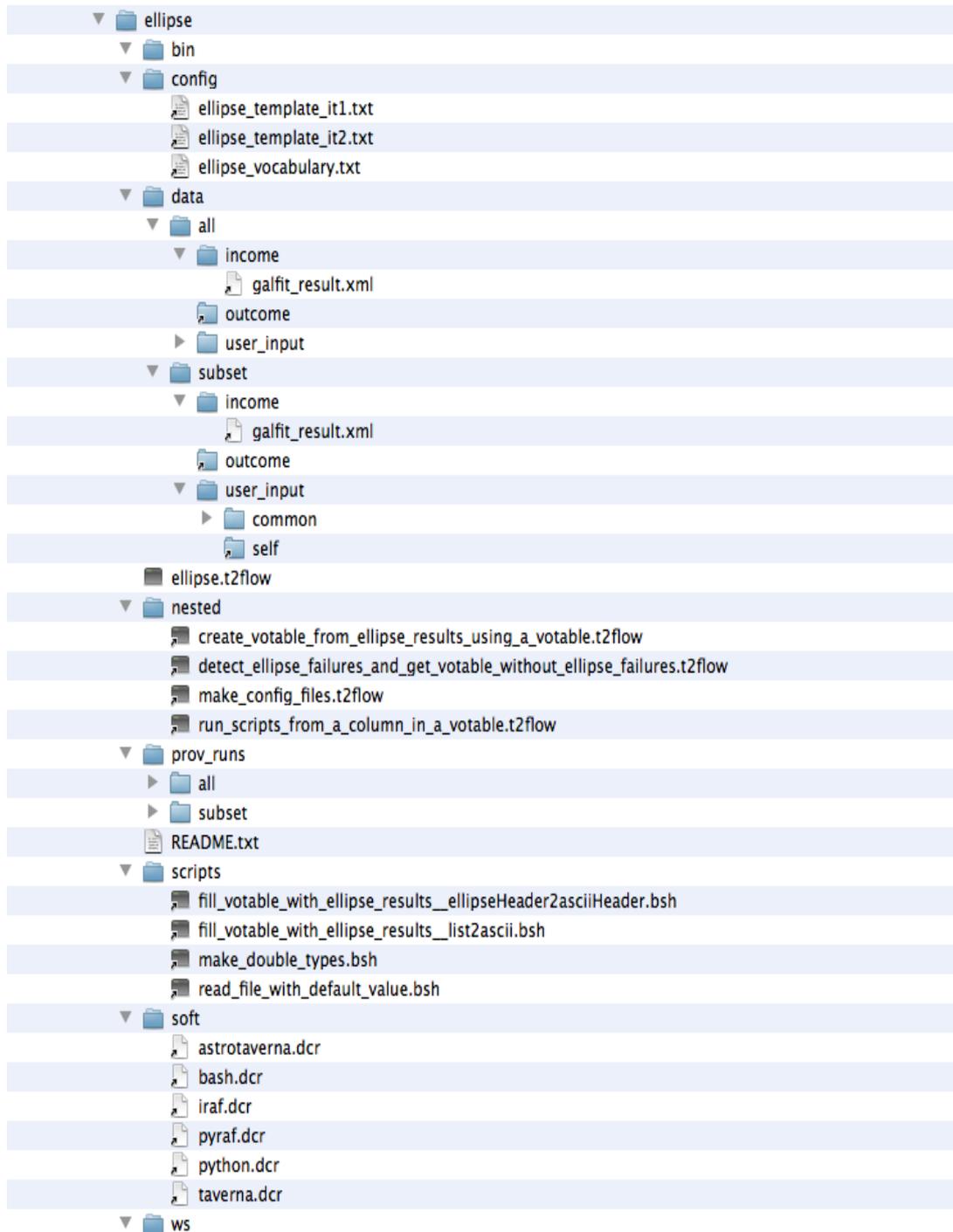


SEXTRACTOR

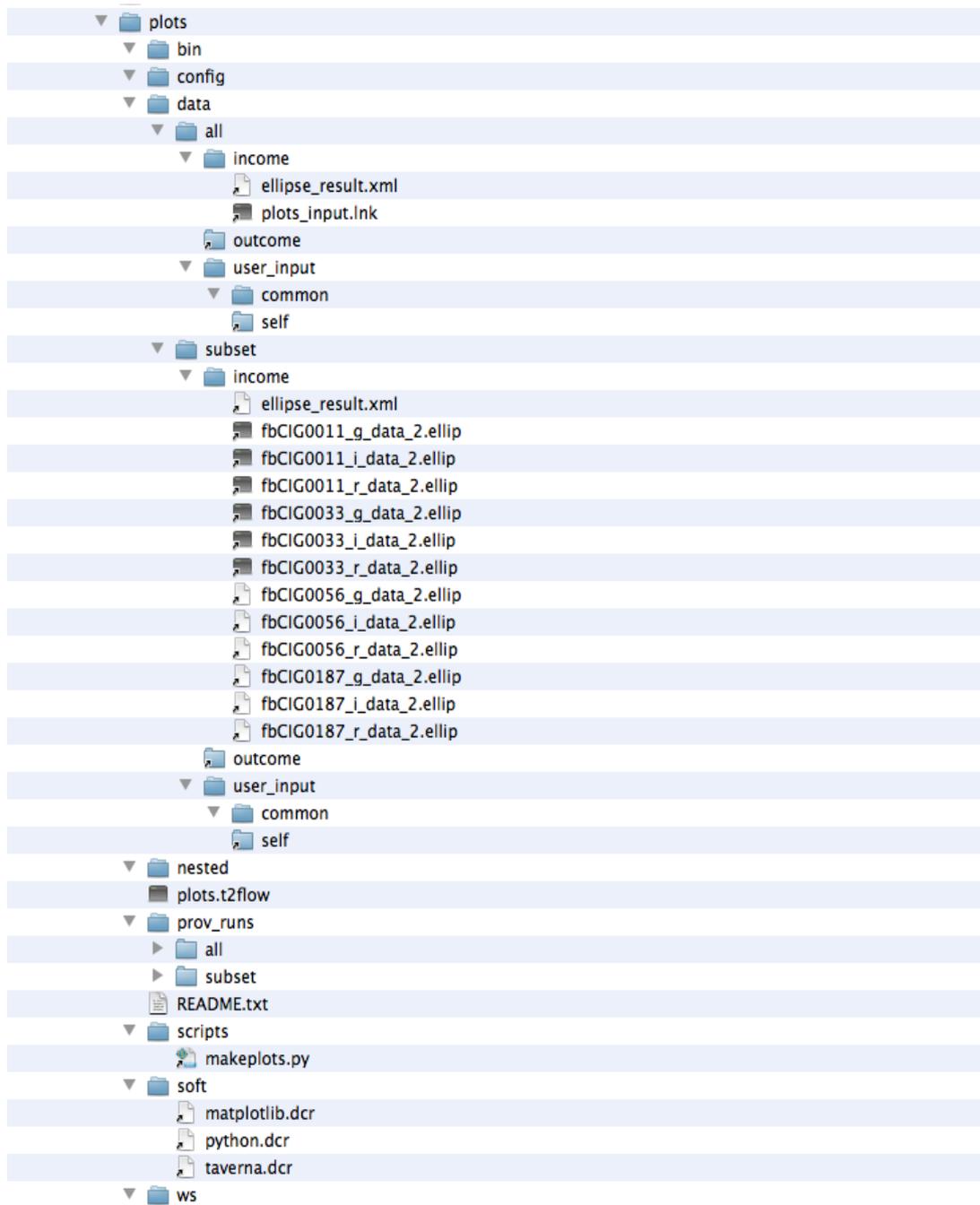
- ▼ sextractor
 - ▼ bin
 - ▼ config
 - 📄 CIG0000.param
 - 📄 sex_template.txt
 - 📄 sex_vocabulary.txt
 - ▼ data
 - ▼ all
 - ▼ income
 - 📄 initVOTable.xml
 - 📄 outcome
 - ▼ user_input
 - ▼ common
 - 📄 data_all_user_input_common_images.lnk
 - 📄 self
 - ▼ subset
 - ▼ income
 - 📄 initVOTable.xml
 - 📄 outcome
 - ▼ user_input
 - ▼ common
 - 📄 images
 - 📄 self
 - ▼ nested
 - 📄 add_columns_to_a_votable_resulting_from_executing_sextractor.t2flow
 - 📄 create_votable_from_sextractor_results.t2flow
 - 📄 make_config_files.t2flow
 - 📄 run_sextractor_using_votables.t2flow
 - ▼ prov_runs
 - ▶ all
 - ▶ subset
 - 📄 README.txt
 - ▼ scripts
 - 📄 read_file.bsh
 - 📄 run_sextractor_create_sextractor_script.bsh
 - 📄 sextractor.t2flow
 - ▼ soft
 - 📄 astrotaverna.dcr
 - 📄 bash.dcr
 - 📄 sextractor.dcr
 - 📄 taverna.dcr
 - ▼ ws

GALFIT

ELLIPSE



PLOTS



Appendix C – Results of Luminosity Profiles RO

Luminosity profiles obtained with ELLIPSE (black points). The solid lines are the model given by GALFIT for each component, and the total model (blue line). In the bottom panels we represented the difference between the observed data and the total model.

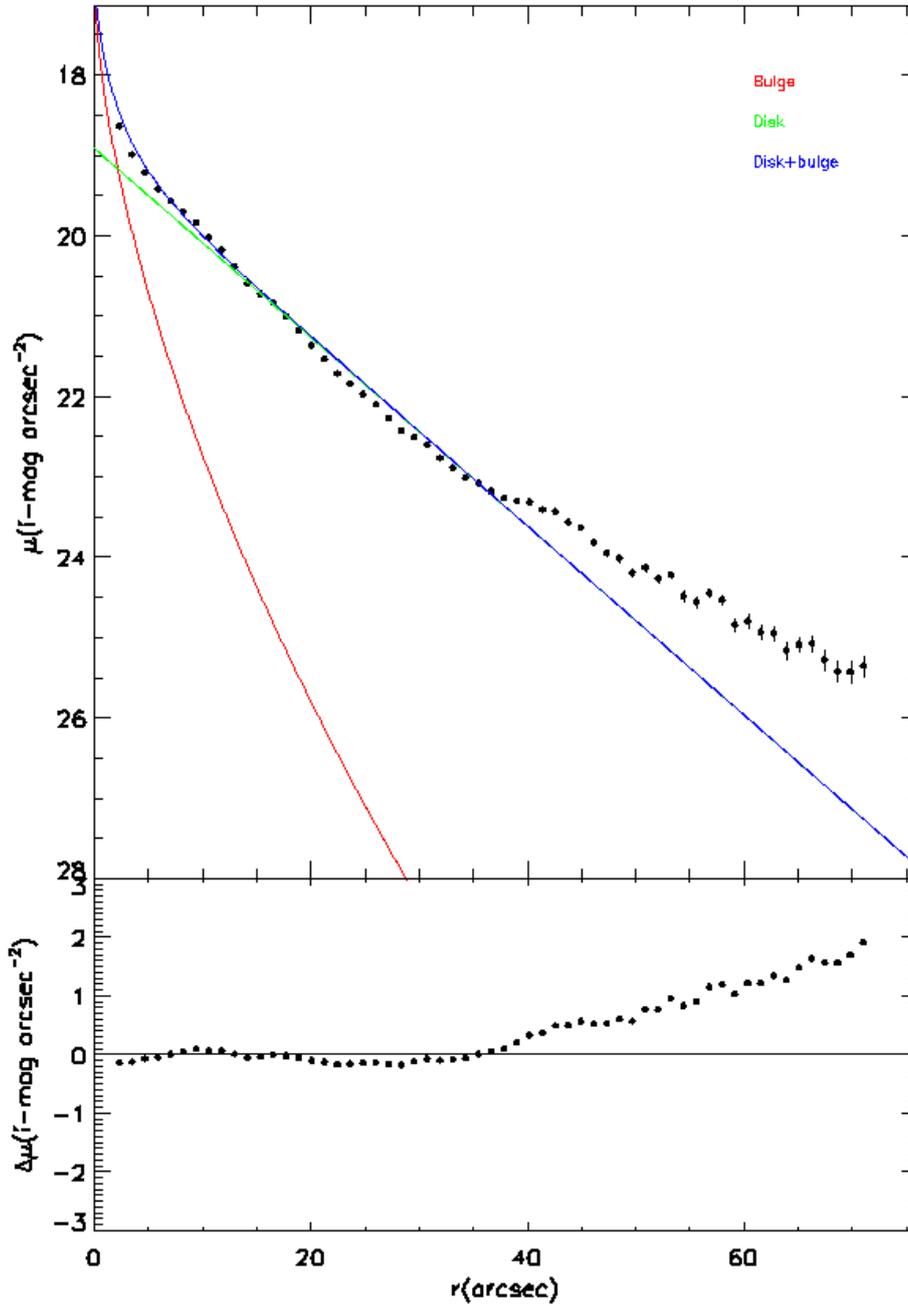


Figure 1: Luminosity profiles for galaxy CIG 33 (anti-truncation)

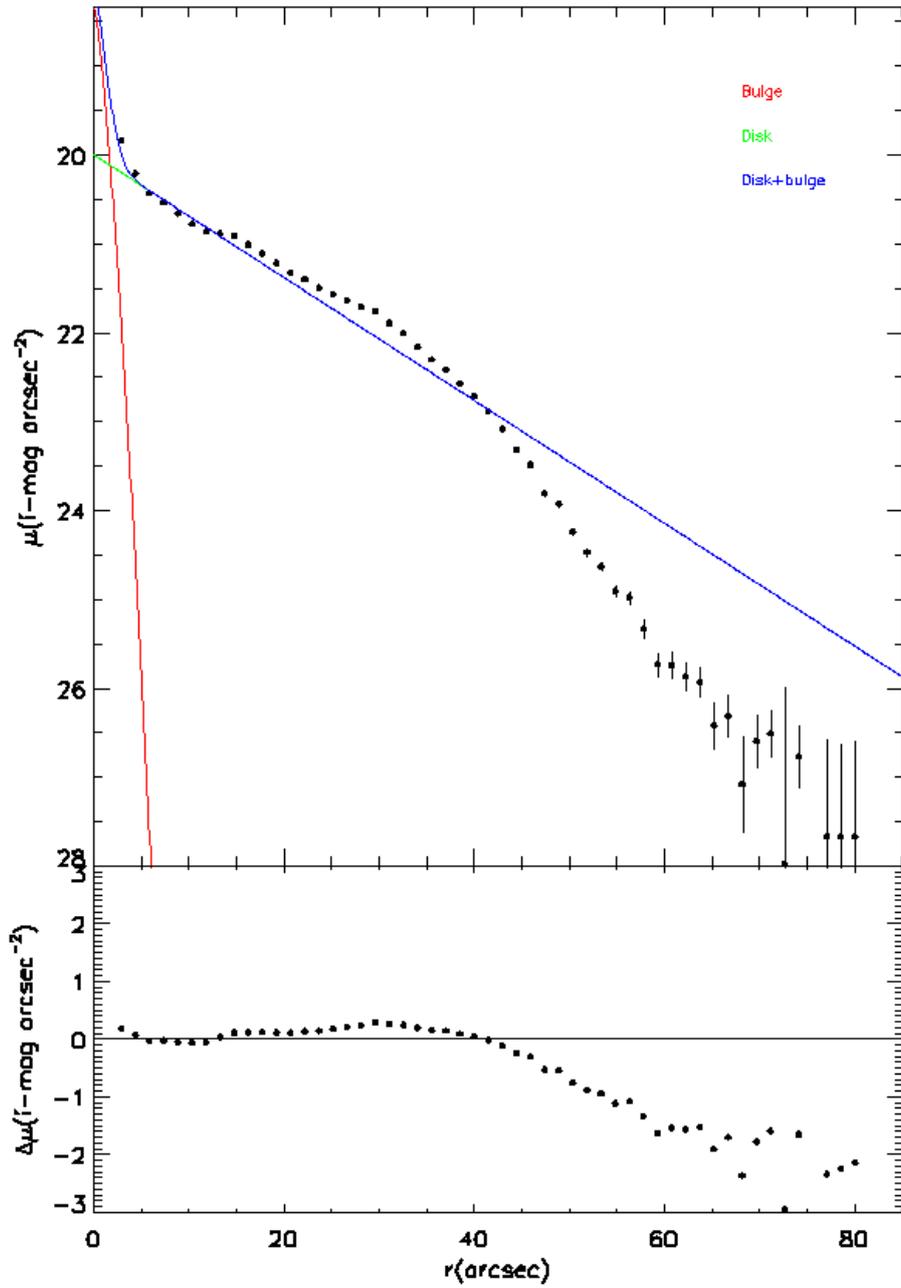


Figure 2: Luminosity profiles for galaxy CIG 281 (truncation)

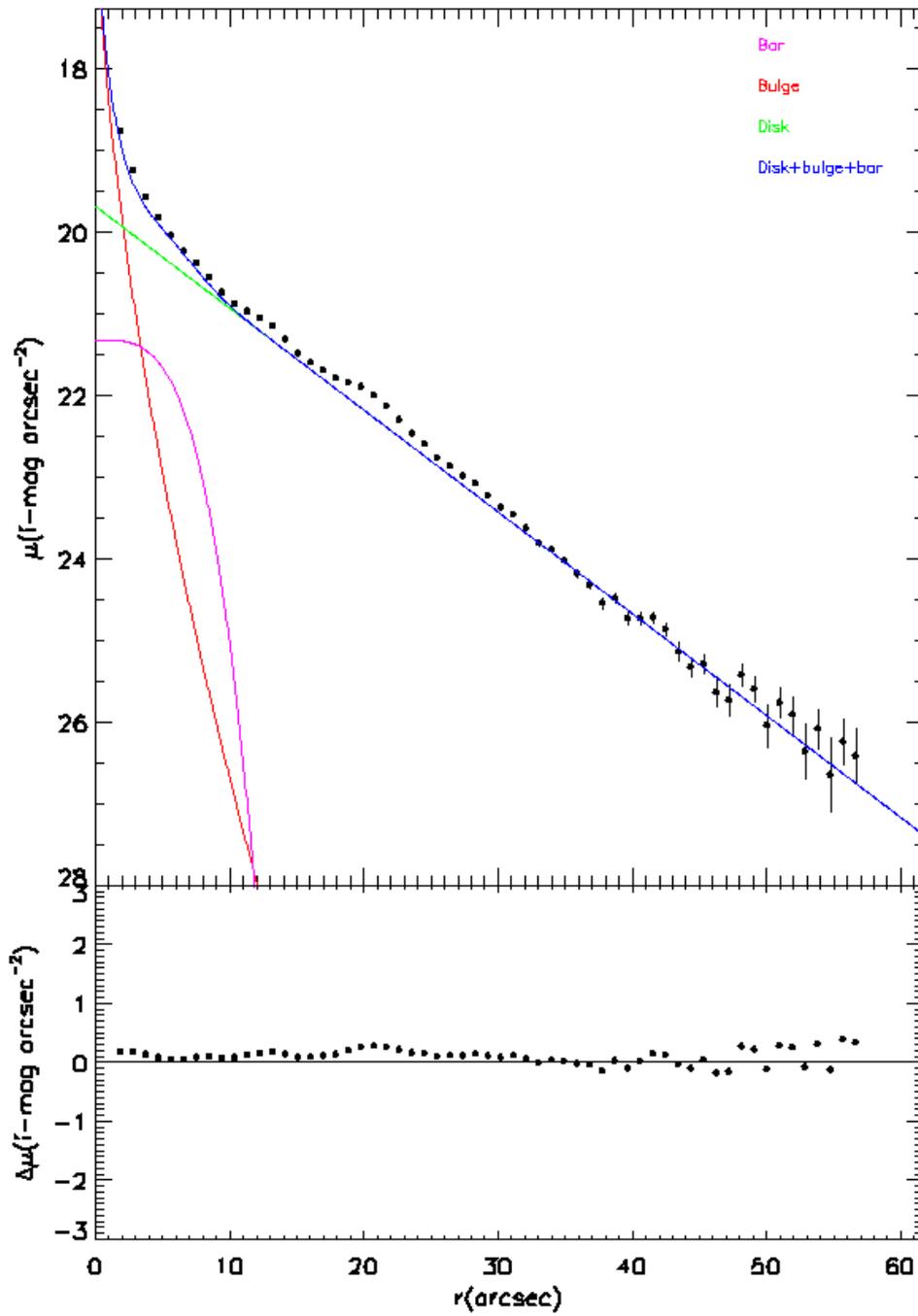


Figure 3: Luminosity profiles for galaxy CIG 520 (normal)