GALFIDL: UPDATED PIPELINE 4 ROUTINES

Heikki Salo 11.08.10

1 INTRO

This manual describes the updated PIPELINE 4 routines, replacing those documented in Section 8 of `galfidl_manual_110110.pdf`. Please have also a look at the original GALFIDL documentation (note that P4 is there called Pipeline 3).

The main changes are:

• All input data is assumed to be copied to the current directory.

Previously the P1,P2,P3 data was supposed to be fetched from the S4G directory-tree in the local disk, assumed to be identical with that in the S4G server (s4g.nrao.edu). However, the S4G directory structure has changed, and is likely to change further in the future, so no attempt will be made to maintain local copies of it (also, there is no rsync possibility). Simplest solution is to copy to current working directory all necessary data, in batches of, say, 10-20 galaxies.

• Interface routines to P3 have been omitted (automatic building of *datalist0*-files), since no P3 data is yet available.

• The current P1 input files are no longer background subtracted, so we have to handle this in the decompositions.

In principle there are two main options (second one is chosen)

1) Include a sky-component to the GALFIT decomposition. However, this would require large sky regions to be included to the iteration, increasing the cpu time consumption. Masking should also be done very carefully, not only on the galaxy region but also on the adjacent sky regions.

2) Estime sky value beforehand and subtract it from the input data fed to GALFIT. Sky value estimates will be eventually done by Pipeline 3 for all galaxies; meanwhile one can do that by other means.

The new PIPELINE 4 routines can be fetched from the directory

`http://cc.oulu.fi/~hsalo/S4G_PIPELINE4/GALFIDL/GALFIDL_110810.dir`

The routines are collected to a tar-file `galfidl_110810.tar`, which replaces the previous `galfidl.tar` release (in `GALFIDL/GALFIDL_100110.dir`. There are also some updates in the display routines etc, so please use these new programs instead of the old ones, even if not using the pipeline 4 specific routines.

2 PIPELINE 4 PROCEDURES: making the *ingal*-templates

• REQUIREMENTS:

The Pipeline 4 -specific procedures assume that the Pipeline 1 and 2 data are stored to the current working directory. This includes

```
GALAXY.phot.1.fits          DATA image
GALAXY.phot.1_wt.fits       WEIGHT image
GALAXY.1.finmask.fits       MASK image
```

`Here GALAXY is the galaxy-identification.`

`The name of the mask files has not yet 'stabilized' - in this manual`
`we use the '1. finmask.fits' suffix, refering to the hand-edited`
`masks made in Oulu from Pipeline 2 '1.lo_mask.fits' files.`

`A PSF-image for each galaxy is also needed: we assume that the composite PSF made by Tom Jarrett is used for each galaxy.`

`NOTE: The current procedures assume that 3.6 micron data is used`
`Using the 4.5 micron band will require small name changes to some of the procedures.`

• MAKING THE INGAL-FILES:

STEP 0: Write the *datalist0*-table.

The basic data for each galaxy to be processed with pipeline 4 is collected to the *datalist0*-file.

This contains:

- Name of the `DATA-image`
- Galaxy center location `xc, yc` (in iraf-coordinates system starting from [1,1])
- Estimated outer disk orientation `PA, ELLIP`. For edge-on galaxies write 'zzzzz' instead of ellipticity; this will indicate that a different fitting function needs to be used by GALFIT
- Estimated galaxy outer radius `A26p5`
- Region used in the decompositions: $\pm$ `AMULT` $\times$ `A26p5 pixels around the galaxy center`.
- Sky background estimate and its uncertainty `SKY, DSKY`
- Name of the `MASK-image`
- Name of the `WEIGHT-image`
- Name of the `PSF-image`

Here is an example (just two data lines retained)

```
IDL> $more example290710.datalist0
version=2
# ------------------------------------------------------------
# 270710.WARM.datafile0_skele --> example290710.datalist0
# made with command make_datalist0,270710.WARM
# laineja@hiisi.oulu.fi    27-Jul-2010 14:25:28.00
# (code version 210610 HS)
# ------------------------------------------------------------
#  IDE   image                    XC(iraf)     YC(iraf)     PA          ELLIP      A26p5      AMULT      SKY         DSKY        mask                    weight                 psf
#...
NGC0918  NGC0918.phot.1.fits      275.092      268.614      157.725   0.414120     220.0      1.3     0.112078   0.00265431   NGC0918.1.finmask.fits   NGC0918.phot.1_wt.fits  PSF-1.composite.fits
NGC0936  NGC0936.phot.1.fits      612.180      569.404      126.611   0.261217     270.0      1.3     0.112393   0.00327166   NGC0936.1.finmask.fits   NGC0936.phot.1_wt.fits  PSF-1.composite.fits
#...
```

NOTE: The first line `'version=2'` is mandatory (compared to the original `'version=1'` datafiles this indicates that new variables `SKY,DSKY` are included). Lines starting with `'#'` are treated as comments.

STEP 1: Final *datalist*-table.

The zero '0' in the end of the previous DATALIST-file indicated that this is not yet the final data compilation we need for the decompositions.

For the decompositions we need the Sigma images (SIGMA), specifying the statistical uncertainty associated with each image pixel, in the same units as the image data. Additionally, the input images should not contain NaN pixel values, since GALFIT does not handle them. Also, the EXPTIME header keyword in the DATA image files should be set to unity, so that GALFIT does not try to normalize the input data values. Finally, we want to subtract the sky background from the data images.

GALFIDL-routine `datalist0_to_datalist.pro` takes care of all that:

```
1) Constructs SIGMA images from the DATA and WEIGHT images:
2) Makes cleaned versions of the DATA image, where NaN pixel values have been removed,
   and EXPTIME is set to unity. Sky background (SKY) is also subtracted.
3) makes new MASK images, where the pixels which had NaN's either in DATA or in WEIGHT
   (and thus also in SIGMA) have been included to the pixels to be masked away.
4) Writes a new datalist-file where file names reflect the above changes
```

To do this for *'example290710.datalist0'*, use the command

```
datalist0_to_datalist,example290710',s4g='./'
```

Note: the keyword `s4g='./'` indicates that the data is read from the current directory, instead of S4G directory-tree.

This creates a new datalist file *'example290710.datalist'*, where the new DATA and MASK image names are inserted, and the WEIGHT images have been replaced by SIGMA images. The other parameters are the same as before.

```
IDL> $more example290710.datalist
version=2
# ============================================================
# Interface between P1/P2 and P4
# datalist-file made from infile= example290710.datalist0
# heikki@ikiturso    29-Jul-2010 13:23:51.00
# ============================================================
# IDE  image  XC(iraf)  YC(iraf)  PA  ELLIP  A26p5  AMULT SKY     mask  sigma    psf
#
NGC0918   NGC0918.phot.1_nonan.fits   275.092    268.614    157.725   0.414120  220.0  1.3   0.112078  0.0026543  NGC0918.1.finmask_nonan.fits   NGC0918.phot.1_sigma.fits   PSF-1.composite.fits
NGC0936   NGC0936.phot.1_nonan.fits   612.180    569.404    126.611   0.261217  270.0  1.3   0.112393  0.0032716  NGC0936.1.finmask_nonan.fits   NGC0936.phot.1_sigma.fits   PSF-1.composite.fits
```

The listed new fits-files are also created, e.g.

```
NGC0918.phot.1_sigma.fits            SIGMA image
NGC0918.phot.1_nonan.fits            new DATA image
NGC0918.1.finmask_nonan.fits         new MASK imag
```

Additionally, the above procedure wrote a new procedure for checking the just-written SIGMA-images. Run this by the command (instructions were also displayed on the screen).

```
.run example290710_make_sigma_driver.pro
```

which makes for each galaxy a plot comparing the calculated sigma with the actual standard deviation in the image (in circular zones). At sky region these should correspond each other (and they do, but only if the fudge-factor about 2-3 is used - a pending topic for P4 QC!).

To get rid of excess plots, delete the IDL windows with the command `wide`.

STEP 2: SEMI-AUTOMATIC MAKING OF *ingal* -FILES

Making template files for GALFIT iterations using the data in *"example290710.datalist"*.

At the moment, the template files created by the procedure `datalist_to_ingal_driver_f.pro` include the following ("ID stands for galaxy id")

```
If the disk is not too inclined (disk ellipticity given in datalist-file)

ID_b.ingal             - 1 component fit initial values (Sersic bulge)
ID_bd.ingal            - 2 component fit initial values (Sersic bulge + exponential disk - fixed orientation)
ID_bdf.ingal           - 2 component fit initial values (Sersic bulge + exponential disk - free orientation)
ID_bdbarnmodel.ingal    - 4 component template (Sersic bulge, expo disk, Ferrers bar, nucleus)

For an edge-on disk (disk ellipticity indicated as 'zzzzz')

ID_b.ingal             - 1 component fit initial values (Sersic bulge)
ID_bz.ingal            - 2 component fit initial values (Sersic bulge + edge-on disk)
ID_bzbarnmodel.ingal    - 4 component template (Sersic bulge, edge-on disk, Ferrers bar, nucleus)
```

The 1 and 2-component templates should usually provide reasonably good starting values (= GALFIT iteration will converge to sensible final models).

The 4-component templates are likely to lead 'to 'crash" if used as they are.
Instead, utilize them as follows. First obtain a good 2-component fit, copy the *outgal*-file to a new name reflecting the used 3-component model, then copy/paste/edit the structure component from *ID_ bdbarmodel.ingal*, then run `galfit_run_new`, perhaps after first inspecting the initial values with `galfit_display_new`.

To make these ingal-files enter the command

```
datalist_to_ingal_driver_f,'example290710'
```

The names of printed templates (without ingal suffix) are printed. Altogether the following files have been created

```
1-component models: sersic
 NGC0918_b.ingal
 NGC0936_b.ingal

2-component models: sersic with exponential disk (fixed orientation)
 NGC0918_bd.ingal
 NGC0936_bd.ingal

2-component models: sersic with exponential disk (free orientation)
 NGC0918_bdf.ingal
 NGC0936_bdf.ingal

And copy/paste/edit templates for 3 and 4 comp models:
 NGC0918_bdbarnmodel.ingal
 NGC0936_bdbarnmodel.ingal
```

The set of 2 example galaxies did not contain any edge-on galaxies, so no '_bz.ingal' files were created.

One can now start making GALFIT decompositions with GALFIDL command `galfit_run_new`, as decribed in the original manual.

The steps 1 + 2 can also be combined: try

```
pipe4,'example290710'
```

STEP 3 MAKING GALFIT ITERATIONS IN 'BATCH' MODE

The `galfit_run_new` calls can be collected to a procedure, in order to make several decompositions.

Here is an example procedure for making the 1- and 2-component decompositions for the above two galaxies. (it also contains the bdf and 4-component templates, but these are commented out).

```
IDL> $more example290710_driver.pro
;Automatically generated driver-procedure
;29-Jul-2010 14:38:14.00
;new=1 -> make decomposition iterations
;new=0 -> display previously done decompositions
;         (if they exist; otherwise make iteration)
;ps=0   --> screen
;ps=-2  --> ps and png-files
new=1
ps=0

galfit_run_new,"NGC0918_b",ps=ps,new=new
galfit_run_new,"NGC0918_bd",ps=ps,new=new
;galfit_run_new,"NGC0918_bdf",ps=ps,new=new
;galfit_run_new,"NGC0918_bdbarnmodel",ps=ps,new=new

galfit_run_new,"NGC0936_b",ps=ps,new=new
galfit_run_new,"NGC0936_bd",ps=ps,new=new
;galfit_run_new,"NGC0936_bdf",ps=ps,new=new
;galfit_run_new,"NGC0936_bdbarnmodel",ps=ps,new=new
end
```

The procedure was created automatically by `datalist_to_ingal_driver_f.pro`, and is run by the command

```
.run example290710_driver.pro
```

By changing the keyword value to `new=0`, the same procedure can be run again, to display the iteration results. Changing `ps=0` to `ps=2` will make ps and png files instead of plotting on the screen.

NOTE: if you run the procedure again WITHOUT changing `new=0`, it will redo the decompositions: use this only if ingal-files have been modified.

This procedure will produce a lot of plots: 6 plots/model *3 models/galaxy* 2 galaxies = 36 plots. So it is handy to run in small batches. Remember the `wide` command for deleting IDL windows. Also, even on the first time you can use `ps=2` to direct output to files.

After running these simple models, proceed with more complicated ones, including bar/nuclear components.

Auxillary small IDL routines which might be useful:

```
outgal_to_ingal                  - make a new ingal from an old outgal-file
s4g_show                         - display profile and residual plots
```

STEP 4 WHAT TO DO WITH ALL THAT OUTPUT?

We still needs to decide exactly how to present the results.

At the present it would be ideal to keep all the input and output files. If this takes too much space, the *minimum* is to store

- the used input files: mask and PSF-file (data and weight images can always be be fetched from the S4G server. Also the nonan and sigma images can be made again with GALFIDL)

- the datalist0 -file

- the `ingal`, `outgal` and `outgal._subcomps.save` for each decomposition model.

All the plots can then be re-created with GALFIDL, without repeating the time consuming iterations.

This will later allow fetching of decomposition parameters and presenting them on the manner that is chosen.

The GALFIDL procedures are under development: for any problems/suggestions connect heikki.salo@oulu.fi.