

PySex

Manual de usuario

Christopher Añorve

1 Introducción

1.1 ¿Que es pysex?

Pysex es una rutina escrita python que permite obtener un catálogo con la fotometría de los objetos en una imagen.

Para obtener una mejor reliabilidad y tamaño de todos los objetos en una imagen, pysex ejecuta Sextractor (Bertin 96) con dos configuraciones diferentes y combina estos dos catálogos en uno final.

En el caso de la rutina pysex3 y pysexbcg Sextractor es ejecutado 3 veces.

1.2 ¿Por que ejecutar dos (o más) veces Sextractor?

No hay una sola configuracion de Sextractor en el que se pueda obtener toda la detección de galaxias para imágenes con grandes cantidades de objetos. Para una determinada configuración sextractor puede detectar bien galaxias pequeños que se encuentren agrupadas, sin embargo si hay una galaxia grande está puede ser dividida incorrectamente en varios objetos pequeños. Por el otro lado, para otra configuración diferente, la buena detección de galaxias grandes puede presentar dificultades en la detección de galaxias pequeñas. Para ilustrar mejor esto vease la siguiente figura (cita haussler).

Además de esto existe otro problema adicional: en las regiones donde hay más densidad objetos, sextractor detecta a las fuentes débiles como objetos de tamaño grande, dando la apariencia de que son objetos con bajo brillo superficial (ver figura). Este es un problema común en las imagenes con cúmulos de galaxias.

Todo esto da como resultado un catálogo con objetos que son divididos en pequeñas partes, objetos débiles con tamaños sobredimensionados y objetos que no son detectados ya sea porque son débiles o están cerca de objetos con brillos grandes.

Una mejor solución es ejecutar Sextractor con una configuración optimizada para objetos grandes y otra para objetos pequeños y en grupos de galaxias. Finalmente el catálogo final es el resultado de la combinación de estos dos catálogos. Esto es lo que hace pysex.

2 Funcionamiento

Pysex une dos catálogos de sextractor. Llamaremos a estos catálogos **hot** y **cold**. La manera en que lo hace es tomar **todos** los objetos del catálogo **hot** y añadir aquellos objetos que están en **cold** pero no en **hot**.

Cuando sextractor se ejecuta para generar el catálogo **hot**, este se encargá de detectar todos los objetos grandes, brillantes y extendidos. Por el otro lado, en la configuración **cold**, sextractor está enfocado en detectar objetos débiles y cercanos entre sí.

Para decidir que objetos del catálogo **cold** serán incluidos en el catálogo final, sextractor incluye los objetos del catálogo cold que no contengan en su área los centros de cualquiera de los objetos que pertenezcan al catálogo hot.

Para identificar área de cada objeto pysex calcula una elipse para cada objeto basado en los siguientes parámetros de Sextractor:

Detalles sobre que configurar los catalogos hot y cold son explicados en 6.

3 El programa

El paquete *pysex* contiene los siguientes archivos:

pysex.py Programa principal

config.txt Archivo de configuración de pysex

default.conv Archivo de sextractor para convolución

default.sex Archivo en el cual se basa pysex para configurar sextractor

default.nnw Archivo neural networks de sextractor

sex.param Archivo de salida de parámetros de sextractor

pysex3.py Pysex que ejecuta sextractor 3 veces

pysexbcg.py Pysex que ejecuta sextractor 3 veces para detectar BCG

makemask.py Crea máscaras a partir del catalogo de pysex

remove.py Programa que remueve mascaras particulares de la imagen creada por makemask.py

LICENSE Licencia del programa

4 Requerimientos

Para ejecutar pysex se necesita tener instalados los siguientes programas: **DS9**, **Sextractor** y **python 3**. En el caso de python, se necesitan las siguientes librerías: *numpy*, *sys*, *os*, *subprocess*, *astropy*, *os*, *scipy*.

Para instalar DS9, y sextractor en *ubuntu* se puede ejecutar la siguiente instrucción:

```
> sudo apt install saods9
> sudo apt install sextractor
```

Para instalar python 3 se recomienda hacerlo a través del paquete anaconda.

5 Manual

Para ejecutar pysex se necesita ejecutar el siguiente comando (linux):

```
> ./pysex.py [ConfigFile] [ImageFile]
```

ImageFile es la imagen en la cual pysex va a trabajar y ConfigFile es el archivo de parámetros que pysex necesita para ejecutar sextractor. Cada uno de los parámetros se explica en la siguiente sección.

5.1 Archivo de parámetros

A continuación se muestra los parámetros

- FirstRun** Permite ejecutar SExtractor para la primera configuración
- SecondRun** Permite ejecutar SExtractor para la segunda configuración
- Scale** Factor de escala para el cual las elipses de cada objeto son agrandadas
- SatDs9** Archivo de regiones para DS9 que indica los objetos saturados
- SatScale** Factor de escala para el cual las regiones saturadas de cada objeto son agrandados
- SatOffset** Cantidad de pixeles que se pueden alargar para las regiones para los objetos saturados
- MakeMask** Bandera que indica si pysex va a realizar una imagen de máscaras (valor 1 para hacerlo)
- OutCatalog** Nombre final de catálogo de salida
- RegDs9** Archivo de regiones para DS9 que muestra los objetos de los catálogo final.
- DEBLEND_NTHRESH1** Número de separadores para umbral. Parámetro para SExtractor para ejecución hot. Ver DEBLEND_NTHRESH del manual de SExtractor.
- DEBLEND_MINCONT1** . Mínimo valor de contraste para separación Parámetro para SExtractor para ejecución hot. Ver DEBLEND_MINCONT del manual de SExtractor.

ANALYSIS_THRESH1 Valor sigma o umbral para realizar análisis. Parámetro para Sextractor para ejecución hot. Ver ANALYSIS_THRESH del manual de Sextractor

DETECT_THRESH1 . Valor sigma o umbral para detectar fuentes. Parámetro para Sextractor para ejecución hot. Ver DETECT_THRESH del manual de Sextractor

DETECT_MINAREA1 . Valor mínimo de área por encima del umbral para detectar fuentes. Parámetro para Sextractor para ejecución hot. Ver DETECT_MINAREA del manual de Sextractor

BACK_SIZE1 . Malla para el fondo. Parámetro para Sextractor para ejecución hot. Ver BACK_SIZE del manual de Sextractor

BACK_FILTERSIZE1 . Tamaño de filtro para el fondo. Parámetro para Sextractor para ejecución hot. Ver BACK_FILTERSIZE del manual de Sextractor

ANALYSIS_THRESH2 Parámetro para Sextractor para ejecución cold. Ver ANALYSIS_THRESH del manual de Sextractor

DETECT_THRESH2 Parámetro para Sextractor para ejecución cold. Ver DETECT_THRESH del manual de Sextractor

DETECT_MINAREA2 Parámetro para Sextractor para ejecución cold. Ver DETECT_MINAREA del manual de Sextractor

DEBLEND_NTHRESH2 Parámetro para Sextractor para ejecución cold. Ver DEBLEND_NTHRESH del manual de Sextractor

DEBLEND_MINCONT2 Parámetro para Sextractor para ejecución cold. Ver DEBLEND_MINCONT del manual de Sextractor

BACK_SIZE2 Parámetro para Sextractor para ejecución cold. Ver BACK_SIZE del manual de Sextractor

BACK_FILTERSIZE2 Parámetro para Sextractor para ejecución cold. Ver BACK_FILTERSIZE del manual de Sextractor

Un ejemplo de como se vería el archivo de configuración para pysex se muestra a continuación:

```
FirstRun 1 # Enable first run (1 = run)
SecondRun 1 # enable second run (1 = run)
Scale 1 # factor scale which ellipses are enlarged
SatDs9 sat.reg
SatScale 3
SatOffset 1
MakeMask 0
OutCatalog hotcold.cat
RegDs9 hotcold.reg
```

```

DEBLEND\_NTHRESH1 64          # Number of deblending sub-thresholds
DEBLEND\_MINCONT1 0.001       # Minimum contrast parameter for deblending
ANALYSIS\_THRESH1 20          # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
DETECT\_THRESH1 20           # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
DETECT\_MINAREA1 10           # minimum number of pixels above threshold
BACK\_SIZE1 100
BACK\_FILTERSIZE1 11
ANALYSIS\_THRESH2 1.5         # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
DETECT\_THRESH2 1.5          # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
DETECT\_MINAREA2 10          # minimum number of pixels above threshold
DEBLEND\_NTHRESH2 16          # Number of deblending sub-thresholds
DEBLEND\_MINCONT2 0.01        # Minimum contrast parameter for deblending
BACK\_SIZE2 10
BACK\_FILTERSIZE2 2

```

5.2 Archivos de salida

Pysex produce principalmente 4 archivos: 1) catálogo final de sextractor, 2) región DS9 de regiones saturadas, 3) archivo de regiones de DS9 de los objetos finales. y 4) una imagen máscara que contiene cada uno los objetos detectados. La descripción de cada archivo se muestra a continuación:

1) El catálogo final de Sextractor indicado por el nombre dado en *OutCatalog* contiene los parámetros de cada una de los objetos detectados en la fuente. Las columnas que contendrá el catálogo final estará indicado por el archivo que está descrito bajo la variable `PARAMETERS_NAME` del archivo *default.sex* (en nuestro ejemplo *sex.param*).

2) El archivo nombrado por la variable *SatDs9* contiene las regiones de los objetos con píxeles saturados. Para ver estas regiones se necesita DS9 y ejecutarlo en la línea de comandos:

```
> ds9 imagen -regions SatDs9
```

3) El archivo de regiones DS9 de los objetos finales contiene a todos los objetos que están en el catálogo final. El nombre de este archivo está dado por la variable *RegDs9*. Para ver estas regiones igualmente se necesita tener DS9 y ejecutarlo en la línea de comandos:

```
> ds9 imagen -regions RegDs9
```

4) El archivo de salida "mask.fits" es una imagen fits que muestra cada uno de los objetos del catálogo final. Cada objeto es mostrado por una elipse. Cada una tiene una cantidad de flujo que corresponde exactamente al número del objeto del catálogo final (exactamente como lo hace Sextractor en la segmentation check image). Así de esta manera es posible identificar el área de cada objeto.

6 Como obtener catálogos óptimos

Para detectar la mayor cantidad de objetos en una imagen, se necesita ajustar **pysex** para que ejecute sextractor con dos configuraciones diferentes. A continuación se muestran recomendaciones de configuración de **pysex** para una buena detección de los objetos.

La idea de la primera ejecución con Sextractor (catálogo **hot**) es detectar los objetos brillantes y grandes. Para esto, se recomienda poner el valor de *DEBLEND_NTHRESH1* con valor pequeño y *DEBLEND_MINCONT1* con valores grandes¹ para evitar que objetos grandes sean divididos en sub-objetos.

Como se desea que el catalogo **hot** contenga solamente a los objetos brillantes y grandes, se recomienda poner los valores de *ANALYSIS_THRESH1*, *DETECT_THRESH1* y *DETECT_MINAREA1* con valores grandes muy por arriba de la señal a ruido.

Los parámetros *BACK_SIZE1* y *BACK_FILTERSIZE1* ayudan a obtener una mejor estimación sobre los límites de cada objeto por lo que se recomienda poner con valores grandes.

Finalmente, un ejemplo de los valores de los parámetros para una ejecución de sextractor para obtener el catálogo hot se muestra a continuación:

```
DEBLEND\_NTHRESH1 32
DEBLEND\_MINCONT1 0.01

ANALYSIS\_THRESH1 20
DETECT\_THRESH1 20
DETECT\_MINAREA1 10

BACK\_SIZE1 100
BACK\_FILTERSIZE1 11
```

Por otro lado, la idea de la segunda ejecución con Sextractor (el catálogo **cold**) es detectar los objetos débiles, pequeños y que probablemente están alrededor de objetos grandes. Para lograr esto, se recomienda poner el valor de *DEBLEND_NTHRESH2* con valor grande y *DEBLEND_MINCONT2* con valor pequeño².

Para garantizar la detección de objetos débiles y pequeños, los valores de *ANALYSIS_THRESH2*, *DETECT_THRESH2* y *DETECT_MINAREA2* deben tener valores pequeños apenas por encima de los valor de la señal a ruido.

Es recomendable tener los parámetros de *BACK_SIZE2* y *BACK_FILTERSIZE2* con valores pequeños.

Finalmente, un ejemplo de valores de los parametros para una ejecución de sextractor para obtener el catálogo **cold** se muestra a continuación:

```
ANALYSIS\_THRESH2 1.5 # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
```

¹chechar manual de Sextractor para una mayor interpretación de estos parámetros

²chechar manual de Sextractor para una mayor interpretación de estos parámetros

```

DETECT\_THRESH2    1.5          # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
DETECT\_MINAREA2   10          # minimum number of pixels above threshold

DEBLEND\_NTHRESH2  64          # Number of deblending sub-thresholds
DEBLEND\_MINCONT2  0.001       # Minimum contrast parameter for deblending

BACK\_SIZE2        10
BACK\_FILTERSIZE2  2

```

Al terminar de ejecutarse pysex, este abrirá ds9 con la imagen y junto con esta se cargará el archivo de las regiones del catálogo final junto con el de los objetos saturados (archivos señalados por los parametros RegDs9 y SatDs9). En este punto el usuario puede verificar si los objetos fueron correctamente detectados.

Por el otro lado, si se desea verificar que los catalogos **hot** y **cold** detectaron las fuentes deseadas uno puede accionar las banderas *FirstRun* y *SecondRun* para que ds9 muestre solamente el catálogo **hot** o **cold**.

Los valores de los parametros explicados en esta sección pueden variar para diferentes tipos de imágenes. Sin embargo, la idea principal es la misma.

7 pysex3 y pysexbcg

Pysex3 y pysexbcg tienen el mismo funcionamiento que pysex, solamente que estos ejecutan Sextractor 3 veces, esto con la finalidad de garantizar una mejor recuperación de las fuentes. pysex3 y pysexbcg se recomienda para aquellas ocasiones en las cuales pysex no pudo detectar satisfactoriamente todos los objetos a pesar de utilizar cualquier configuración

Para el caso de **pysex3**, se adicionan los siguientes parámetros:

ThirdRun

DEBLEND_NTHRESH3

DEBLEND_MINCONT3

ANALYSIS_THRESH3

DETECT_THRESH3

DETECT_MINAREA3

BACK_SIZE3

BACK_FILTERSIZE3

El principio de funcionamiento es el mismo: todos los objetos de la primera ejecución son añadidos, y después se agregan todos aquellos de la segunda ejecución que no están en el primer catalogo. Finalmente, todos aquellos objetos en el tercer catálogo que no están en la combinación de los primeros dos.

Para el caso de **pysexbcg** el objetivo es usarlo en imágenes de cúmulos de galaxias. En estas normalmente existe una galaxia que sobresale de las

demás en cuanto a tamaño y brillo y se llama bright cluster member(BCG). La configuración de la tercera ejecución de sextractor debe estar diseñada para detectar la o las galaxias BCG y añadirlas al catálogo final que resulto de la combinación de las dos primeras ejecuciones de sextractor.

Los parámetros para **pysexbcg** son los mismos que **pysex3** con la excepción que existe un parámetro adicional que es *BCG*. Este le indica al programa cuantas galaxias de la ejecución para detectar BCG son agregadas al catálogo final. Normalmente este valor es 1 o 2, dependiendo de la cantidad de BCG que se encuentren en la imagen.

8 ¿Preguntas? ¿Sugerencias?

Cualquier pregunta, sugerencia o reporte de un bug mandar mail a canorve@gmail.com