

PySex

Manual de usuario

Christopher Añorve

1 Introducción

1.1 ¿Que es pysex?

Pysex es una rutina en python que combina dos catalogos de Sextractor (Bertin 96) para obtener una mejor reliabilidad y tamaño de los objetos en una imagen.

Para hacer esto, pysex ejecuta Sextractor sobre una imagen con dos configuraciones diferentes y obtiene dos catalogos por cada configuración.

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

1.2 ¿Por que ejecutar dos o más veces Sextractor?

Es difícil obtener con Sextractor toda la detección de galaxias en una sola configuración. Para determinadas configuraciones sextractor puede detectar pequeñas galaxias agrupadas, sin embargo si hay una galaxia grande esta puede ser dividida en varios objetos pequeños. Por el otro lado, en otra configuración, la buena detección de galaxias grandes puede presentar dificultades en la detección de galaxias pequeñas (ver figura) (cita haussler).

Además existe otro problema adicional: en las regiones donde hay más densidad objetos, sextractor detecta a las fuentes débiles como objetos de radio grandes con bajo brillo superficial (ver figura). Este es un problema común en las imagenes de cúmulos de galaxias.

Todo esto da como resultado un catalogo con objetos que son partidos en pequeñas piezas, objetos debiles con tamaños sobredimensionados y objetos que no son detectados ya sea porque son debiles o estan en la compañía de objetos con brillos grandes.

Una mejor solución sería ejecutar Sextractor dos veces y tomar las mejores características de cada configuración para combinarlo en un solo catalogo. Esto es lo que hace pysex.

2 Funcionamiento

Pysex junta dos catalogos de dos ejecuciones con diferentes configuraciones de sextractor. Llamaremos a estos catalogos **hot** y **cold**. La manera en que lo hace es tomar **todos** los objetos del primer catalogo y añadir aquellos objetos que están en **cold** pero no estan en **hot**.

En la configuración **hot**, sextractor se encarga de detectar todos los objetos grandes, brillantes y extendidos. Por el otro lado en la configuración **cold**, sextractor esta enfocado en detectar objetos débiles y cercanos entre sí.

Para identificar el tamaño de cada objeto pysex estima una elipse para cada objeto basado en los siguientes parametros de SExtractor:

Para combinar estos dos catálogos, primero pysex incluye a todos los objetos que estan en el catálogo **hot**. Para decidir que objetos del catálogo **cold** seran incluidos en el catalogo final, sextractor incluye los objetos cold que no abarquen en el area de su elipse los centros de los objetos del catalogo hot.

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

3 El programa

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 mascaras 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**. Para python, se necesitan las siguientes librerias: *numpy, sys, os, subprocess, astropy, os, scipy*.

Para instalar DS9, y sextractor en ubuntu se puede realizar con la siguiente instrucción:

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

Para python 3 se recomienda instalar a traves del paquete anaconda.

5 Manual

Para ejecutar pysex se realiza por mediante del 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 parametros se explica en la siguiente sección.

5.1 Archivo de parámetros

A continuación se muestra los parametros

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 las reigones para los objetos saturados
MakeMask	Bandera que indica si pysex va a realizar una imagen de mascaras con el objeto final (valor 1 para hacerlo)
OutCatalog	Nombre final de catalogo de salida
RegDs9	Archivo de regiones para DS9 que muestra los objetos de los catalogo final.
DEBLEND_NTHRESH1	Número de separadores para umbral. Parámetro para SExtractor para ejecucion hot. Ver DEBLEND_NTHRESH del manual de SExtractor.
DEBLEND_MINCONT1	. Mínimo valor de contraste para separació.n 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 minimo de area 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 veria el archivo de configuracion para pysex seria:

```
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 parapymer for deblending
BACK\_SIZE2       10
BACK\_FILTERSIZE2 2

```

5.2 Archivos de salida

Pysex produce 4 archivos: 1) catálogo final de sextractor, 2) region DS9 de regiones saturadas, 3) archivo de regiones de DS9 de los objetos finales. y 4) una imagen mascara que contiene los objetos detectados.

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

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

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

3) El archivo de regiones de DS9 de los objetos finales contiene a todos los objetos detectados por el catálogo final. El nombre de este archivo esta dado por la variable *RegDs9*. Para ver estas regiones es necesario 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 el tamaño de los objetos del catalogo final. Cada objeto es indicado por una elipse. Cada elipse tiene una cantidad de flujo que corresponde exactamente al número del objeto del catalogo final (exactamente como lo hace Sextractor en la segmentation check image). Así de esta manera es posible identificar el area de cada objeto.

6 Como obtener buenos catalogos

El objetivo principal de **pysex** es detectar la mayor cantidad de objetos en una imagen y además estimar el tamaño real de cada objeto.

La idea del primera ejecución con Sextractor (el 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 grande¹ 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 estimacion sobre los limites de cada objeto por lo que se recomienda poner con valores grandes.

Finalmente, un ejemplo de valores de los parametros 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 del 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
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 parapymeter for deblending
```

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

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

```
BACK\_SIZE2          10
BACK\_FILTERSIZE2 2
```

Al terminar de ejecutarse pysex, este desplegara a ds9 con la imagen y junto con esta se cargara el archivo de las regiones del catalogo final junto con el de los objetos saturados (archivos señalados por los parametros RegDs9 y SatDs9). Aquí el usuario puede verificar si los objetos fueron correctamente detectados. Si se desea verificar si los catalogos **hot** y **cold** detectaron las fuentes deseadas uno puede accionar las banderas *FirstRun* y *SecondRun* y ejecutar pysex para al final muestre a ds9 ya sea solamente con el catálogo **hot** o **cold**.

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

7 pysex3 y pysexbcg

Pysex3 y pysexbcg basicamente usan el mismo principio que pysex, solamente que estos ejecutan Sextractor 3 veces.

Esto para garantizar una mejor recuperacion de las fuentes para aquellas ejecuciones de pysex que no pudieron detectar satisfactoriamente todos los objetos haciendo cualquier posible combinacion de parametros.

En el caso de **pysex3**, se adicionan los siguientes parametros:

ThirdRun

DEBLEND_NTHRESH3

DEBLEND_MINCONT3

ANALYSIS_THRESH3

DETECT_THRESH3

DETECT_MINAREA3

BACK_SIZE3

BACK_FILTERSIZE3

El principio es el mismo: todos los objetos de la primera ejecución son añadidos, y despues todos aquellos de la segunda ejecución que no estan en el primer catalogo. Finalmente, todos aquellos en el tercer catalogo que no estan en la combinación de los primeros dos.

Para el caso de **pysexbcg** el objetivo es usarlo en imagenes de cumulos de galaxias. Debido a que en estas normalmente existe una galaxia a la que llamamos bright cluster member(BCG), la configuracion de la tercera ejecucion de sextractor debe estar diseñada para detectar la o las galaxias BCG y añadirlas al catalogo final que resulto de la combinacion de las dos primeras ejecuciones de sextractor.

Los parametros para **pysexbcg** son los mismos que **pysex3**. Solamente existe un parametro adicional que es *BCG*. Este le indica al programa cuantas

galaxias de la ejecucion para detectar BCG son agregadas al catalogo final. Normalmente este valor es 1 o 2, dependiendo de la cantidad de BCG que se contengan en la imagen.

8 ¿Preguntas? ¿Sugerencias?

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