

final_project

March 5, 2024

```
[1]: from google.colab import files
      uploaded = files.upload()
```

<IPython.core.display.HTML object>

Saving image.fits to image.fits

#setup

```
[2]: import numpy as np
```

```
[3]: pip install sep
```

Collecting sep

Downloading

sep-1.2.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (1.2 MB)
1.2/1.2 MB

9.3 MB/s eta 0:00:00

Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from sep) (1.25.2)

Installing collected packages: sep

Successfully installed sep-1.2.1

```
[4]: import sep
```

#extra setup

```
[5]: from astropy.io import fits
      import matplotlib.pyplot as plt
      from matplotlib import rcParams
```

%matplotlib inline

```
rcParams['figure.figsize'] = [10., 8.]
```

#read image into 2d numpy array

```
[6]: hdul = fits.open("image.fits")
      data = hdul[0].data
```

WARNING: The following header keyword is invalid or follows an unrecognized non-standard convention:

ESO-LOG 00:00:00> DATE = '1992-10-26' / Mon Oct 26, 1992

[astropy.io.fits.card]

WARNING:astropy:The following header keyword is invalid or follows an unrecognized non-standard convention:

ESO-LOG 00:00:00> DATE = '1992-10-26' / Mon Oct 26, 1992

WARNING: The following header keyword is invalid or follows an unrecognized non-standard convention:

ESO-LOG 03:04:08>-START EXPO EMMI RED / Start exp. on EMMI Red CC

[astropy.io.fits.card]

WARNING:astropy:The following header keyword is invalid or follows an unrecognized non-standard convention:

ESO-LOG 03:04:08>-START EXPO EMMI RED / Start exp. on EMMI Red CC

WARNING: The following header keyword is invalid or follows an unrecognized non-standard convention:

ESO-LOG 03:04:09> EXPO EMMI RED NO = 24887 / Exp. num. on EMMI Red CCD

[astropy.io.fits.card]

WARNING:astropy:The following header keyword is invalid or follows an unrecognized non-standard convention:

ESO-LOG 03:04:09> EXPO EMMI RED NO = 24887 / Exp. num. on EMMI Red CCD

WARNING: The following header keyword is invalid or follows an unrecognized non-standard convention:

ESO-LOG 03:10:52>-STOP EXPO EMMI RED / Stop exp. on EMMI Red CCD

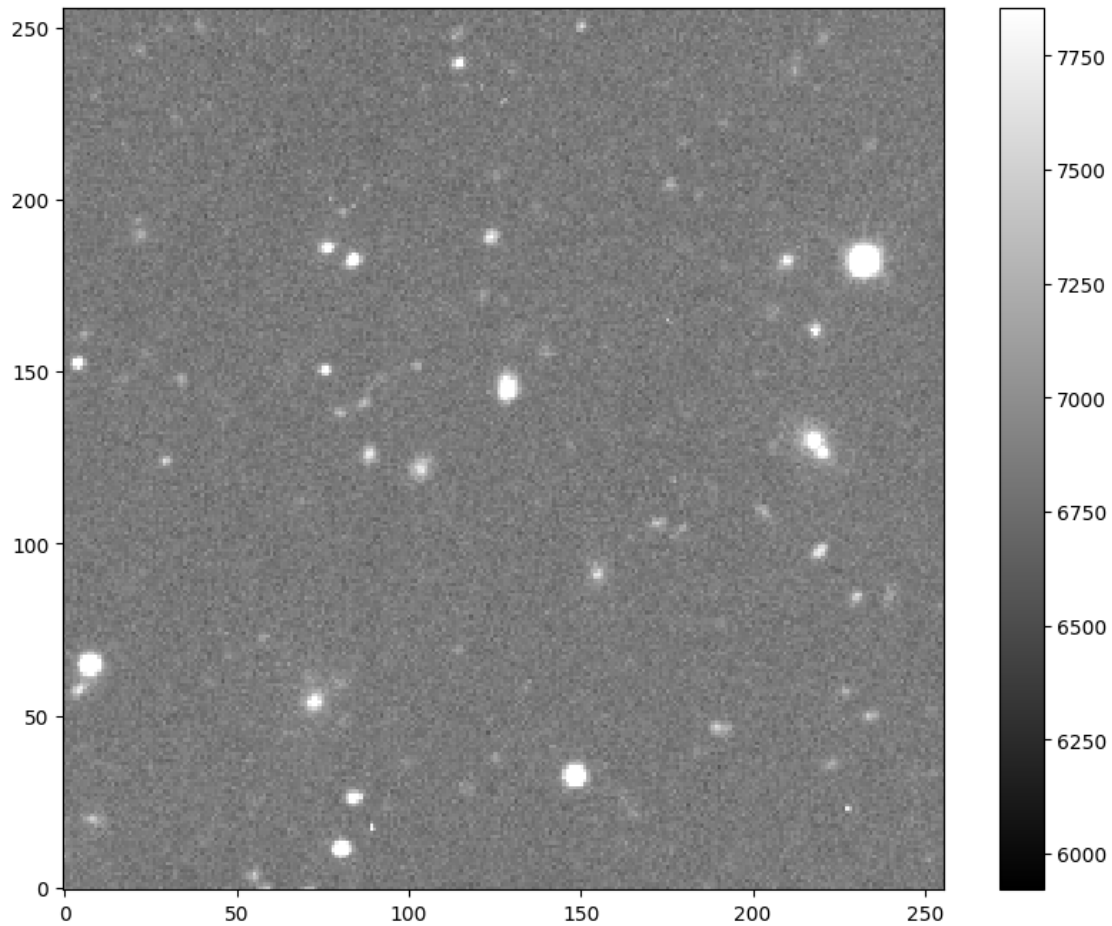
[astropy.io.fits.card]

WARNING:astropy:The following header keyword is invalid or follows an unrecognized non-standard convention:

ESO-LOG 03:10:52>-STOP EXPO EMMI RED / Stop exp. on EMMI Red CCD

#show the image

```
[29]: m, s = np.mean(data), np.std(data)
plt.imshow(data, interpolation='nearest', cmap='gray', vmin=m-s, vmax=m+s,
           origin='lower')
plt.colorbar()
plt.savefig('2dimage.png')
plt.show()
```



```
[ ]:
```

```
#measure background
```

```
[8]: mask = None
```

```
[9]: bkg = sep.Background(data)
```

```
[10]: bkg = sep.Background(data, mask=mask, bw=64, bh=64, fw=3, fh=3)
```

```
get global mean and noise of image background
```

```
[11]: print(bkg.globalback)
      print(bkg.globalrms)
```

```
6852.04931640625
```

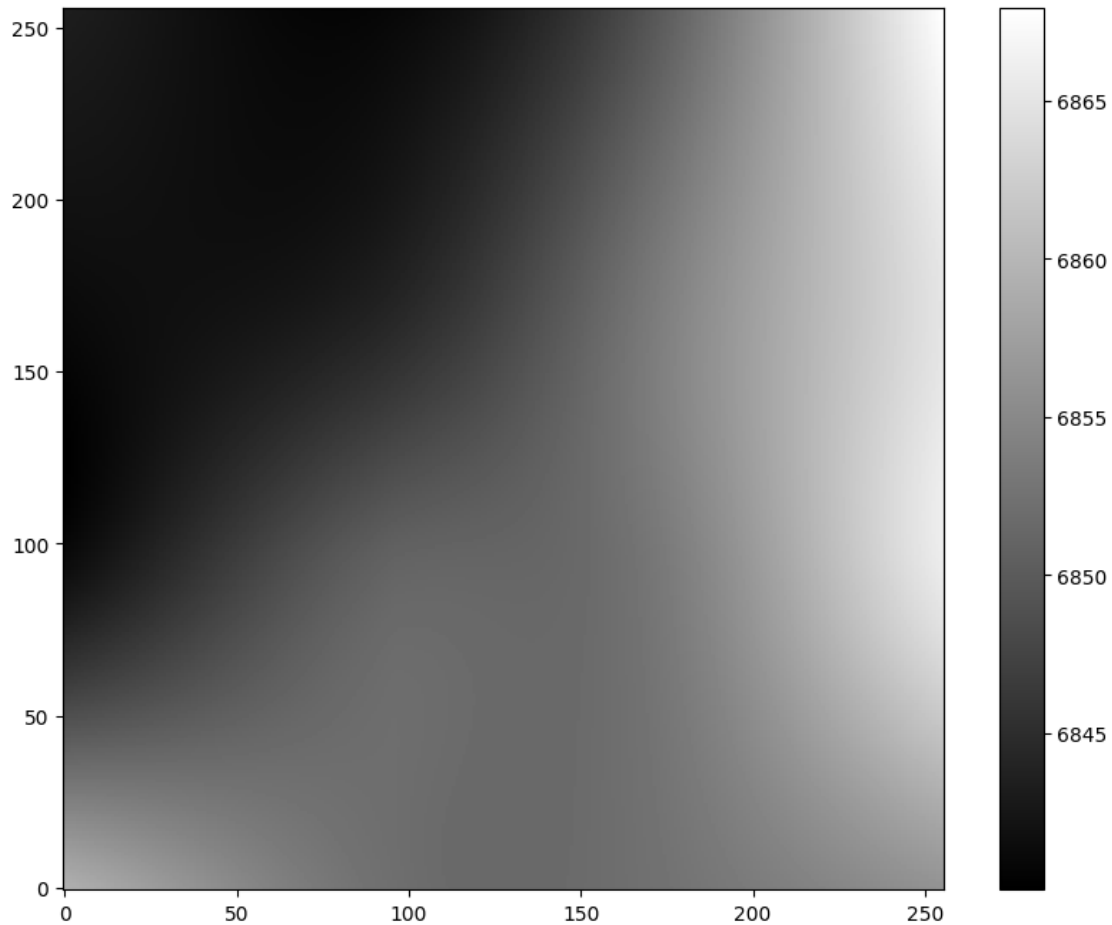
```
65.46174621582031
```

```
evaluate background
```

```
[12]: bkg_image = bkg.back()
```

show background

```
[37]: plt.imshow(bkg_image, interpolation='nearest', cmap='gray', origin='lower')  
plt.colorbar();  
plt.savefig('bkg_image.png')
```

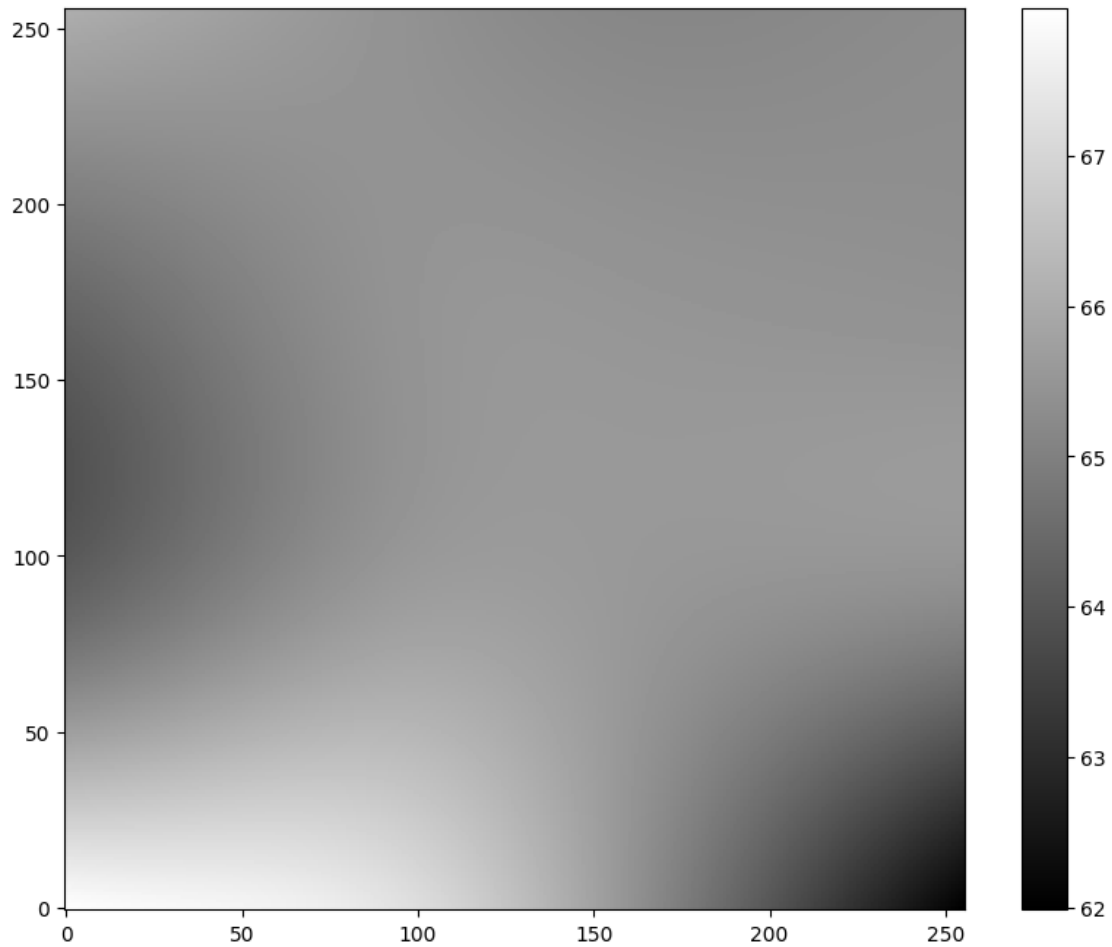


evaluate background

```
[14]: bkg_rms = bkg.rms()
```

show background noise

```
[38]: plt.imshow(bkg_rms, interpolation='nearest', cmap='gray', origin='lower')  
plt.colorbar();  
plt.savefig('bkgnnoise_image.png')
```



subtract background

```
[16]: data_sub = data - bkg
```

object detection

```
[17]: objects = sep.extract(data_sub, 1.5, err=bkg.globalrms)
```

objects detected

```
[18]: len(objects)
```

```
[18]: 69
```

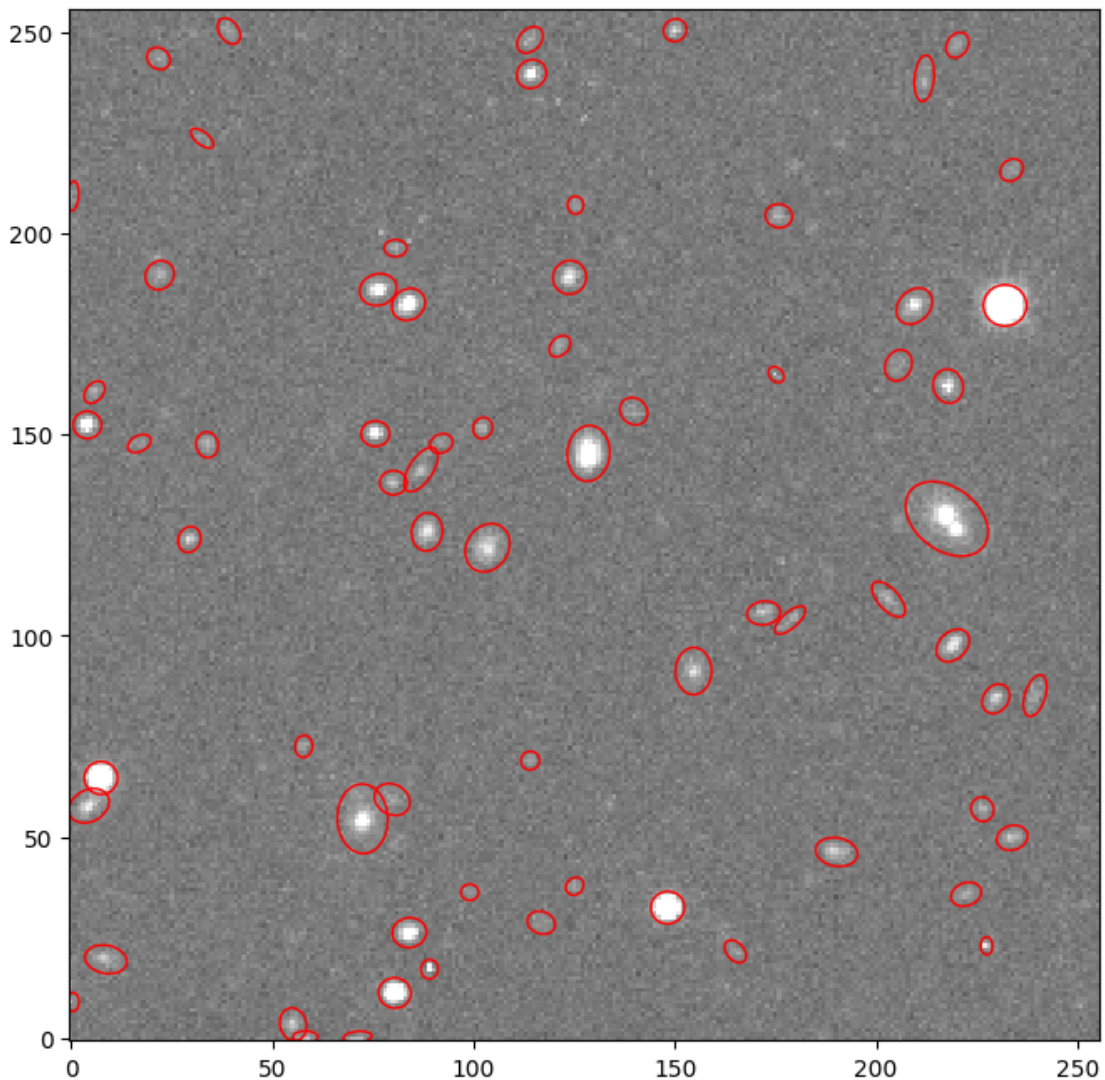
plot background subtracted image

```
[19]: from matplotlib.patches import Ellipse
```

```
[41]: fig, ax = plt.subplots()
m, s = np.mean(data_sub), np.std(data_sub)
im = ax.imshow(data_sub, interpolation='nearest', cmap='gray',
               vmin=m-s, vmax=m+s, origin='lower')

# plot an ellipse for each object
for i in range(len(objects)):
    e = Ellipse(xy=(objects['x'][i], objects['y'][i]),
                width=6*objects['a'][i],
                height=6*objects['b'][i],
                angle=objects['theta'][i] * 180. / np.pi)
    e.set_facecolor('none')
    e.set_edgecolor('red')
    ax.add_artist(e)

plt.savefig('bkgsubtracted_image.png')
```



available fields

```
[21]: objects.dtype.names
```

```
[21]: ('thresh',  
      'npix',  
      'tnpix',  
      'xmin',  
      'xmax',  
      'ymin',  
      'ymax',  
      'x',  
      'y',  
      'x2',  
      'y2',  
      'xy',  
      'errx2',  
      'erry2',  
      'errxy',  
      'a',  
      'b',  
      'theta',  
      'cxx',  
      'cyy',  
      'cxy',  
      'cflux',  
      'flux',  
      'cpeak',  
      'peak',  
      'xcpeak',  
      'ycpeak',  
      'xpeak',  
      'ypeak',  
      'flag')
```

aperture photometry

```
[22]: flux, fluxerr, flag = sep.sum_circle(data_sub, objects['x'], objects['y'],  
                                           3.0, err=bkg.globalrms, gain=1.0)
```

```
[23]: for i in range(10):  
      print("object {:d}: flux = {:.f} +/- {:.f}".format(i, flux[i], fluxerr[i]))
```

```
object 0: flux = 2249.159297 +/- 291.027802  
object 1: flux = 3092.220430 +/- 291.592204  
object 2: flux = 5949.868379 +/- 356.562003
```

```
object 3: flux = 1851.426582 +/- 295.028816
object 4: flux = 72736.386914 +/- 440.172206
object 5: flux = 3860.756152 +/- 352.163162
object 6: flux = 6418.913789 +/- 357.458973
object 7: flux = 2210.707656 +/- 350.791223
object 8: flux = 2741.607227 +/- 352.277746
object 9: flux = 20916.875566 +/- 376.966138
```

```
[ ]:
```