Notebook - March 18 - Part 2

March 21, 2016

```
In [1]: %%bash
        pip install aplpy
        pip install https://github.com/ericmandel/pyds9/archive/master.zip

Requirement already satisfied (use --upgrade to upgrade): aplpy in /Users/adam/anaconda/envs/esopython20
Requirement already satisfied (use --upgrade to upgrade): astropy in /Users/adam/anaconda/envs/esopython
Requirement already satisfied (use --upgrade to upgrade): numpy>=1.6.0 in /Users/adam/anaconda/envs/esop
Collecting https://github.com/ericmandel/pyds9/archive/master.zip
  Downloading https://github.com/ericmandel/pyds9/archive/master.zip (1.0MB)
  Requirement already satisfied (use --upgrade to upgrade): pyds9==1.8.1 from https://github.com/ericman
Requirement already satisfied (use --upgrade to upgrade): six in /Users/adam/anaconda/envs/esopython2016
```

```
In [2]: %%bash
        curl -O https://astropy.stsci.edu/data/galactic_center/gc_bolocam_gps.fits
        curl -O https://astropy.stsci.edu/data/galactic_center/gc_2mass_k.fits
```

| % Total		% Received	% Xferd	Average Speed		Time	Time	Time	Current
				Dload	Upload	Total	Spent	Left	Speed
100 1605k	100 1605k	0	0	318k	0	0:00:05	0:00:05	--:--:--	525k
% Total		% Received	% Xferd	Average Speed		Time	Time	Time	Current
				Dload	Upload	Total	Spent	Left	Speed
100 1020k	100 1020k	0	0	385k	0	0:00:02	0:00:02	--:--:--	385k

```
In [3]: %matplotlib inline
        import pylab as pl
```

```
In [4]: from astropy.io import fits
```

```
In [5]: # load the data (no headers, first extension by default)
        # if there are extensions, fits.getdata('file.fits', ext=extension_number)
        # if you want many extension, use fits.open('file.fits'), then access each independently
        stellardata = fits.getdata('gc_2mass_k.fits')
```

```
In [21]: # show the image: vmax sets the brightest displayed pixel
         # cmap can be any of the valid matplotlib colormaps (pl.cm....)
         pl.imshow(stellardata, cmap='viridis', vmax=1000)
```
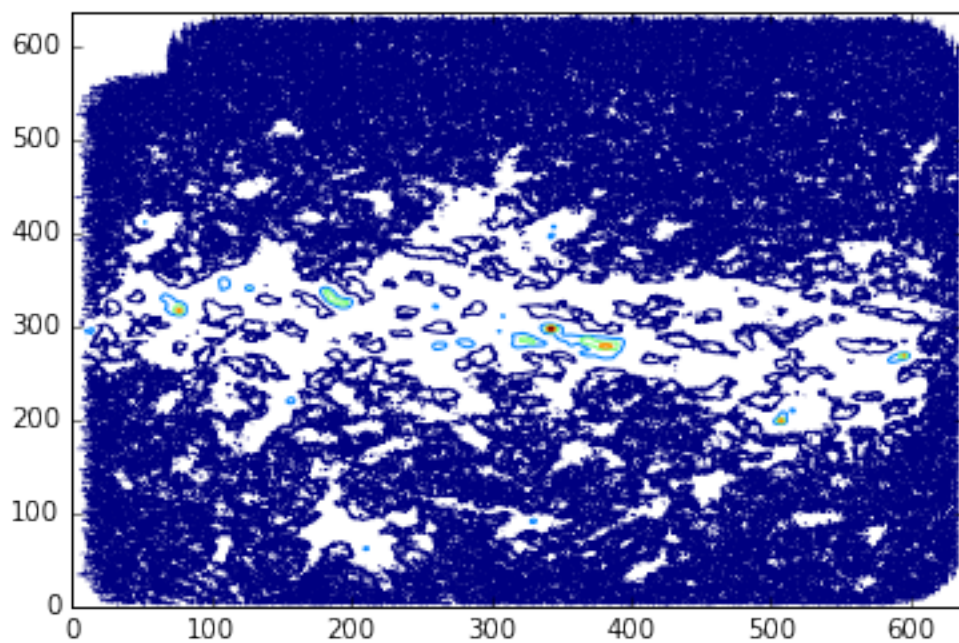
```
Out[21]: <matplotlib.image.AxesImage at 0x118bc4438>
```
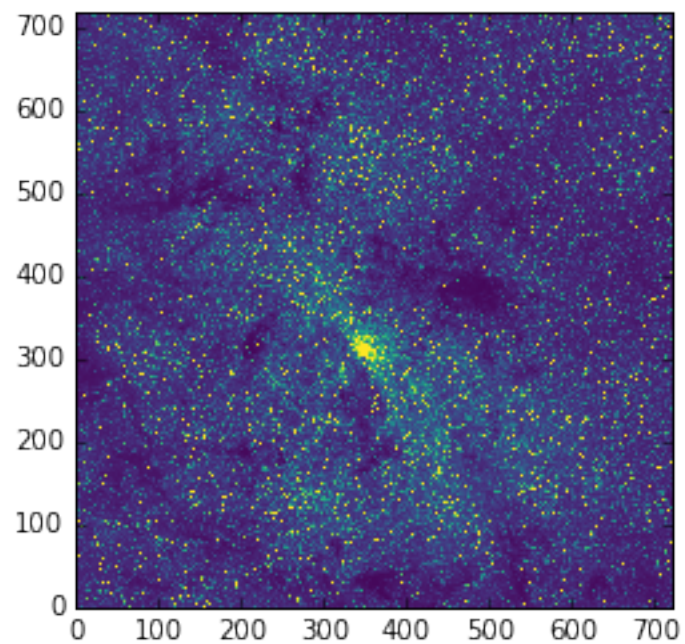
```
In [8]: dustdata = fits.getdata('gc_bolocam_gps.fits')

In [9]: pl.contour(dustdata)

Out[9]: <matplotlib.contour.QuadContourSet at 0x10dc240f0>
```

```
In [13]: dustdata.shape, dustdata.flatten().shape

Out[13]: ((638, 640), (408320,))

In [14]: np.any(np.isnan(dustdata))

Out[14]: True

In [33]: # subset of the data that is not nan
         # implicitly flattens
         non_nan_dustdata = dustdata[~np.isnan(dustdata)]
         non_nan_dustdata = dustdata[np.isfinite(dustdata)]
         non_nan_dustdata = np.compress(np.isfinite(dustdata.flatten()), dustdata.flatten())
         len(non_nan_dustdata)

Out[33]: 387921

In [17]: pl.hist(dustdata[~np.isnan(dustdata)], bins=np.linspace(0,2,50))

Out[17]: (array([  8.99060000e+04,   4.93190000e+04,   2.31990000e+04,
                  1.36530000e+04,   9.18600000e+03,   7.06700000e+03,
                  5.64800000e+03,   4.59500000e+03,   3.70300000e+03,
                  3.22000000e+03,   2.68900000e+03,   2.36500000e+03,
                  2.08000000e+03,   1.80300000e+03,   1.51600000e+03,
                  1.36000000e+03,   1.19100000e+03,   1.01000000e+03,
                  8.55000000e+02,   7.91000000e+02,   7.35000000e+02,
                  5.62000000e+02,   5.14000000e+02,   4.70000000e+02,
                  4.21000000e+02,   3.82000000e+02,   3.33000000e+02,
                  2.89000000e+02,   3.03000000e+02,   2.54000000e+02,
                  2.61000000e+02,   2.24000000e+02,   2.39000000e+02,
                  1.72000000e+02,   2.20000000e+02,   1.57000000e+02,
                  1.70000000e+02,   1.63000000e+02,   1.47000000e+02,
                  1.54000000e+02,   1.26000000e+02,   1.32000000e+02,
                  1.24000000e+02,   1.12000000e+02,   9.00000000e+01,
                  9.30000000e+01,   7.90000000e+01,   8.70000000e+01,
                  7.40000000e+01]),
          array([ 0.        ,  0.04081633,  0.08163265,  0.12244898,  0.16326531,
                  0.20408163,  0.24489796,  0.28571429,  0.32653061,  0.36734694,
                  0.40816327,  0.44897959,  0.48979592,  0.53061224,  0.57142857,
                  0.6122449 ,  0.65306122,  0.69387755,  0.73469388,  0.7755102 ,
                  0.81632653,  0.85714286,  0.89795918,  0.93877551,  0.97959184,
                  1.02040816,  1.06122449,  1.10204082,  1.14285714,  1.18367347,
                  1.2244898 ,  1.26530612,  1.30612245,  1.34693878,  1.3877551 ,
                  1.42857143,  1.46938776,  1.51020408,  1.55102041,  1.59183673,
                  1.63265306,  1.67346939,  1.71428571,  1.75510204,  1.79591837,
                  1.83673469,  1.87755102,  1.91836735,  1.95918367,  2.        ]),
          <a list of 49 Patch objects>)
```
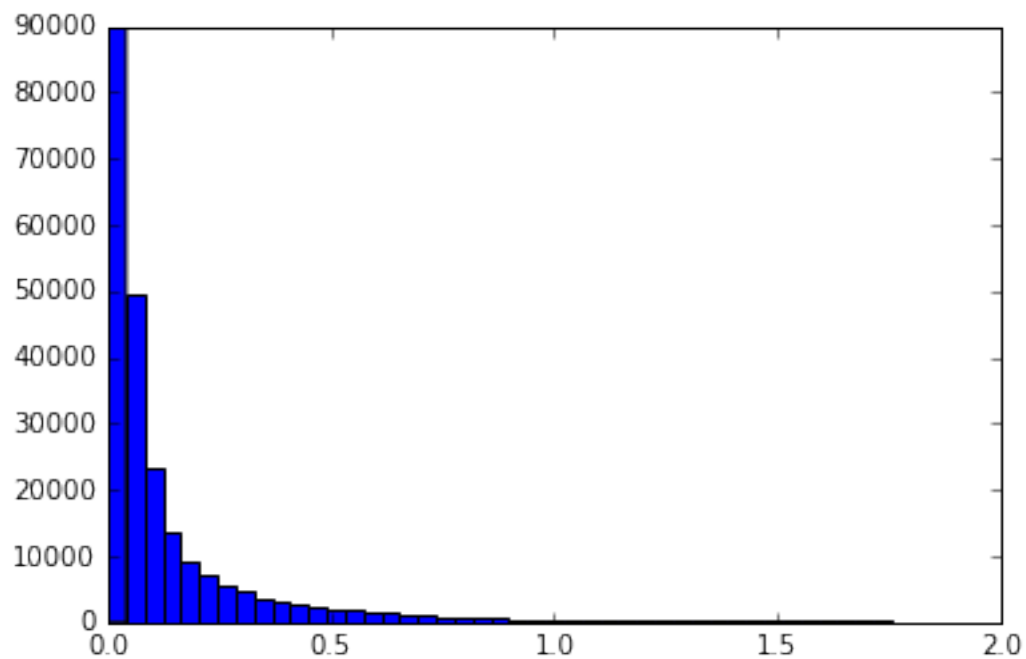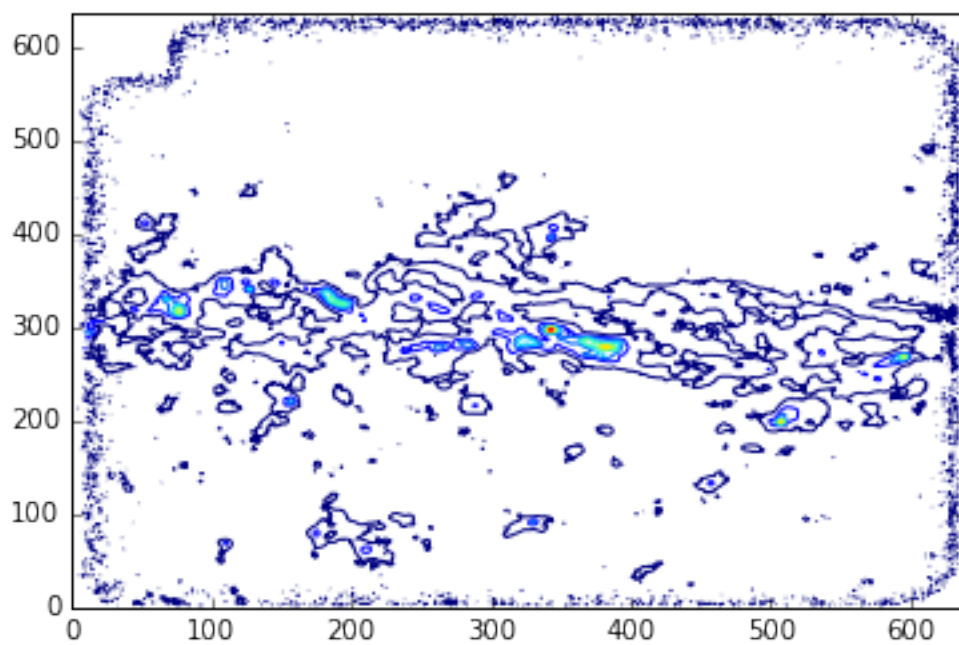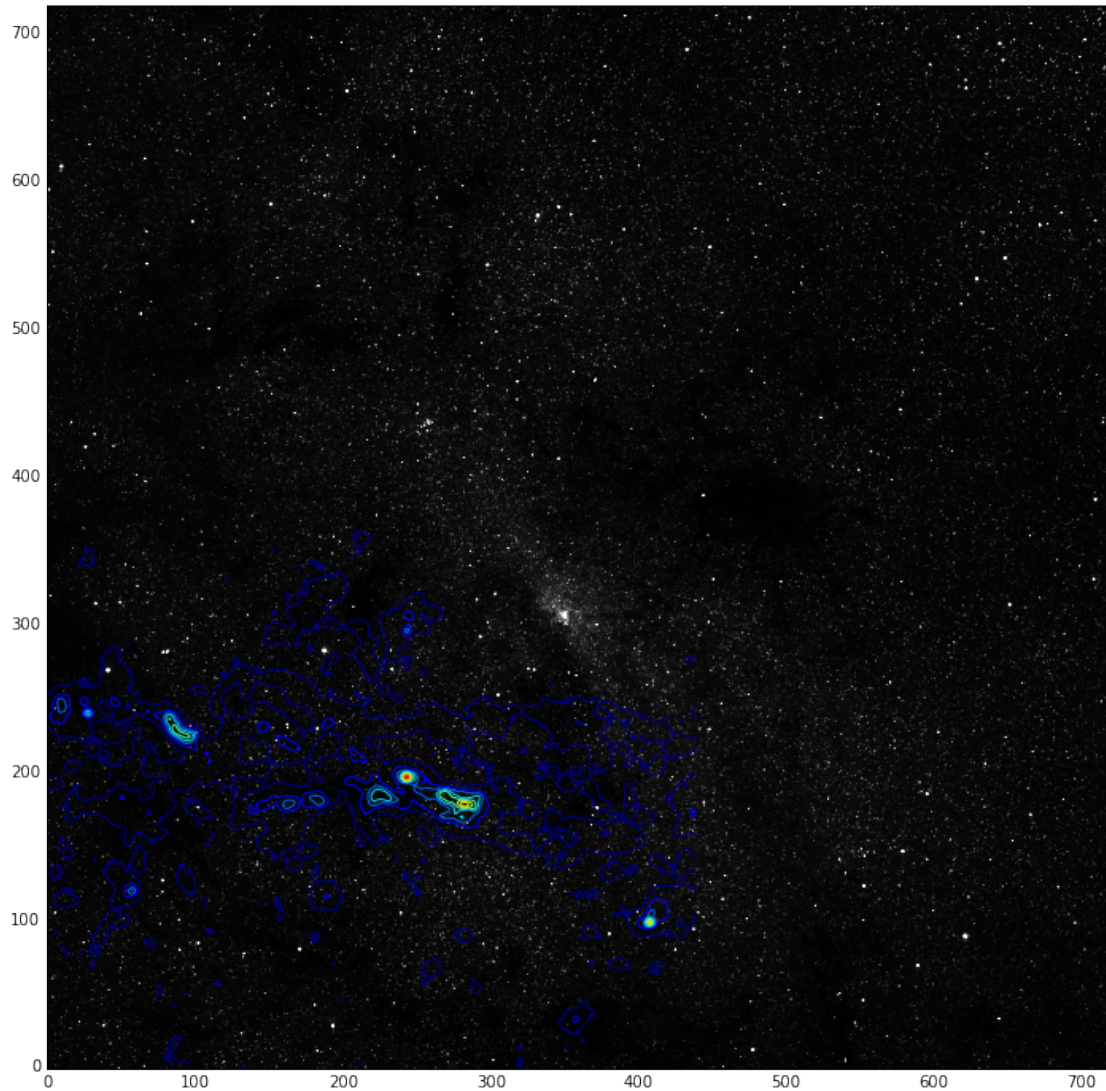
```
In [18]: pl.contour(dustdata, levels=np.linspace(0.2, 10, 10))
```

```
Out[18]: <matplotlib.contour.QuadContourSet at 0x116a67c88>
```


4

```
In [34]: pl.figure(figsize=(12,12))
         pl.imshow(stellardata, cmap='gray')
         pl.contour(dustdata[100:-100, 100:-100], levels=np.linspace(0.2, 10, 10))
```

Out[34]: <matplotlib.contour.QuadContourSet at 0x11618ab70>


```
In [35]: import aplpy
```

```
In [40]: %matplotlib nbagg
         FF = aplpy.FITSFigure('gc_2mass_k.fits')
         FF.show_grayscale(vmax=1000)
```

<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>

```
/Users/adam/anaconda/envs/esopython2016/lib/python3.5/site-packages/matplotlib/artist.py:221: Matplotlib
axes property.  A removal date has not been set.
  warnings.warn(_get_axes_msg, mplDeprecation, stacklevel=1)

INFO: Auto-setting vmin to  4.221e+02 [aplpy.core]
```

In [42]: `%matplotlib nbagg`
```python
         FF = aplpy.FITSFigure('gc_2mass_k.fits')
         FF.show_grayscale(vmax=1000)
         # convention not generally needed, only for specific (CAR) FITS projections
         FF.show_contour('gc_bolocam_gps.fits', convention='calabretta')
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


/Users/adam/anaconda/envs/esopython2016/lib/python3.5/site-packages/matplotlib/artist.py:221: Matplotlib
axes property.  A removal date has not been set.
  warnings.warn(_get_axes_msg, mplDeprecation, stacklevel=1)

INFO: Auto-setting vmin to  4.221e+02 [aplpy.core]

WARNING: FITSFixedWarning: LONPOLE2= 180.000000000 /lonpole
invalid alternate code, keyword resembles LONPOLEa but isn't. [astropy.wcs.wcs]
WARNING: FITSFixedWarning: LATPOLE2= 0.00000000000 /latpole
invalid alternate code, keyword resembles LATPOLEa but isn't. [astropy.wcs.wcs]
```

In [63]: `%matplotlib nbagg`
```python
         FF = aplpy.FITSFigure('gc_2mass_k.fits')
         FF.show_grayscale(vmax=1000)
         # convention not generally needed, only for specific (CAR) FITS projections
         FF.show_contour('gc_bolocam_gps.fits', convention='calabretta')
         scalebar = FF.add_scalebar(0.1, label='0.1$^\circ$', color='orange')
         FF.scalebar.set_corner('top right')
         FF.scalebar.set_font_size(40)
         FF.scalebar.set_font_weight('bold')
         FF.scalebar.set_linewidth(4)
         FF.scalebar.set_label('0.1$^\circ$')
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


/Users/adam/anaconda/envs/esopython2016/lib/python3.5/site-packages/matplotlib/artist.py:221: Matplotlib
axes property.  A removal date has not been set.
  warnings.warn(_get_axes_msg, mplDeprecation, stacklevel=1)

INFO: Auto-setting vmin to  4.221e+02 [aplpy.core]

WARNING: FITSFixedWarning: LONPOLE2= 180.000000000 /lonpole
invalid alternate code, keyword resembles LONPOLEa but isn't. [astropy.wcs.wcs]
WARNING: FITSFixedWarning: LATPOLE2= 0.00000000000 /latpole
invalid alternate code, keyword resembles LATPOLEa but isn't. [astropy.wcs.wcs]
```

```
In [68]: import astroquery
         from astropy import units as u

In [83]: from astroquery.irsa import Irsa
         from astroquery.vizier import Vizier
         from astroquery.eso import Eso

In [90]: Eso.ROW_LIMIT = 500

In [89]: Eso.query_instrument('naco', help=True)

INFO: List of the column_filters parameters accepted by the naco instrument query. [astroquery.eso.core]
INFO: The presence of a column in the result table can be controlled if prefixed with a [ ] checkbox. [a
INFO: The default columns in the result table are shown as already ticked: [x]. [astroquery.eso.core]

Target Information
------------------
    target:
    resolver: simbad (SIMBAD name), ned (NED name), none (OBJECT as specified by the observer)
    coord_sys: eq (Equatorial (FK5)), gal (Galactic)
    coord1:
    coord2:
    box:
    format: sexagesimal (Sexagesimal), decimal (Decimal)
[x] wdb_input_file:

Observation and proposal parameters
-----------------------------------
[ ] night:
    stime:
    starttime: 00 (00 hrs [UT]), 01 (01 hrs [UT]), 02 (02 hrs [UT]), 03 (03 hrs [UT]), 04 (04 hrs [UT])
    etime:
    endtime: 00 (00 hrs [UT]), 01 (01 hrs [UT]), 02 (02 hrs [UT]), 03 (03 hrs [UT]), 04 (04 hrs [UT]), 0
[x] prog_id:
[ ] prog_type: % (Any), 0 (Normal), 1 (GTO), 2 (DDT), 3 (ToO), 4 (Large), 5 (Short), 6 (Calibration)
[ ] obs_mode: % (All modes), s (Service), v (Visitor)
[ ] pi_coi:
    pi_coi_name: PI_only (as PI only), none (as PI or CoI)
[ ] prog_title:

Generic File Information
------------------------
[x] dp_id:
[x] ob_id:
[x] obs_targ_name:
[x] exptime:

Instrument Specific Information
-------------------------------
[x] ins_opti1_name: % (Any), 4QPM (4QPM), 4QPM_H (4QPM_H), 4QPM_K (4QPM_K), AGPM (AGPM), C_0.7 (C_0.7), C_
[x] ins_opti2_name: % (Any), CutOff_2.5um (CutOff_2.5um), FPI (FPI), open (open)
[x] ins_opti3_name: % (Any), 7Holes (7Holes), 9Holes (9Holes), 18Holes (18Holes), APP_coro (APP_coro), Ap
[x] ins_opti4_name: % (Any), Grism1 (Grism1), Grism2 (Grism2), Grism3 (Grism3), Grism4 (Grism4), J (J),
[x] ins_opti5_name: % (Any), Br_gamma% (Br_gamma), NB_2.12% (NB_2.12), NB_2.17% (NB_2.17), NB_4.05% (NB_4.
[x] ins_opti6_name: % (Any), H (H), HeI (HeI), J (J), K (K), Ks (Ks), L (L), L_prime (L_prime), M_prime (
```

```
[x] ins_opti7_name: % (Any), L27 (L27), L54 (L54), S100 (S100), S13 (S13), S27 (S27), S54 (S54), SDI (SD
[ ] det_ncorrs_name: % (Any), Double_RdRstRd (Double_RdRstRd), FowlerNsamp (FowlerNsamp), Uncorr (Uncorr)
[ ] det_mode_name: % (Any), HighBackground (HighBackground), HighDynamic (HighDynamic), HighSensitivity
[ ] det_dit:
[ ] det_ndit:
[ ] aos_ins_dich_posnam: % (Any), JHK (JHK), K (K), N20C80 (N20C80), N90C10 (N90C10), VIS (VIS)
[ ] aos_ocs_wfs_type: % (Any), IR (IR), LGS (LGS), VIS (VIS)


Ambient Parameters
-------------------
[x] fwhm_avg:
[ ] airmass_range:
[ ] night_flag: % (Any), 0 (Night), 1 (Twilight), 2 (Daytime)
[ ] moon_illu:


Result set
----------
    order:  (nothing (faster)), dp_id (Observation Time), dp_cat (DPR.CATG), dp_tech (DPR.TECH), tpl_sta
```

In [85]: tbl = Eso.query_instrument('naco', target='Sgr A*')
         tbl

Out[85]: <Table masked=True length=50>
```
        Object       Target Ra Dec        ... INS.OPTI7.NAME  DIMM S-avg
         str8            str23            ...      str4         str11
        -------- ------------------------ ... -------------- -----------
           Sgr_A 17:45:39.98 -29:00:24.0 ...           25_S 2.12 [1.57]
           Sgr_A 17:45:39.56 -29:00:23.8 ...           25_S 1.56 [0.94]
           Sgr_A 17:45:39.48 -29:00:27.2 ...           25_S 1.56 [0.94]
           Sgr_A 17:45:40.14 -29:00:32.8 ...           25_S 1.50 [0.88]
           Sgr_A 17:45:40.07 -29:00:25.5 ...           25_S 1.50 [0.88]
           Sgr_A 17:45:39.89 -29:00:24.4 ...           25_S 1.42 [0.78]
           Sgr_A 17:45:39.94 -29:00:31.8 ...           25_S 1.42 [0.78]
           Sgr_A 17:45:39.73 -29:00:30.9 ...           25_S 1.46 [0.81]
           Sgr_A 17:45:39.44 -29:00:30.9 ...           25_S 1.46 [0.81]
           Sgr_A 17:45:40.16 -29:00:23.0 ...           25_S 1.78 [1.18]
             ...                      ... ...            ...         ...
        OphE-MM3 17:45:39.57 -29:00:18.4 ...           25_L         N/A
        OphE-MM3 17:45:39.95 -29:00:22.7 ...           25_L         N/A
        OphE-MM3 17:45:40.25 -29:00:20.2 ...           25_L         N/A
        OphE-MM3 17:45:40.18 -29:00:24.0 ...           25_L         N/A
        OphE-MM3 17:45:40.05 -29:00:27.6 ...           25_L         N/A
        OphE-MM3 17:45:39.83 -29:00:25.0 ...           25_L         N/A
        OphE-MM3 17:45:39.62 -29:00:26.0 ...           25_L         N/A
        OphE-MM3 17:45:39.80 -29:00:19.9 ...           25_L         N/A
        OphE-MM3 17:45:39.62 -29:00:23.0 ...           25_L         N/A
        OphE-MM3 17:45:39.94 -29:00:18.3 ...           25_L         N/A
```

In [71]: rslt = Irsa.query_region('Sgr A*', radius=10*u.arcmin, catalog='pt_src_cat')
         #rslt

In [79]: bright = rslt[rslt['k_m'] < 9]

In [93]: %matplotlib nbagg
         FF = aplpy.FITSFigure('gc_2mass_k.fits')

```python
        FF.show_grayscale(vmax=1000, invert=True)
        # convention not generally needed, only for specific (CAR) FITS projections
        FF.show_contour('gc_bolocam_gps.fits', convention='calabretta', colors=['r'])
        scalebar = FF.add_scalebar(0.1, label='0.1$^\circ$', color='orange')
        FF.scalebar.set_corner('top right')
        FF.scalebar.set_font_size(40)
        FF.scalebar.set_font_weight('bold')
        FF.scalebar.set_linewidth(4)
        FF.scalebar.set_label('0.1$^\circ$')
        FF.show_markers(bright['ra'], bright['dec'])
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


/Users/adam/anaconda/envs/esopython2016/lib/python3.5/site-packages/matplotlib/artist.py:221: Matplotlib
axes property.  A removal date has not been set.
  warnings.warn(_get_axes_msg, mplDeprecation, stacklevel=1)

INFO: Auto-setting vmin to  4.221e+02 [aplpy.core]

WARNING: FITSFixedWarning: LONPOLE2= 180.000000000 /lonpole
invalid alternate code, keyword resembles LONPOLEa but isn't. [astropy.wcs.wcs]
WARNING: FITSFixedWarning: LATPOLE2= 0.00000000000 /latpole
invalid alternate code, keyword resembles LATPOLEa but isn't. [astropy.wcs.wcs]
```

Aside: running external code files

```python
In [ ]: %run file.py
        %run -i file.py
        execfile('file.py') # is equivalent to %run -i ...
```

Visualization Part 2: pyds9

```python
In [5]: import pyds9
        from astropy.io import fits
```

```python
In [6]: DD = pyds9.DS9('mine')
```

```python
In [7]: DD.set('frame 1')
```

```
Out[7]: 1
```

```python
In [8]: DD.set_pyfits(fits.open('gc_2mass_k.fits'))
```

```
Out[8]: 1
```

```python
In [9]: DD.set('frame lock wcs')
        DD.set('frame 2')
        DD.set_pyfits(fits.open('gc_bolocam_gps.fits'))
```

```
Out[9]: 1
```

```python
In [12]: DD.set('single')
```

```
Out[12]: 1
```

For more information about ds9 xpa access points, see: ds9.si.edu/doc/ref/xpa.html