FITS文件操作入门

余 恒

北师大天文系

2019.10.24

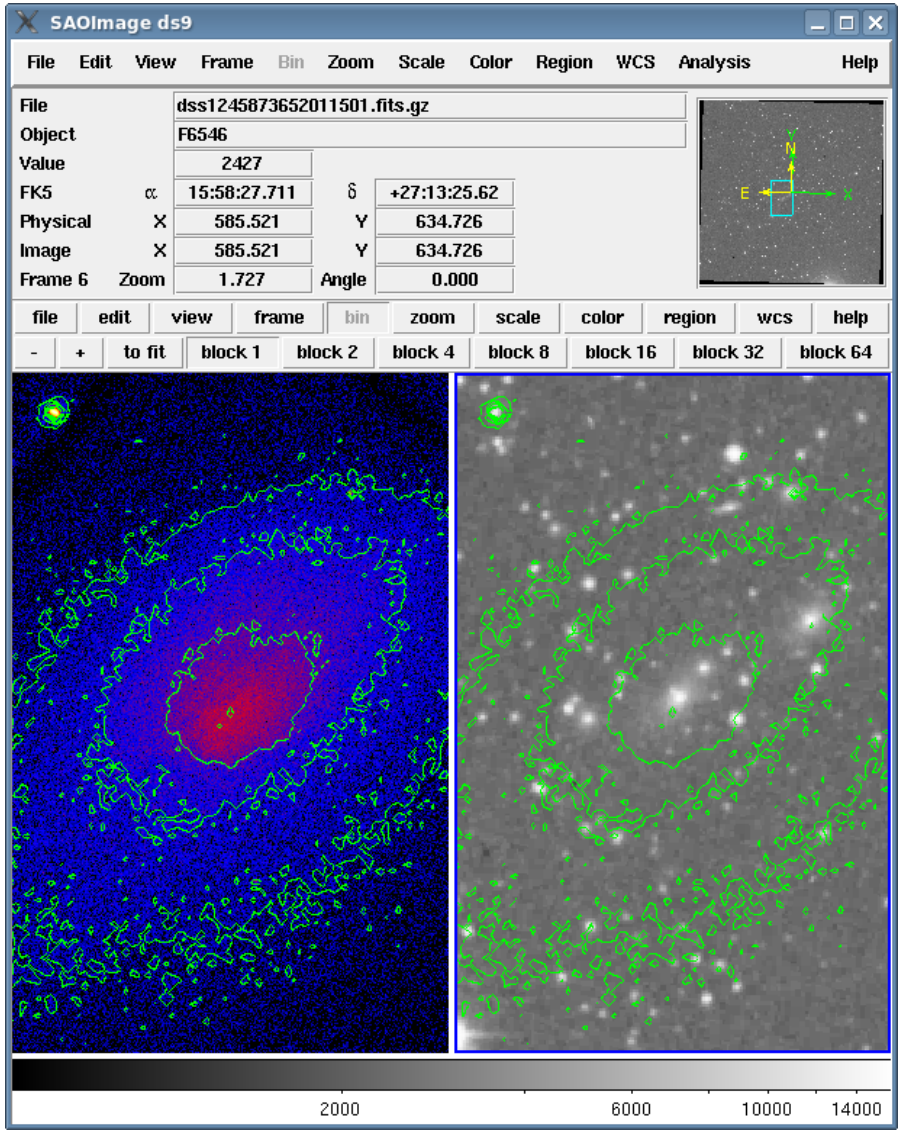
什么是FITS

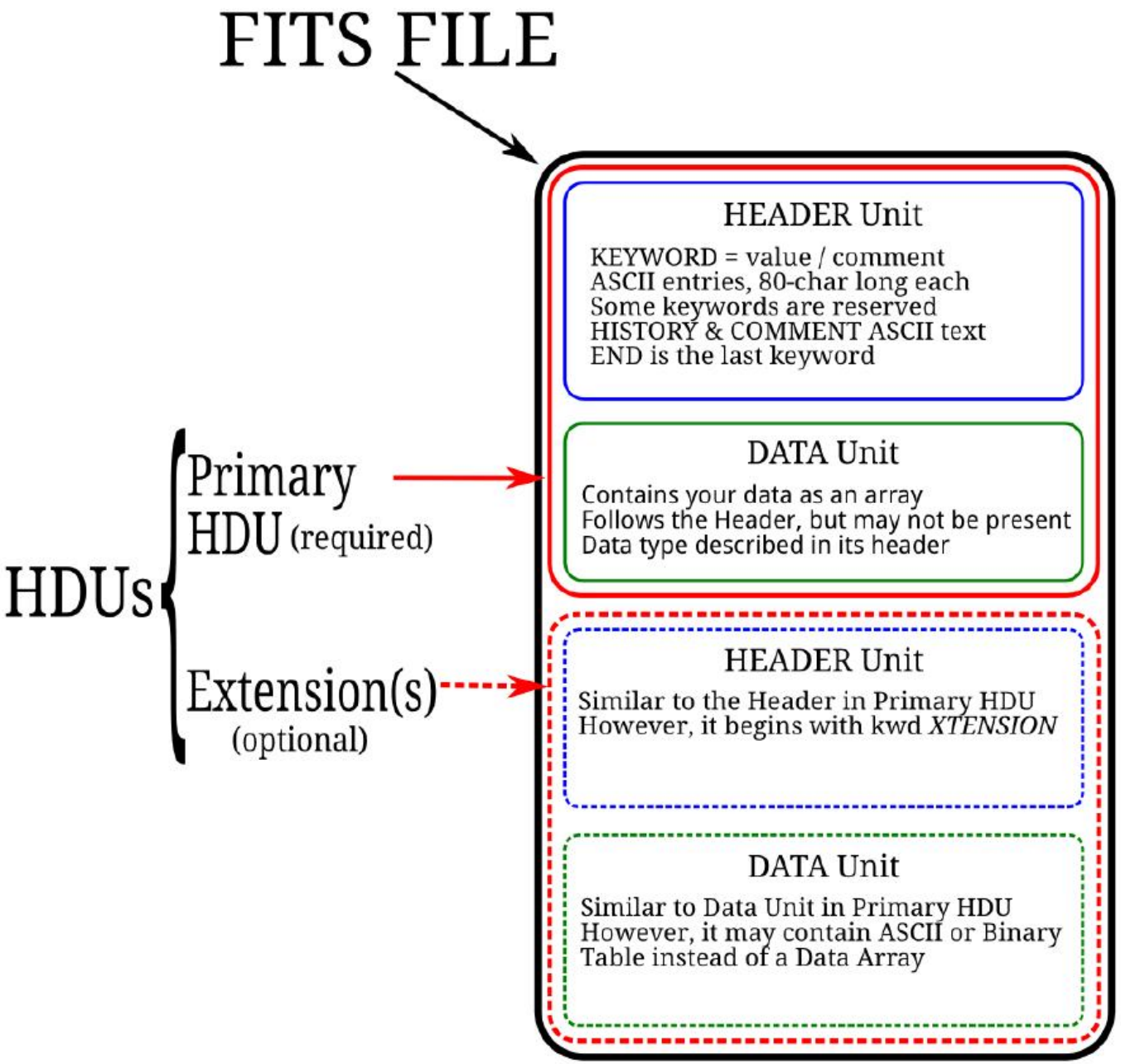
- “普适图像传输系统”(Flexible Image Transport System),专为不同平台之间的数据交换而设计。
- 可以记录包括图像、表格、多维数组在内的各类数据。
- 1988年国际天文学联合会（IAU）大会指定IAU的FITS工作组全权负责此格式的修订。
- IAU规定，FITS标准的修改不得破坏前后一致性，也就是说“once FITS, always FITS”。
- 最新的FITS文件标准为2018年公布的4.0。

如何使用FITS

- 天文界各编程语言都有支持读写FITS文件的函数库
  - CFITSIO（C/Fortran）
  - C++ FITS（C++）
  - PyFITS、Astropy（Python）

- 专业软件：
  - 图像：ds9、fv、Aladin、MaximDL ……
  - 表格：Topcat、 fv、STIL





FITS文件结构

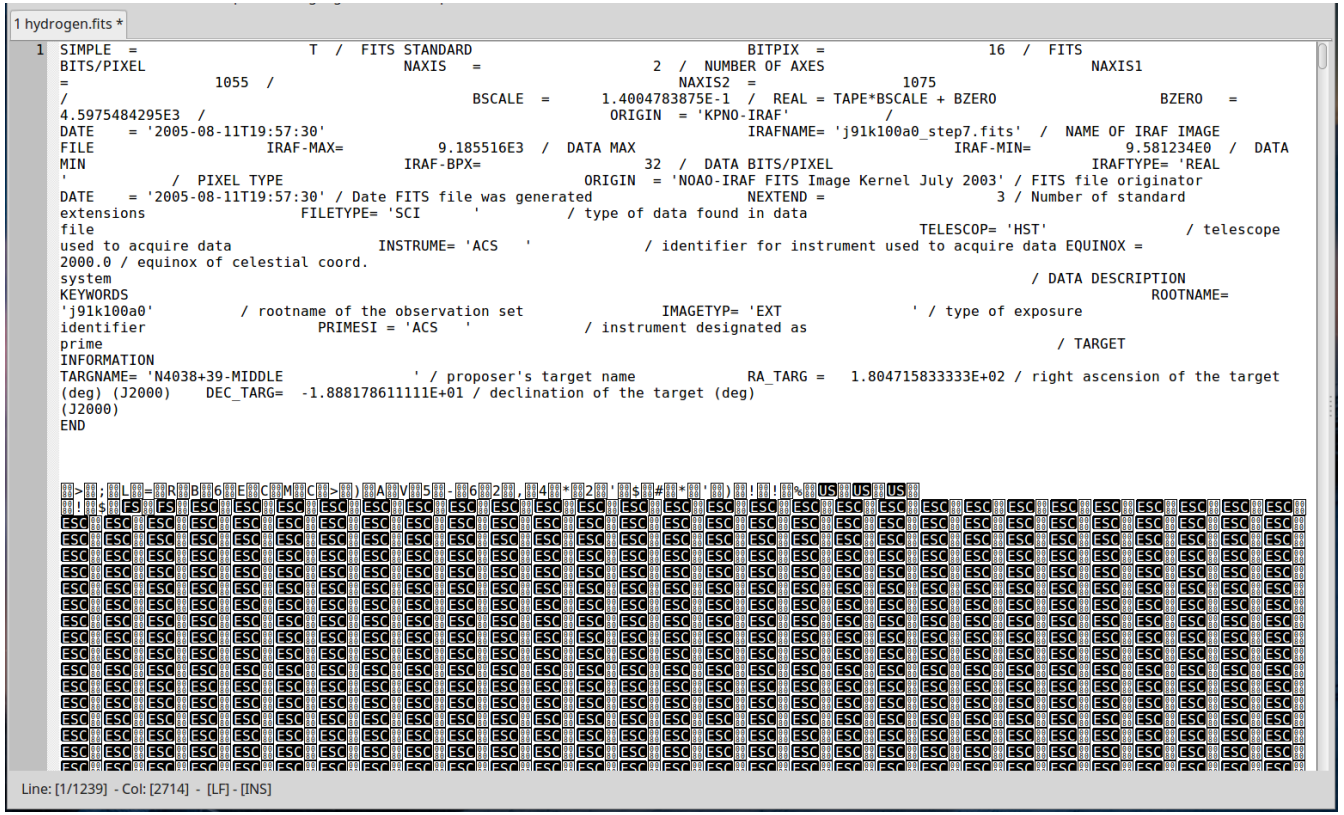
- primary header and data unit(HDU)

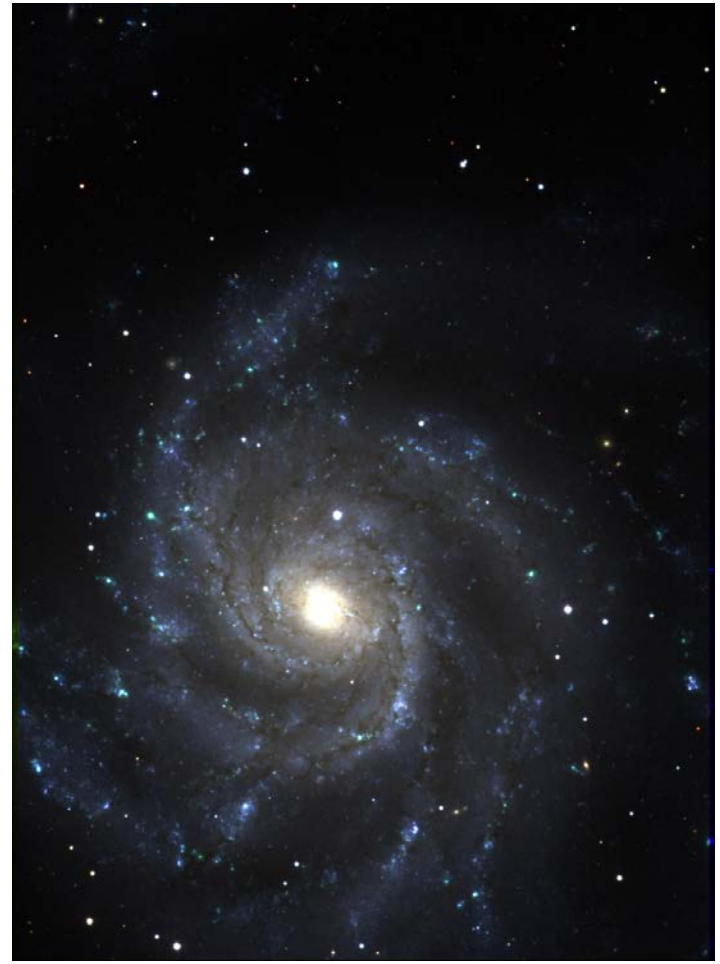
  基本头单元与数据单元
- Conforming Extensions(optional)

  可选的扩展
- 基础HDU组成的FITS文称为基础FITS文件（Basic FITS file)
- 包含扩展部分的FITS文件称为多扩展文件 （Multi-Extension FITS,MEF)

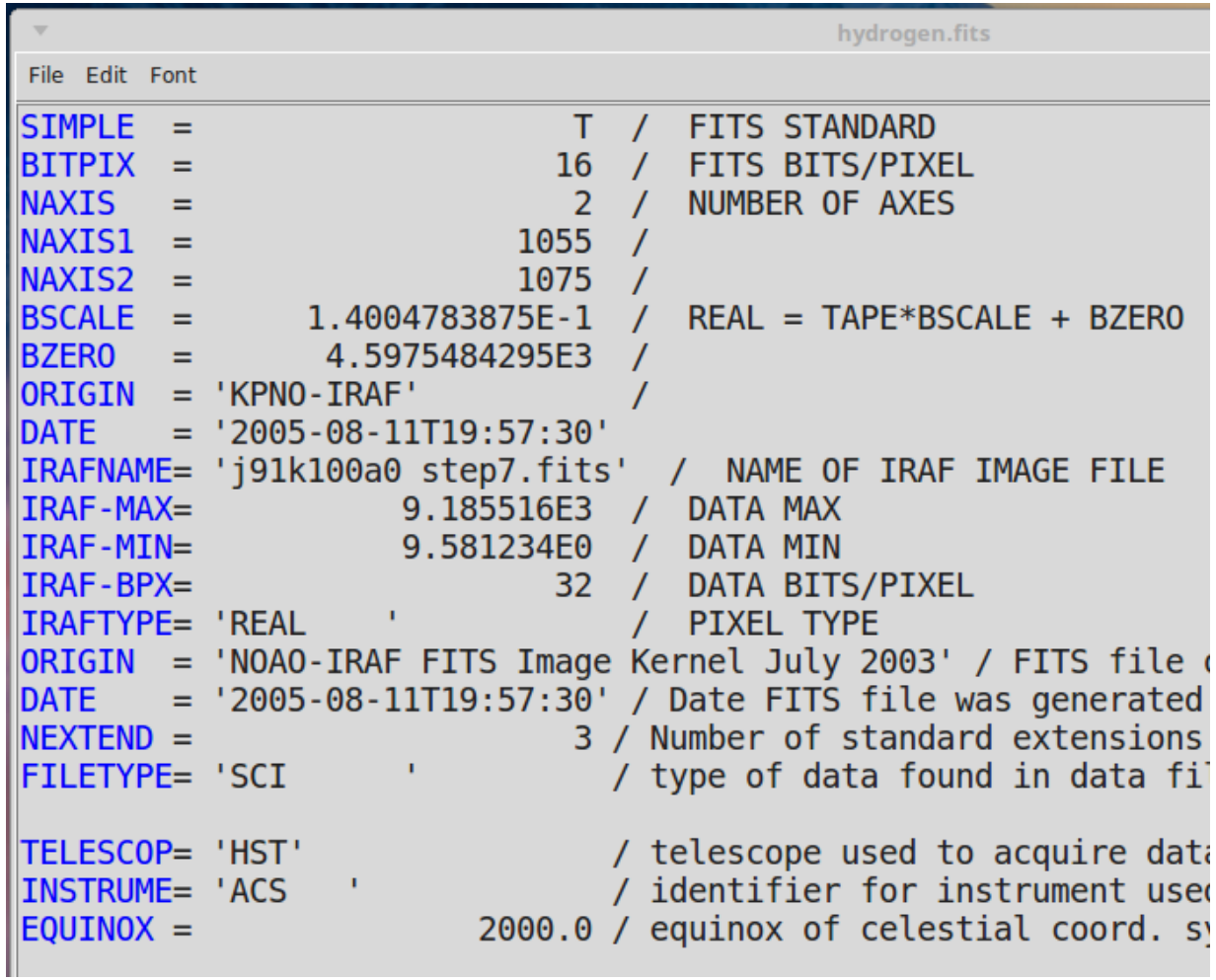
数据准备

- http://paperdata.china-vo.org/yuheng/material.zip
- 表格 gal_info_dr7_v5_2.fit.gz
- 图像 frame-[gir]-003712-3-0187.fits





FITS文件头 Header



- 以2880字节为单位组织
- 每行80个字符 （第8位为等号）
- 行数为36的倍数。
- 只包含ASCII字符
- 以END结尾
- 五个关键变量
  - SIMPLE 逻辑变量,指明文件是否符合基本 FITS 标准。
  - BITPIX 整数变量,指明每一个像元值的位数。
  - NAXIS 整数变量,指明坐标轴数。
  - NAXIS1 整数变量,指明在数组内轴1像元数。
  - NAXIS2 整数变量,指明在数组内轴2像元数。

python+ FITS

- PyFITS

  https://pythonhosted.org/pyfits/

  2016年1月29日最后一次独立发布 v3.4.0

- astroPy

  https://github.com/astropy/astropy

  Read, write and manipulate all aspects of FITS files

  (Extensions, headers, images, tables)

```
from astropy.io import fits as pyfits
```

- 其他模块： fitsio (cfitsio)，sunpy

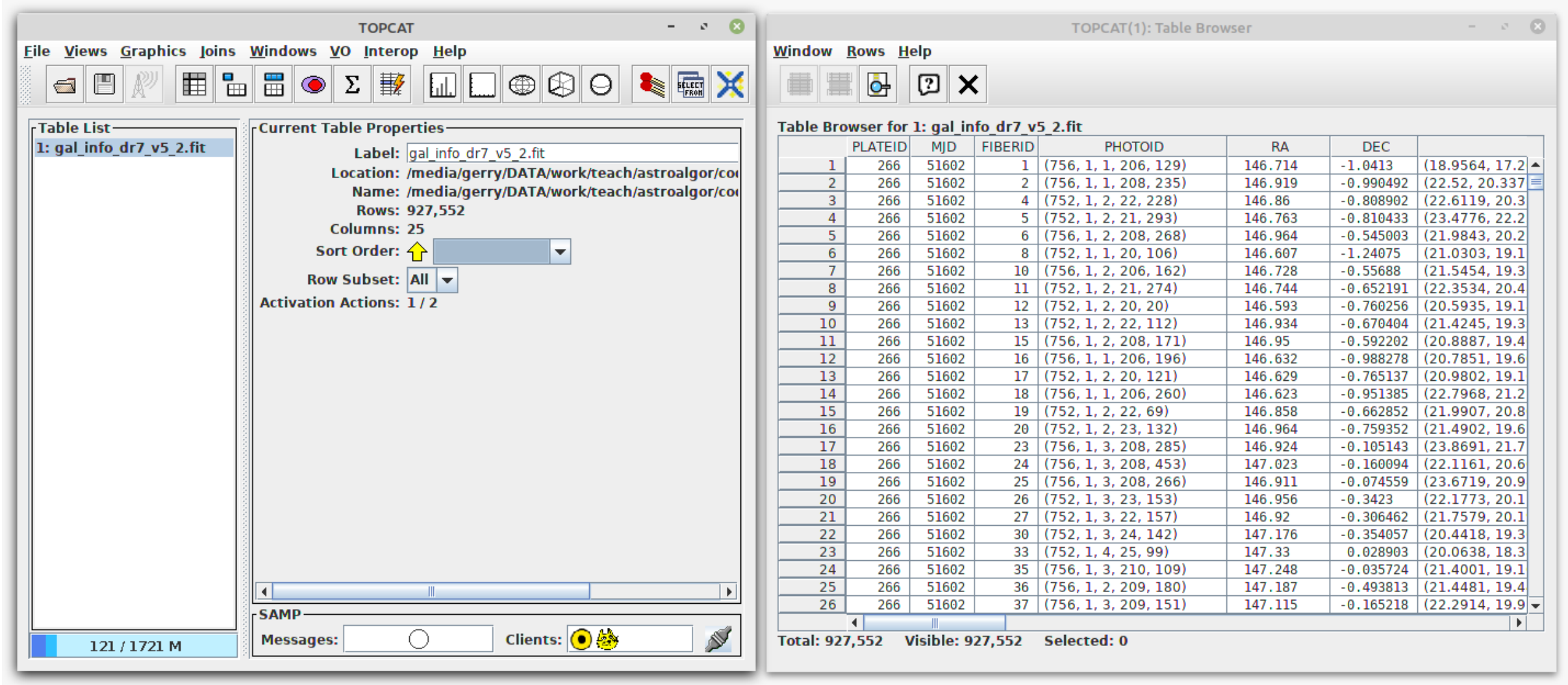
FITS内容检查

- `hdu = fits.open('frame-g-003712-3-0187.fits')`
- `hdu.info()`

```
Filename: frame-g-003712-3-0187.fits
No.    Name      Ver    Type      Cards   Dimensions   Format
  0  PRIMARY       1 PrimaryHDU      96   (2048, 1489)   float32
  1                1 ImageHDU         6   (2048,)   float32
  2                1 BinTableHDU     27   1R x 3C   [49152E, 2048E, 1489E]
  3                1 BinTableHDU     79   1R x 31C   [J, 3A, J, A, D, D, 2J, J, …  E]

Filename: gal_info_dr7_v5_2.fit
No.    Name      Ver    Type      Cards   Dimensions   Format
  0  PRIMARY       1 PrimaryHDU       4   ()
  1                1 BinTableHDU     67   927552R x 25C   [I, J, I, 5I, E, E, … 12A]
```

1. FITS表格



	PLATEID	MJD	FIBERID	PHOTOID	RA	DEC	
1	266	51602	1	(756, 1, 1, 206, 129)	146.714	-1.0413	(18.9564, 17.2
2	266	51602	2	(756, 1, 1, 208, 235)	146.919	-0.990492	(22.52, 20.337
3	266	51602	4	(752, 1, 2, 22, 228)	146.86	-0.808902	(22.6119, 20.3
4	266	51602	5	(752, 1, 2, 21, 293)	146.763	-0.810433	(23.4776, 22.2
5	266	51602	6	(756, 1, 2, 208, 268)	146.964	-0.545003	(21.9843, 20.2
6	266	51602	8	(752, 1, 1, 20, 106)	146.607	-1.24075	(21.0303, 19.1
7	266	51602	10	(756, 1, 2, 206, 162)	146.728	-0.55688	(21.5454, 19.3
8	266	51602	11	(752, 1, 2, 21, 274)	146.744	-0.652191	(22.3534, 20.4
9	266	51602	12	(752, 1, 2, 20, 20)	146.593	-0.760256	(20.5935, 19.1
10	266	51602	13	(752, 1, 2, 22, 112)	146.934	-0.670404	(21.4245, 19.3
11	266	51602	15	(756, 1, 2, 208, 171)	146.95	-0.592202	(20.8887, 19.4
12	266	51602	16	(756, 1, 1, 206, 196)	146.632	-0.988278	(20.7851, 19.6
13	266	51602	17	(752, 1, 2, 20, 121)	146.629	-0.765137	(20.9802, 19.1
14	266	51602	18	(756, 1, 1, 206, 260)	146.623	-0.951385	(22.7968, 21.2
15	266	51602	19	(752, 1, 2, 22, 69)	146.858	-0.662852	(21.9907, 20.8
16	266	51602	20	(752, 1, 2, 23, 132)	146.964	-0.759352	(21.4902, 19.6
17	266	51602	23	(756, 1, 3, 208, 285)	146.924	-0.105143	(23.8691, 21.7
18	266	51602	24	(756, 1, 3, 208, 453)	147.023	-0.160094	(22.1161, 20.6
19	266	51602	25	(756, 1, 3, 208, 266)	146.911	-0.074559	(23.6719, 20.9
20	266	51602	26	(752, 1, 3, 23, 153)	146.956	-0.3423	(22.1773, 20.1
21	266	51602	27	(752, 1, 3, 22, 157)	146.92	-0.306462	(21.7579, 20.1
22	266	51602	30	(752, 1, 3, 24, 142)	147.176	-0.354057	(20.4418, 19.3
23	266	51602	33	(752, 1, 4, 25, 99)	147.33	0.028903	(20.0638, 18.3
24	266	51602	35	(756, 1, 3, 210, 109)	147.248	-0.035724	(21.4001, 19.1
25	266	51602	36	(756, 1, 2, 209, 180)	147.187	-0.493813	(21.4481, 19.4
26	266	51602	37	(756, 1, 3, 209, 151)	147.115	-0.165218	(22.2914, 19.9

FITS 表格读取

```
from astropy.io import fits
hdu = fits.open('gal_info_dr7_v5_2.fit')
print(hdu[0].header)
print(hdu['primary'].header)
print(repr(hdu[0].header))
print(hdu[0].header.keys())
```

```
SIMPLE  =                    T
BITPIX  =                    8
NAXIS   =                    0
EXTEND  =                    T
['SIMPLE', 'BITPIX', 'NAXIS', 'EXTEND']
```

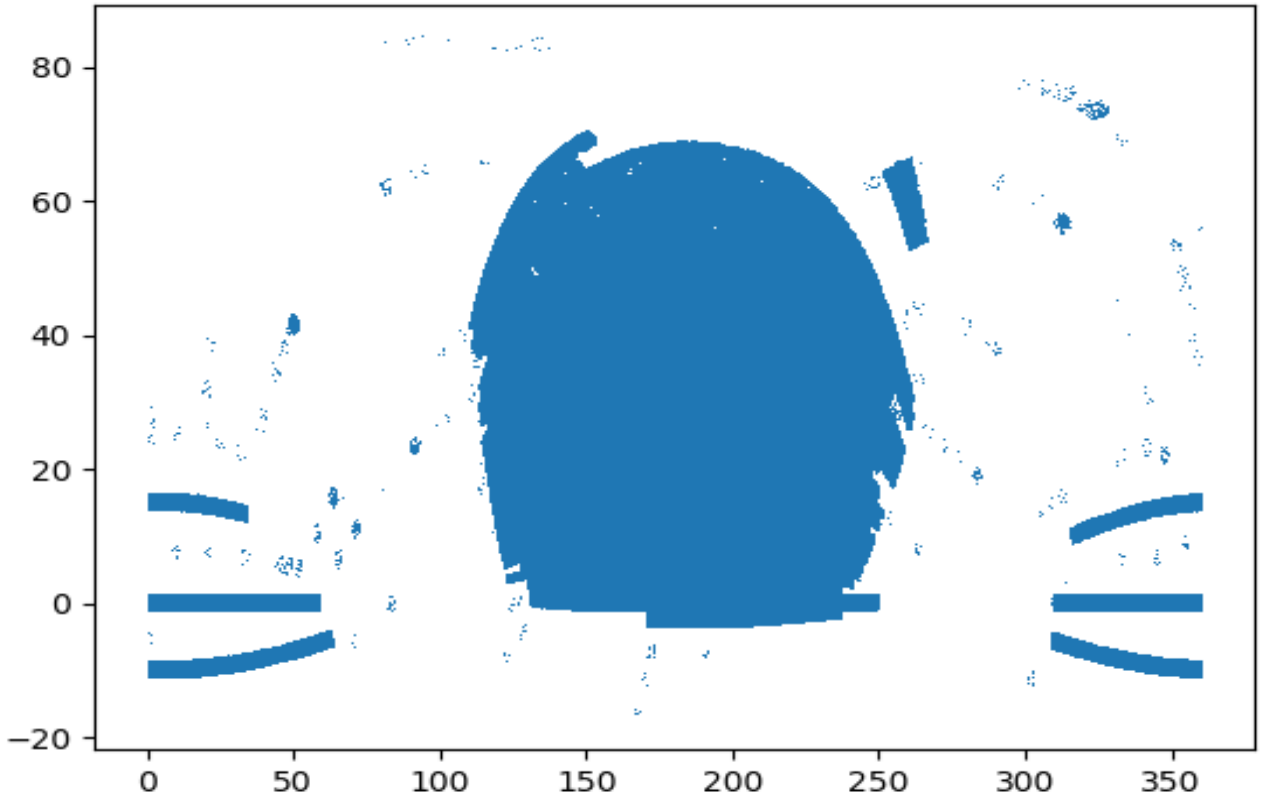
```
print(repr(hdu[1].header[:5]))
print(len(hdu[1].data))
print(hdu[1].data.names)
ra = hdu[1].data['ra']
dec = hdu[1].data.field(5)
hdu.close()
```

```
['PLATEID', 'MJD', 'FIBERID', 'PHOTOID',
'RA', 'DEC', 'PLUG_MAG', 'PRIMTARGET',
'SECTARGET', 'TARGETTYPE', 'SPECTROTYPE',
'SUBCLASS', 'Z', 'Z_ERR', 'Z_WARNING',
'V_DISP', 'V_DISP_ERR', 'SN_MEDIAN',
'E_BV_SFD', 'ZTWEAK', 'ZTWEAK_ERR',
'SPECTRO_MAG', 'KCOR_MAG',
'KCOR_MODEL_MAG', 'RELEASE']
```

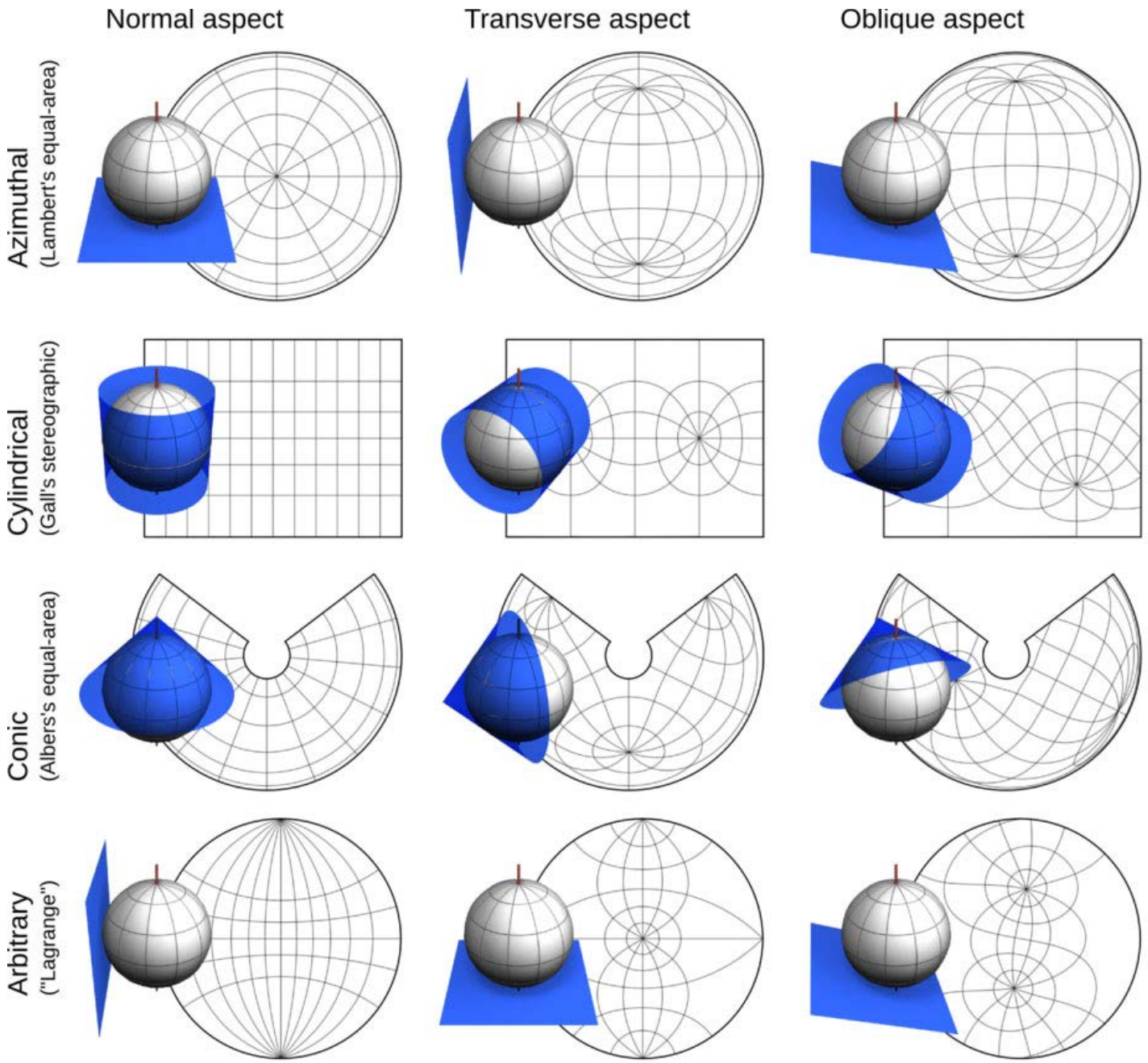
1.1 天区覆盖

- `import pylab as plt`
- `plt.figure()`
- `plt.plot(ra,dec,',')`
- `plt.show()`

- `plt.xlim([0,360])`
- `plt.ylim([-90,90])`



- `plt.plot(ra[ra>-360],dec[ra>-360],',')`
- `Plt.grid("on")`

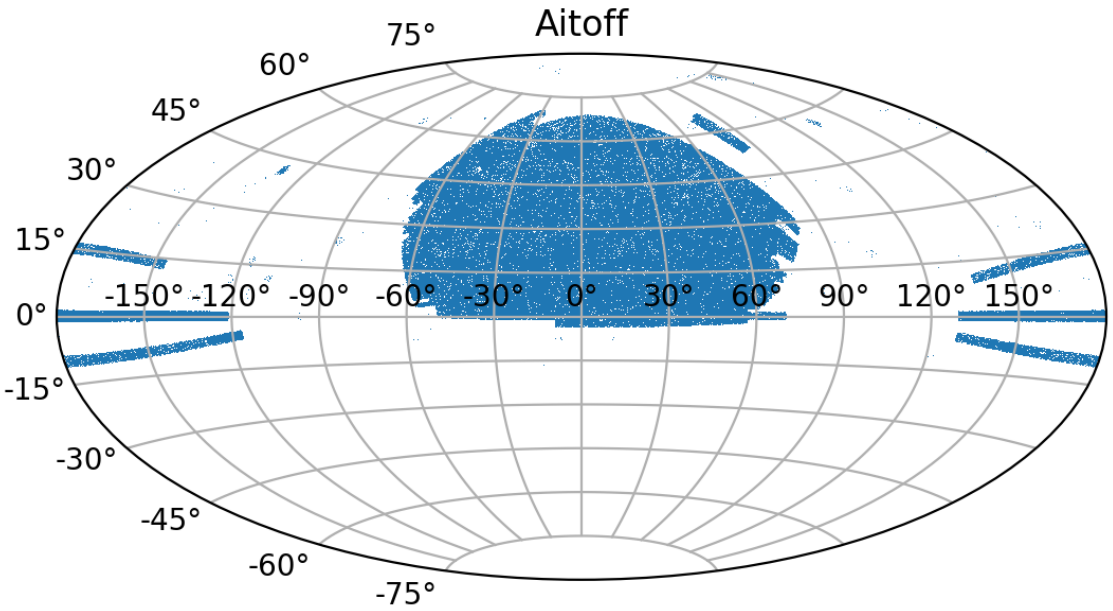


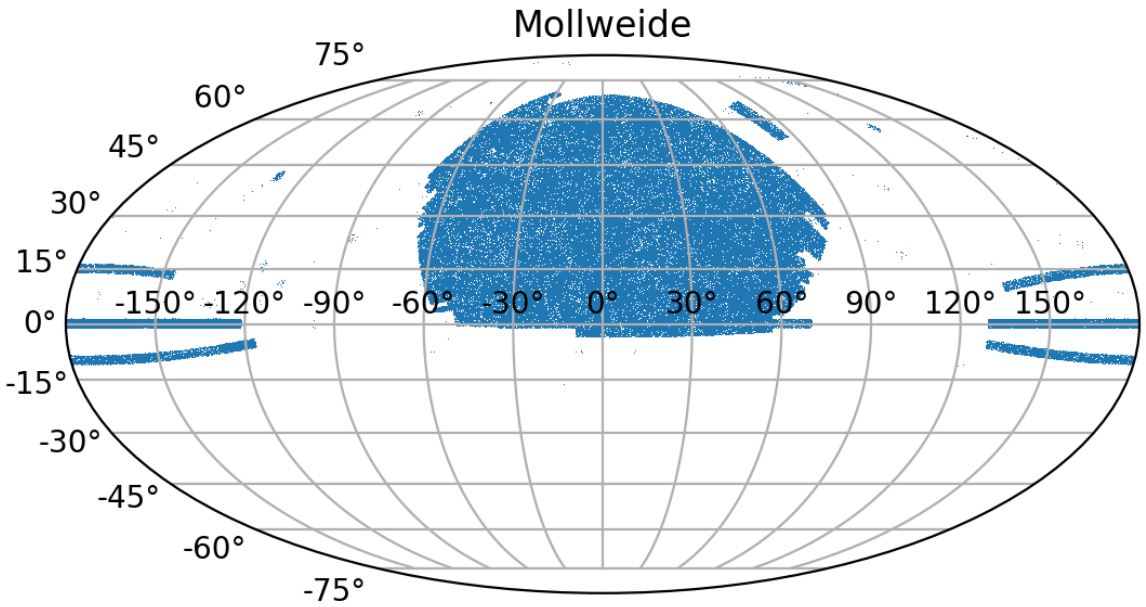
投影分类

- 按特性分：
  - 等角投影：墨卡托投影、极座标
  - 等积投影：Hammer，mollweide
  - 等距投影：Aitoff
- 按方法分：
  - 方位投影、圆柱投影、圆锥投影、任意投影
  - 正轴、横轴、斜轴

投影图(subplot)

- `plt.subplot(111, projection="aitoff")`
- `plt.axes(projection='mollweide')`
- `'aitoff', 'hammer', 'lambert', 'mollweide', 'polar', ......`



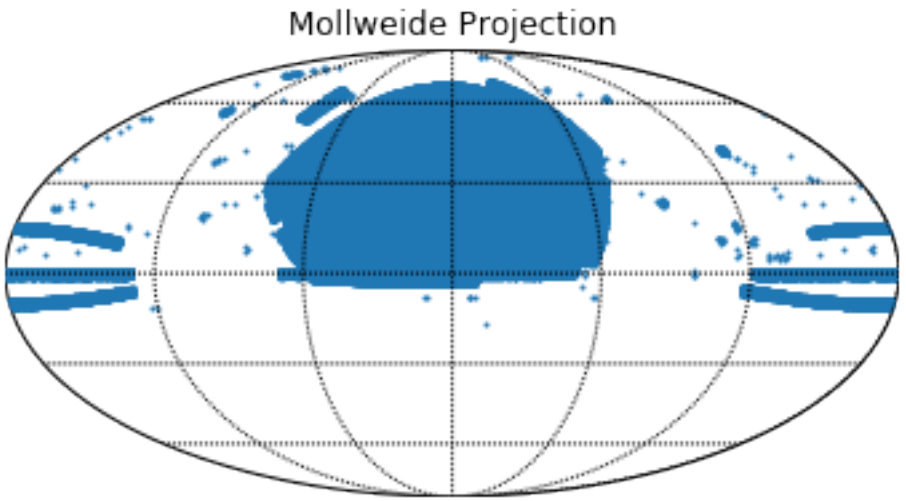


Basemap

- Matplotlib框架下的地图投影模块
- 下一代为Cartopy
- 自动安装：
  - Anaconda 提示符下输入 conda install basemap
- 手动安装：
  - 网站 https://www.lfd.uci.edu/~gohlke/pythonlibs/ 下载安装包
    - basemap-1.2.0-cp36-cp36m-win32.whl
    - basemap-1.2.0-cp36-cp36m-win_amd64.whl
  - pip install basemap-1.2.0-*
  - 但需要解决依赖关系，如geos, pyproj 等

Basemap 全天投影

```
from mpl_toolkits.basemap import Basemap
import numpy as np
import matplotlib.pyplot as plt
import os
# os.environ['PROJ_LIB'] = 'PATH\\Anaconda3\\Library\\share'  # for windows
os.environ['PROJ_LIB'] = 'PATH/anaconda3/share/proj'    # for linux
m = Basemap(projection='moll',lon_0=180)
xpt, ypt = m(ra[::4], dec[::4])
m.plot(xpt,ypt, '.', ms=2)
# draw parallels and meridians.
m.drawparallels(np.arange(-90.,120.,30.))
m.drawmeridians(np.arange(0.,420.,60.))
plt.title("Mollweide Projection")
```

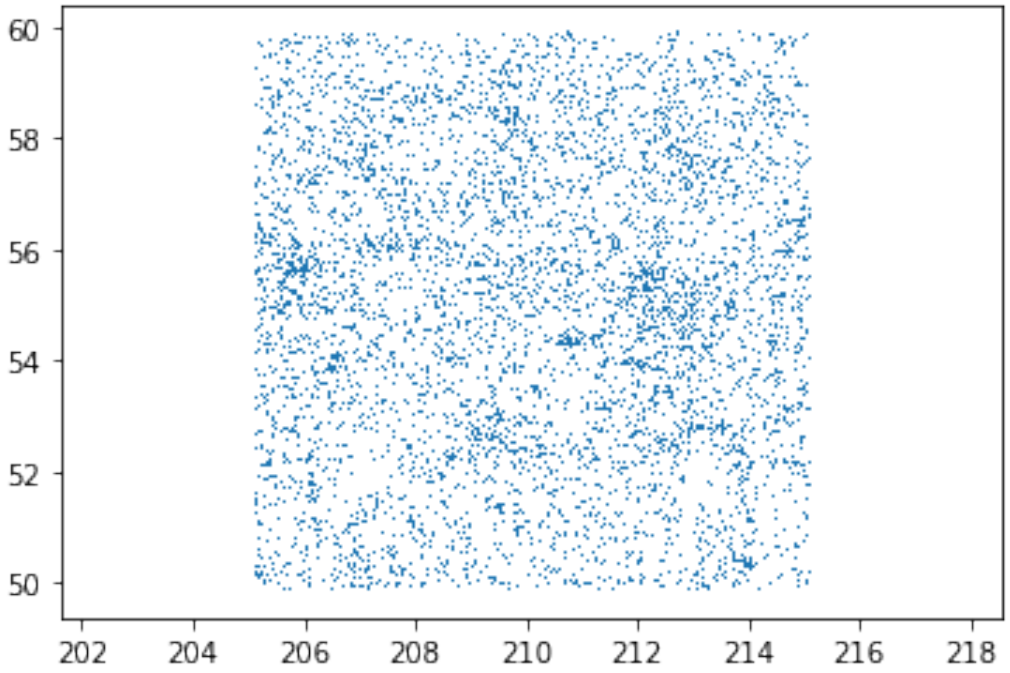


1.2 星表提取（方形）

- 从星表中提取星系 M101

(210.08759 , 54.876254) 周围

5度视场内所有星系的 RA, DEC, z



- `rad = 5`
- `centroid = [210.08759 , 54.876254]`
- `ind = ( np.abs(ra -centroid[0])< rad ) *`
  `( np.abs(dec -centroid[1])< rad )`
- `print(np.sum(ind))`
- `field = np.r_[ra[ind],dec[ind],z[ind]]`

SkyCoord 模块

- `from astropy import units as u`
- `from astropy.coordinates import SkyCoord`
- `c = SkyCoord(ra=10.625*u.degree, dec=41.2*u.degree)`
- `c = SkyCoord(10.625, 41.2, unit='deg')`
- `c = SkyCoord('00h42m30s', '+41d12m00s')`
- `c = SkyCoord('00h42.5m', '+41d12m')`
- `c = SkyCoord('00 42 30 +41 12 00', unit=(u.hourangle, u.deg))`
- `c = SkyCoord('00:42.5 +41:12', unit=(u.hourangle, u.deg))`

- `c.ra, c.ra.hour, c.ra.hms, c.dec,`
- `c.dec.degree, c.dec.radian`

坐标转换

- `c.to_string('decimal')`
- `c.to_string('dms')`
- `c.to_string('hmsdms')`
- `c.galactic`
- `c.galactic.l`
- `c.galactic.b`

- `u'10.625 41.2'`
- `u'10d37m30s 41d12m00s'`
- `u'00h42m30s +41d12m00s'`
- `<SkyCoord (Galactic): (l, b) in deg (121.12334339, -21.6403587)>`

- `gc = SkyCoord(l=0*u.degree, b=45*u.degree, frame='galactic')`
- `gc.fk5, gc.icrs`

角距

- `c1 = SkyCoord(ra=10*u.degree, dec=9*u.degree)`
- `c2 = SkyCoord(ra=11*u.degree, dec=10*u.degree)`
- `c1.separation(c2)`
- `c1.separation(c2).degree`

- `c1 = SkyCoord(ra=10*u.degree, dec=9*u.degree,frame="fk5")`
- `c1 = SkyCoord(ra=10*u.degree, dec=9*u.degree,frame="icrs")`
- `c1.separation(c2)`

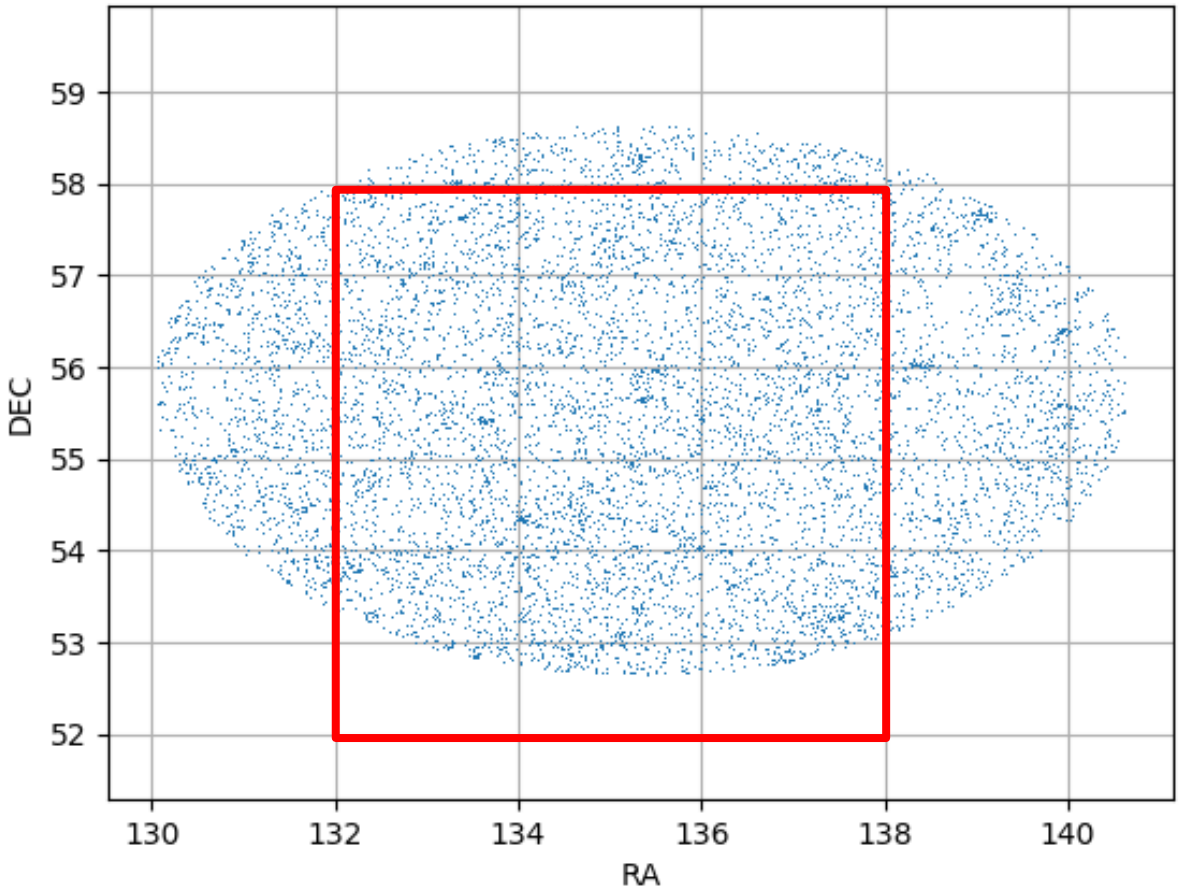
1.2 星表提取 （圆形区域）

```
rad = 5
c1 = SkyCoord(ra_n, dec_n,unit="deg")
c2 = SkyCoord(ra=centroid[0]*u.degree,
              dec=centroid[1]*u.degree)
ind = c1.separation(c2).degree < rad
print(np.sum(ind))
field = np.r_[ra_n[ind],dec_n[ind],z_n[ind]]
```

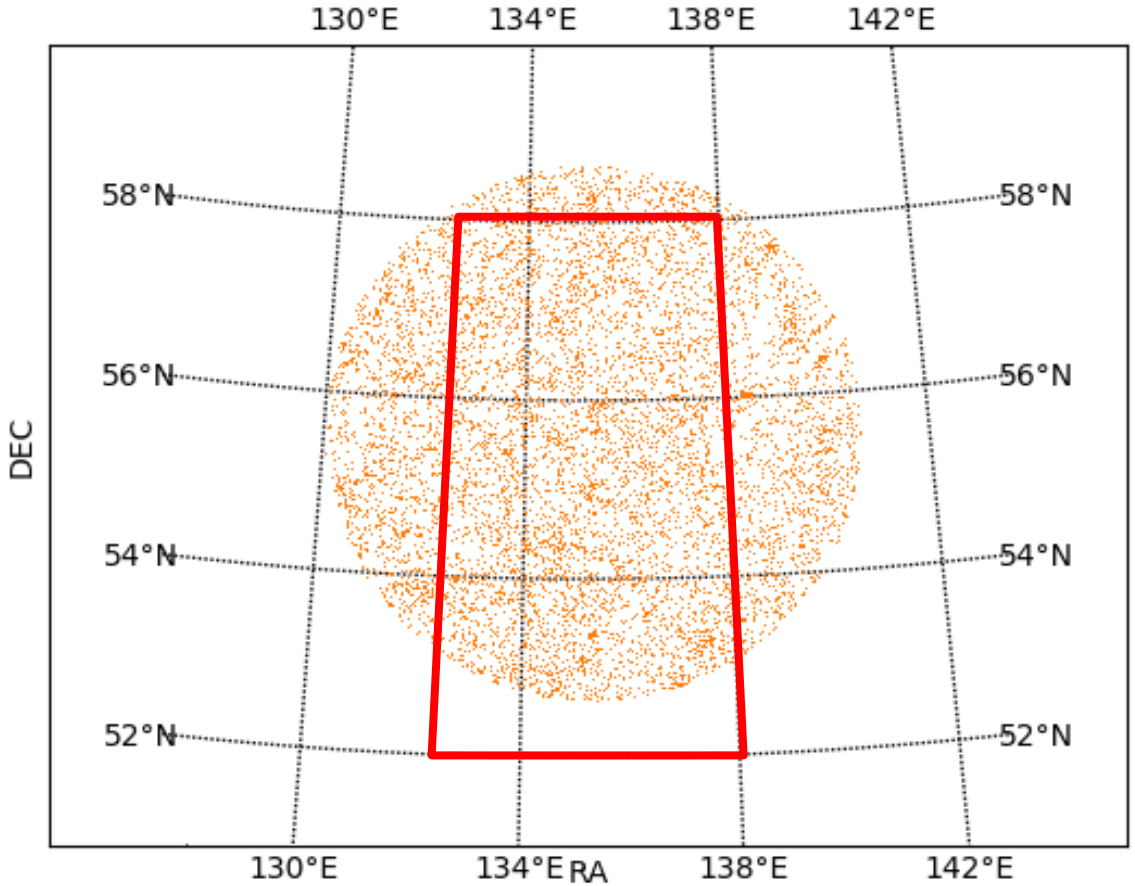
```
ra_n = ra[ra!=-9999]
dec_n = dec[ra!=-9999]
z_n = z[ra!=-9999]
```

1.3: 视场绘制

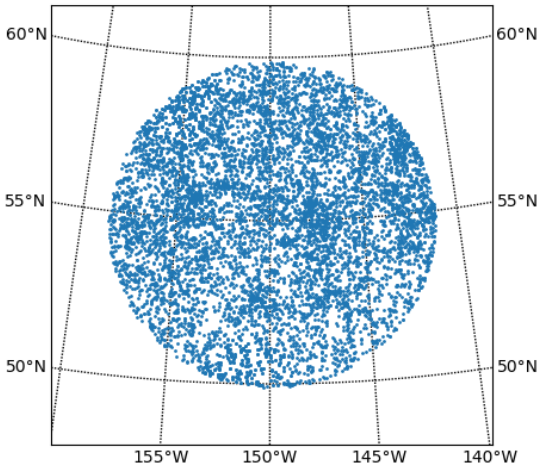
墨卡托投影（等角）



方位角等距投影



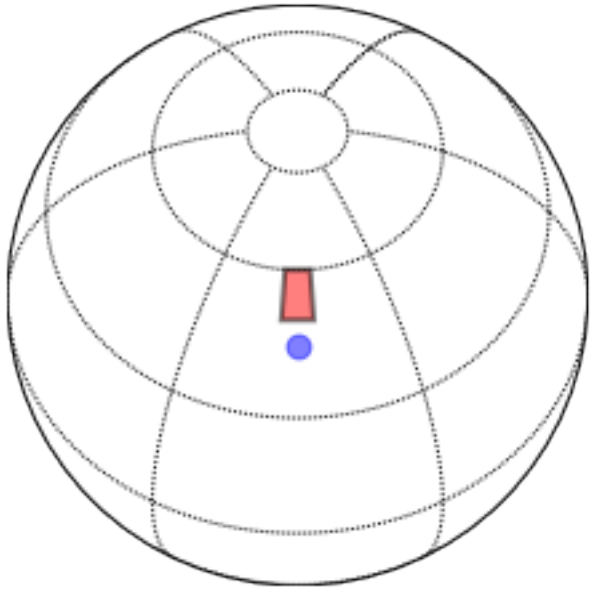
Basemap




```
from mpl_toolkits.basemap import Basemap
width = 1500000    #  in meters
m = Basemap(width=width, height=width,
            projection='aeqd', lat_0=centroid[0],lon_0=centroid[1])
m.drawparallels(np.arange(40,70,5),labels=[1,1,0,0])  # left, right
m.drawmeridians(np.arange(200,230,5),labels=[0,0,0,1])  # up, bottom
xpt, ypt = m(ra_n[ind], dec_n[ind])
m.plot(xpt,ypt, '.', ms=2)
plt.savefig("sky_real.png")
```

```
import os
os.environ['PROJ_LIB'] = '/home/gerry/prog/anaconda3/share/proj'
```

天区高亮（圆形）



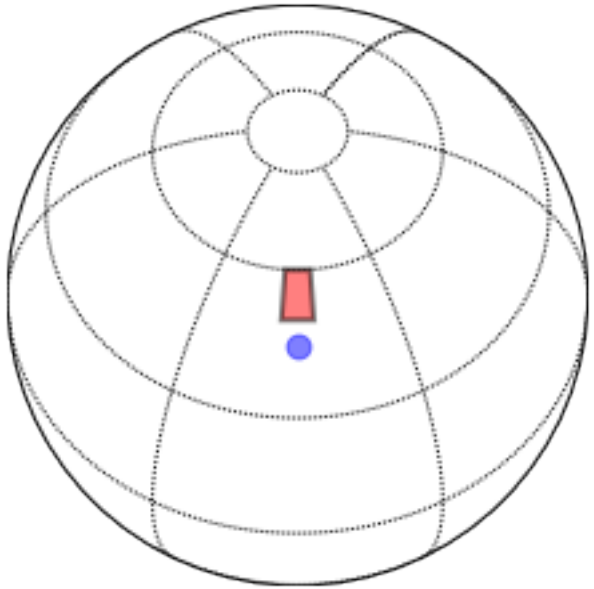
```
import matplotlib.pyplot as plt
from mpl_toolkits.basemap import Basemap

fig, ax = plt.subplots()
m = Basemap(projection='ortho', lon_0 = 210, lat_0 = 55)

m.drawparallels(np.arange(-90.,120.,30.))
m.drawmeridians(np.arange(0.,420.,60.))

x,y = m(210,45)
m.scatter(x, y, s=25*np.pi, c='b', alpha=0.5)
```

天区高亮（矩形）



- `from matplotlib.path import Path`
- `import matplotlib.patches as patches`

- `verts = [m(205,50), m(215,50), m(215,60), m(205,60),m(205,50)]`
- `codes = [Path.MOVETO, Path.LINETO, Path.LINETO,`
  `Path.LINETO, Path.CLOSEPOLY]`

- `path = Path(verts, codes)`
- `patch = patches.PathPatch(path, facecolor='r', lw=2, alpha=0.5)`
- `ax.add_patch(patch)`

1.4 FITS表格创建保存

```
# L: Logical (Boolean)
# B: Unsigned Byte
# I: 16-bit Integer
# J: 32-bit Integer
# K: 64-bit Integer
# E: Single-precision Floating Point
# D: Double-precision Floating Point
# C: Single-precision Complex
# M: Double-precision Complex
# A: Character
```

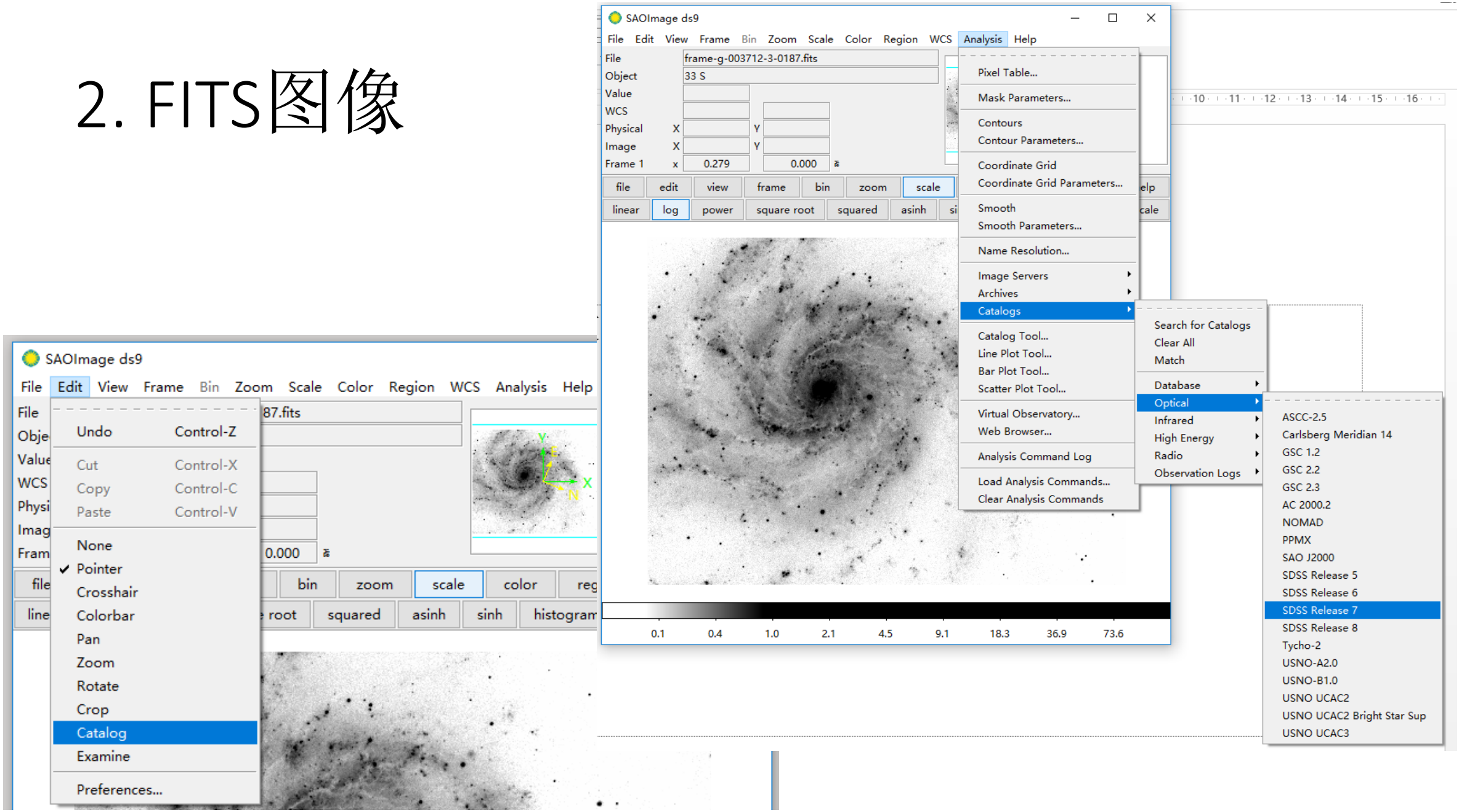
```
# 创建 HDU 对象
cols_ = []
cols_.append(fits.Column(name='RA', format='E', array=ra_n[ind]))
cols_.append(fits.Column(name='DEC', format='E', array=dec_n[ind]))
cols_.append(fits.Column(name='z', format='E', array=dec_n[ind]))
cols = fits.ColDefs(cols_)

hdu = fits.BinTableHDU.from_columns(cols)
```

```
# 创建 PrimaryHDU 对象
hdr = fits.Header()
hdr['OBSERVER'] = 'SDSS'
hdr['COMMENT'] = "SDSS galaxy around M101."
primary_hdu = fits.PrimaryHDU(header=hdr)

#结合PrimaryHDU和HDU对象，并且写入FITS文件
hdul = fits.HDUList([primary_hdu, hdu])
hdul.writeto('M101_r5.fits')
```

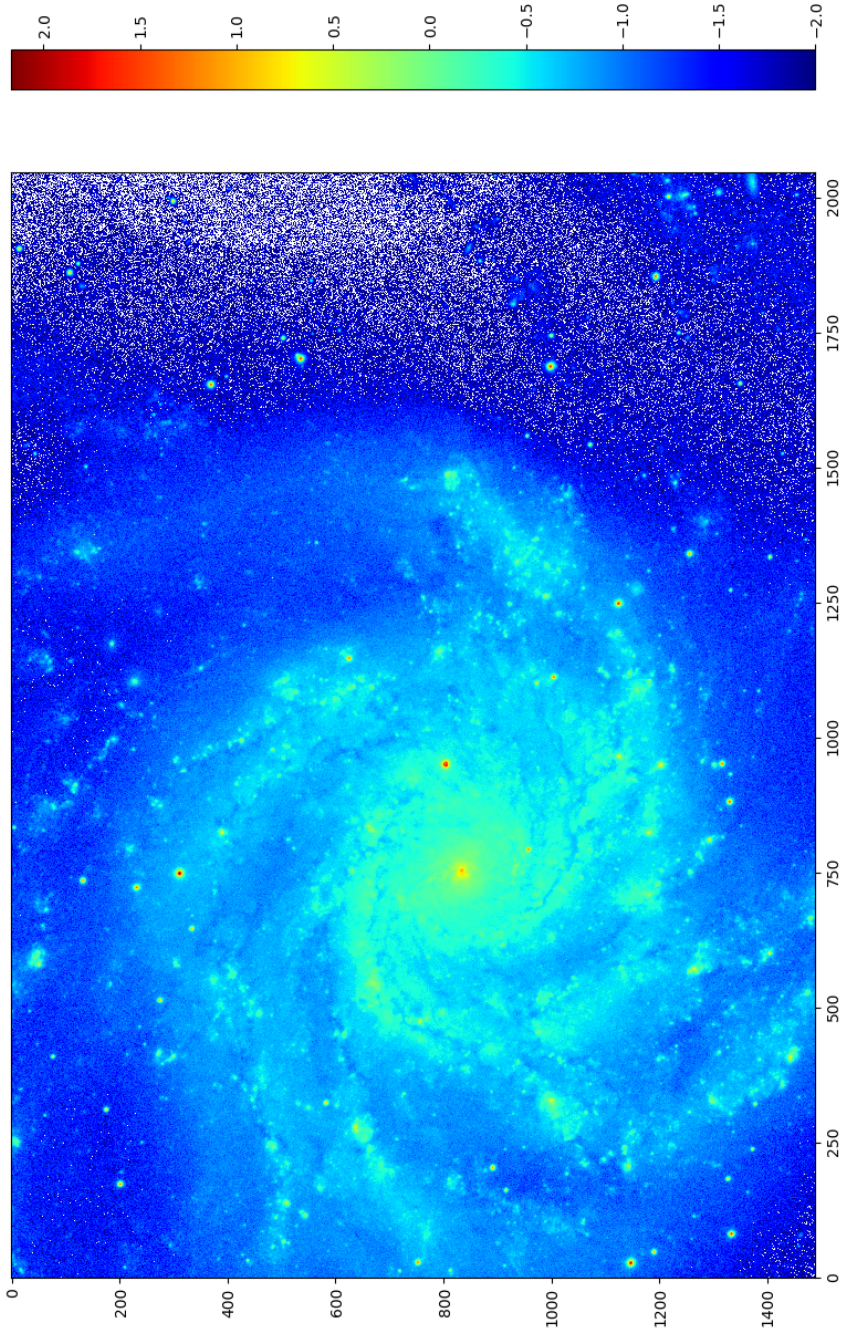
2. FITS图像


SAOImage ds9
File Edit View Frame Bin Zoom Scale Color Region WCS Analysis Help
Undo Control-Z
Cut Control-X
Copy Control-C
Paste Control-V
None
Pointer
Crosshair
Colorbar
Pan
Zoom
Rotate
Crop
Catalog
Examine
Preferences...
SAOImage ds9
File Edit View Frame Bin Zoom Scale Color Region WCS Analysis Help
File frame-g-003712-3-0187.fits
Object 33 S
Value
WCS
Physical X Y
Image X Y
Frame 1 x 0.279 0.000
file edit view frame bin zoom scale
linear log power square root squared asinh
Pixel Table...
Mask Parameters...
Contours
Contour Parameters...
Coordinate Grid
Coordinate Grid Parameters...
Smooth
Smooth Parameters...
Name Resolution...
Image Servers
Archives
Catalogs
Catalog Tool...
Line Plot Tool...
Bar Plot Tool...
Scatter Plot Tool...
Virtual Observatory...
Web Browser...
Analysis Command Log
Load Analysis Commands...
Clear Analysis Commands
Search for Catalogs
Clear All
Match
Database
Optical
Infrared
High Energy
Radio
Observation Logs
ASCC-2.5
Carlsberg Meridian 14
GSC 1.2
GSC 2.2
GSC 2.3
AC 2000.2
NOMAD
PPMX
SAO J2000
SDSS Release 5
SDSS Release 6
SDSS Release 7
SDSS Release 8
Tycho-2
USNO-A2.0
USNO-B1.0
USNO UCAC2
USNO UCAC2 Bright Star Sup
USNO UCAC3
0.1 0.4 1.0 2.1 4.5 9.1 18.3 36.9 73.6

FITS 图像读取

- `from astropy.io import fits`
- `hdu_f=fits.open("frame-g-003712-3-0187.fits")`
- `print(hdu_f.info())`
- `print(hdu_f[0].header)`
- `print(hdu_f[0].header['BITPIX'])`
- `print(hdu_f[0].header.comments['BITPIX'])`
- `print(hdu_f[0].header['NAXIS2'])`
- `print(len(hdu_f[0].data))`

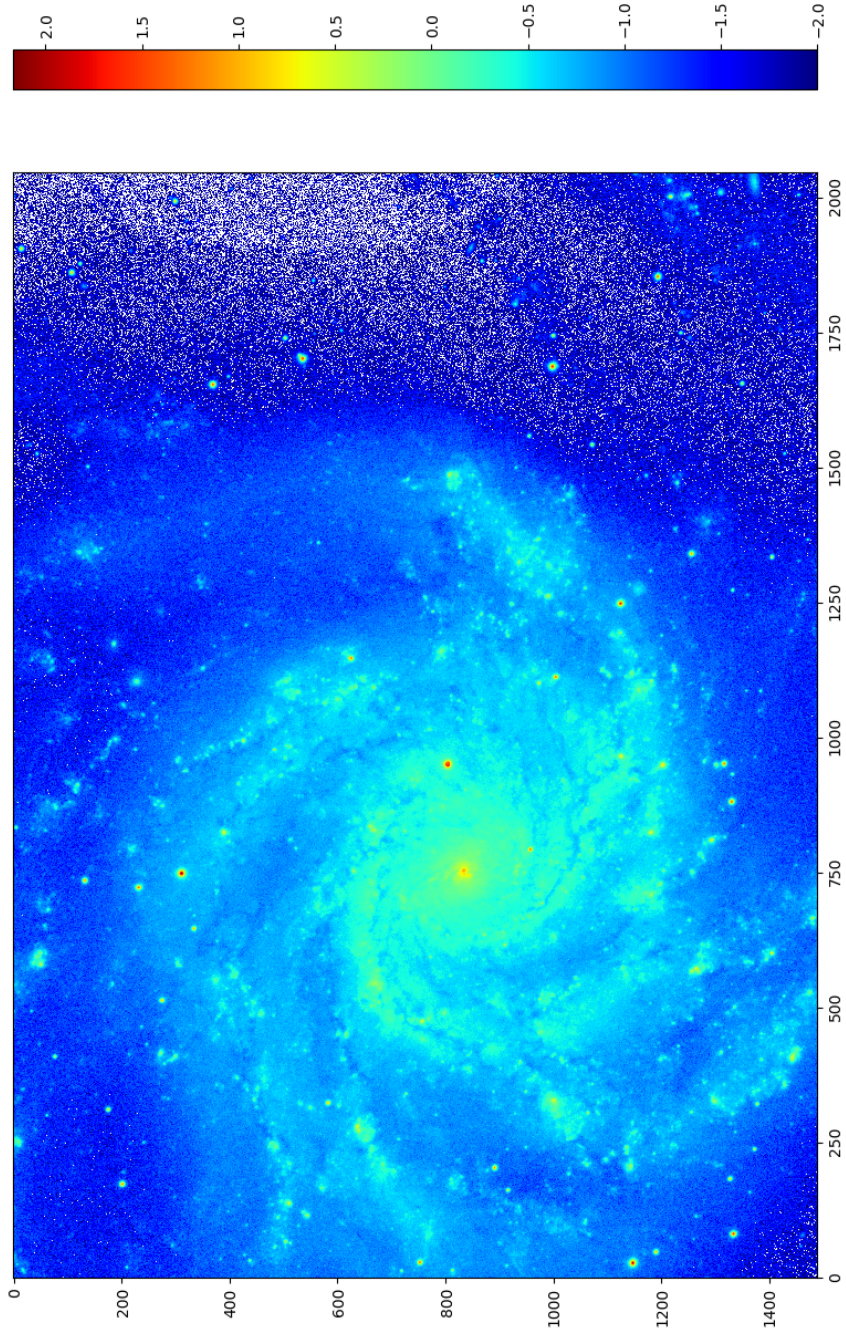
- `import pylab as plt`
- `plt.imshow(hdu_f[0].data)`
- `plt.colorbar()`
- `plt.show()`



图像检查

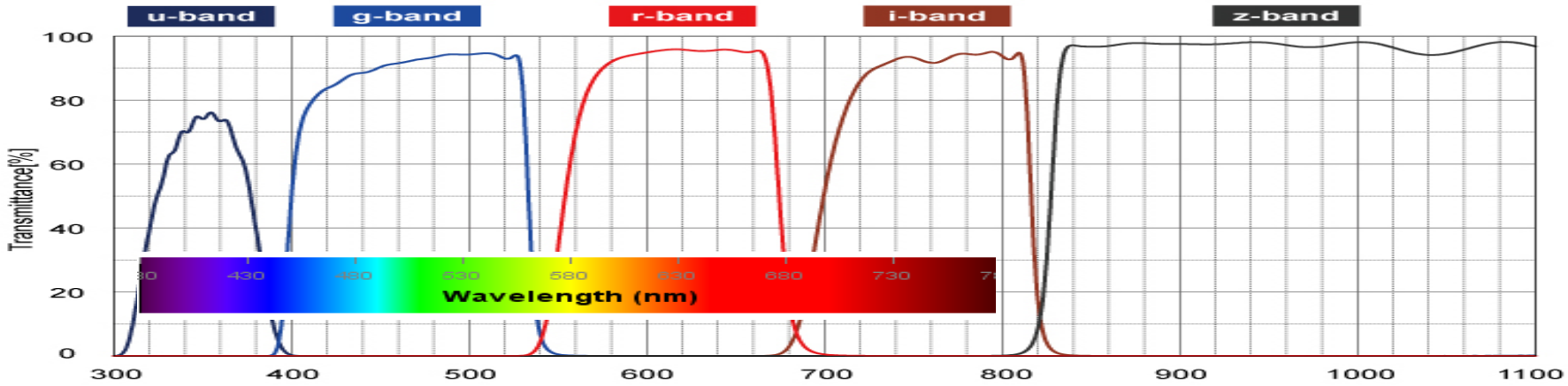
- `import pylab as plt`
- `flux = hdu_f[0].data.flatten()`
- `print(np.max(flux),np.min(flux))`
- `np.histogram(flux)`
- `flux0 = flux[flux>0]`
- `plt.hist(np.log10(flux0),100)`
- `plt.show`

- `img=hdu_f[0].data`
- `img[img<1e-6]=1e-6`
- `plt.imshow(np.log(img),cmap="jet")`
- `plt.show()`



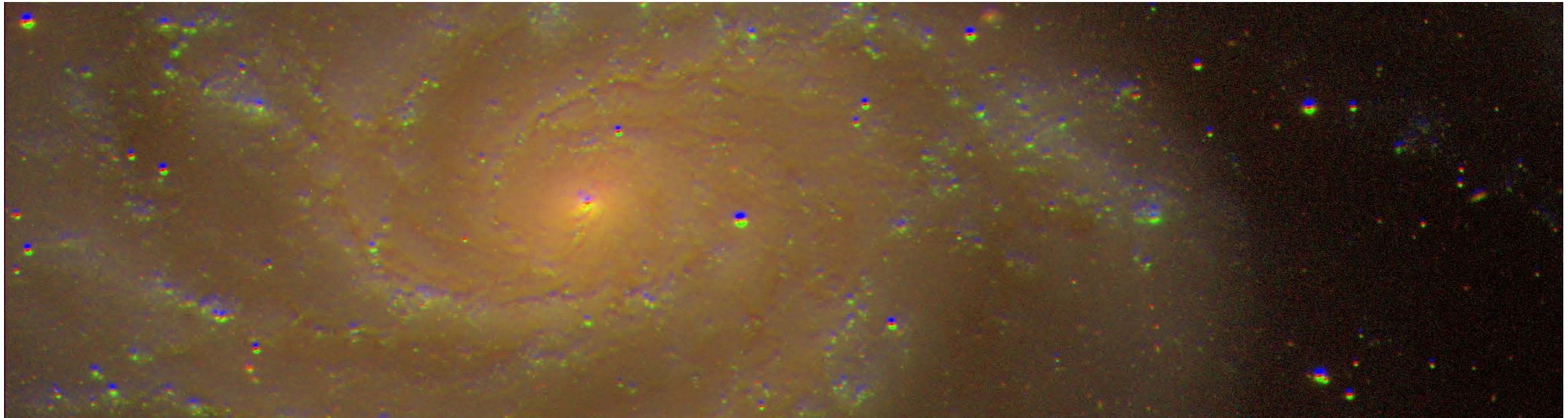
FITS 三色图像合成

```
import matplotlib.pyplot as plt
from astropy.io import fits
from astropy.visualization import make_lupton_rgb
im_r = fits.open('frame-i-003712-3-0187.fits')[0].data
im_g = fits.open('frame-r-003712-3-0187.fits')[0].data
im_b = fits.open('frame-g-003712-3-0187.fits')[0].data
rgb_default = make_lupton_rgb(im_r, im_g, im_b, Q=10, stretch=0.3,
                              filename="sdss_rgb.png")
plt.imshow(rgb_default, origin='lower')
```



文件头检查

```
for band in "irg":
    fname = 'frame-'+band+'-003712-3-0187.fits'
    data = fits.open(fname)
    print(fname)
    print(data[0].header['NAXIS1'],data[0].header['NAXIS2'])
    print(data[0].header['RA'],data[0].header['DEC'])
```



FITS图像切割

```
from astropy.nddata import Cutout2D
from astropy import wcs


n_cent = [210.7194019, 54.4090593]
imfile = "frame-i-003712-3-0187"
hdu_f = fits.open(imfile+".fits")
im = hdu_f[0].data
w = wcs.WCS(hdu_f[0].header)
center = SkyCoord(n_cent[0]*u.deg, n_cent[1]*u.deg, frame='fk5')
# then make an array cutout
co = Cutout2D(im, center, size=1000, wcs=w)
```

FITS图像保存

- plt.imshow(co.data,cmap="jet")
- print(co.data.shape)
- plt.show()
- # create a new FITS HDU

- hdu = fits.PrimaryHDU(data=co.data, header=co.wcs.to_header())

- # write to disk
- hdu.writeto(imfile+'_crop.fits',clobber=True)

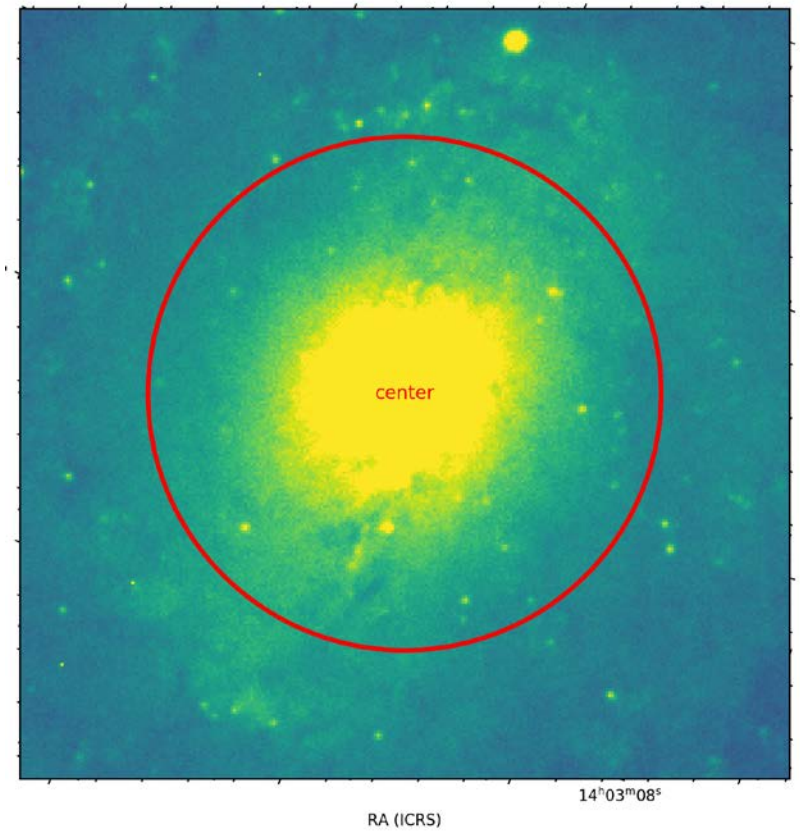
APLpy多波段合成

- pip3 install montage-wrapper / MontagePy

```
import aplpy
aplpy.make_rgb_cube(['i.fits', 'r.fits', 'g.fits'],
                    'M101_cube.fits')
aplpy.make_rgb_image('M101_cube.fits', 'sdss_rgb.png')
```

2.2 APLpy作图

- pip install aplpy
- 依赖于Numpy， Matplotlib，Astropy，部分功能还需要pyregion， PyAVM， PIL等几个模块的支持




```
import aplpy
import pylab as plt
fig = aplpy.FITSFigure('i.fits')
fig.show_colorscale()
plt.show()
```

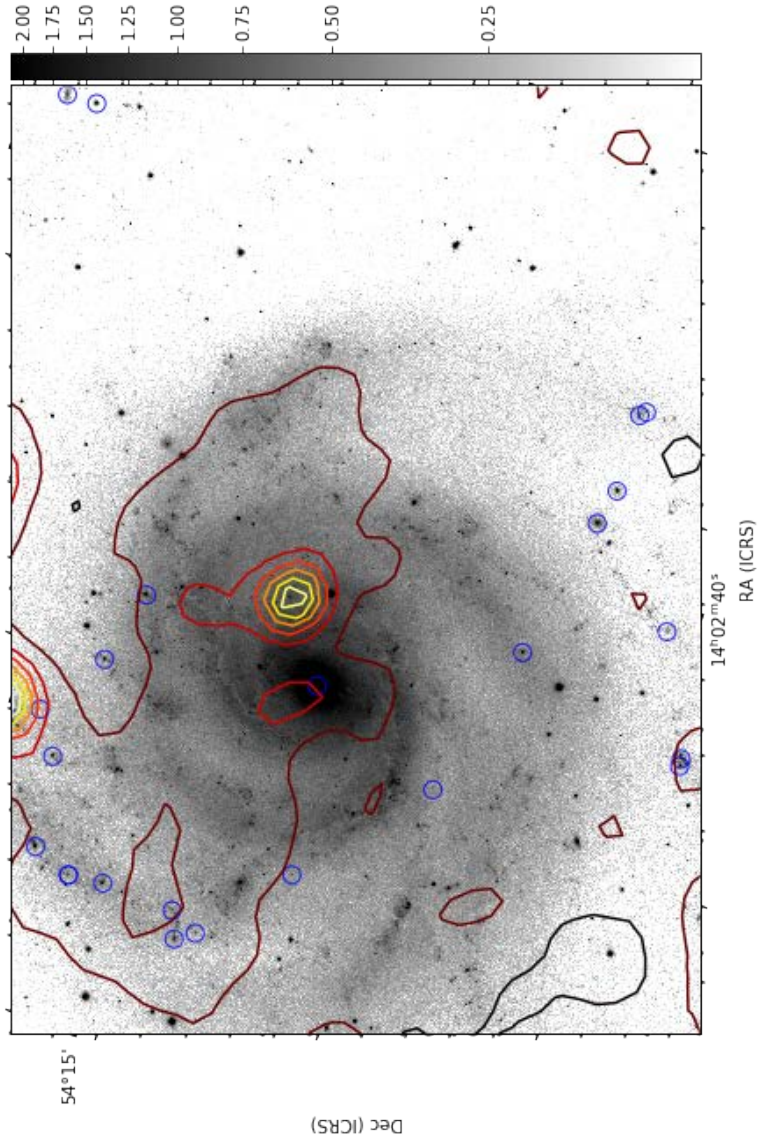
```
fig.recenter(210.802, 54.349, width=0.03, height=0.03)
fig.add_label(210.802, 54.349, 'center',color='r',size=12)
fig.show_circles(210.802, 54.349,0.01,color='r',lw=3)
```

多波段合成

```
fig = aplpy.FITSFigure('frame-i-003712-3-0187.fits')
fig.show_colorscale(cmap="gray_r", vmin=1e-1, stretch="log")
fig.add_colorbar()

fig.show_contour('M101_NVSS.fits', levels=8, cmap="hot")
#fig.show_regions('m101.reg')

hdu_t = fits.open('table_m101_r5.fits')
ra = hdu_t[1].data['ra']
dec = hdu_t[1].data.field('dec')
fig.show_markers(ra, dec, layer='marker_set_1', edgecolor='r',
                 facecolor='none', marker='o', s=100, alpha=0.8)
plt.show()
```



3 练习：FITS光谱

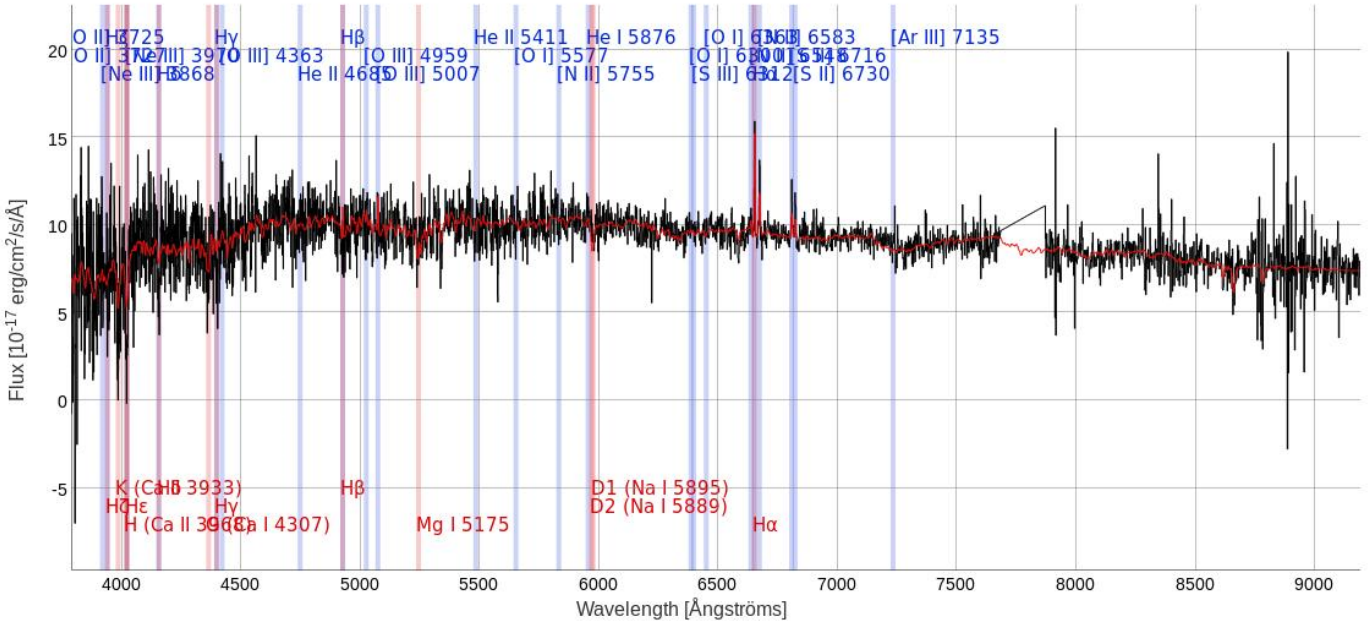
- 读取并绘制下列文件的光谱
- spec-1939-53389-0138.fits
- apStar-r8-2M09273449+2847577.fits

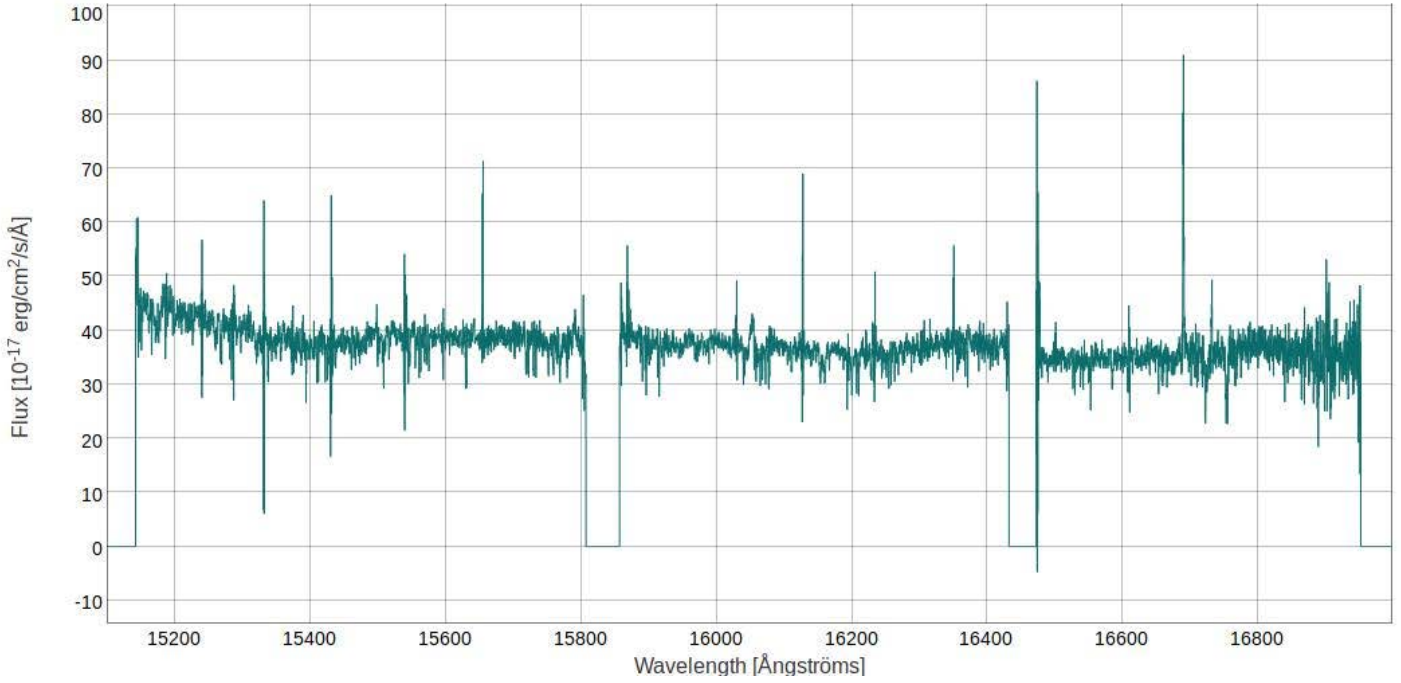
- `hdu=fits.open("spec-1939-53389-0138.fits")`
- `hdu.info()`

- `Filename: spec-1939-53389-0138.fits`
- `No.  Name  Ver  Type  Cards  Dimensions  Format`
- `0  PRIMARY  1 PrimaryHDU  139  ()`
- `1  COADD  1 BinTableHDU  26  3847R x 8C  [E, E, E, J, J, E, E, E]`
- `2 SPECOBJ 1 BinTableHDU 262 1R x 126C [6A, 4A, 16A, 23A, 16A, 8A, E, E, E, J, E, E, J, B, B, B, B, B, B, J, 22A, 19A, 19A, 22A, 19A, I, 3A, 3A, 1A, J, D, D, D, E, E, 19A, 8A, J, J, J, J, K, K, J, J, J, J, J, J, K, K, K, K, I, J, J, J, J, 5J, D, D, 6A, 21A, E, E, E, J, E, 24A, 10J, J, 10E, E, E, E, E, E, E, J, E, E, E, J, E, 5E, E, 10E, 10E, 10E, 5E, 5E, 5E, 5E, 5E, J, J, E, E, E, E, E, E, 25A, 21A, 10A, E, E, E, E, E, E, E, E, J, E, E, J, 1A, 1A, E, E, J, J, 1A, 5E, 5E]`
- `3  SPZLINE  1 BinTableHDU  48  29R x 19C  [J, J, J, 13A, D, E, E, E, E, E, E, E, E, E, E, J, J, E, E]`

```
hdu2=fits.open("apStar-r8-2M09273449+2847577.fits")
hdu2[0].header()
hdu2.info()
Filename: apStar-r8-2M09273449+2847577.fits
No.    Name      Ver    Type      Cards   Dimensions   Format
  0  PRIMARY       1 PrimaryHDU     538   ()
  1                1 ImageHDU        13   (8575, 25)   float32
  2                1 ImageHDU        13   (8575, 25)   float32
  3                1 ImageHDU        13   (8575, 25)   int16
  4                1 ImageHDU        13   (8575, 25)   float32
  5                1 ImageHDU        13   (8575, 25)   float32
  6                1 ImageHDU        13   (8575, 25)   float32
  7                1 ImageHDU        13   (8575, 25)   float32
  8                1 ImageHDU        29   (26,)   float64
  9                1 BinTableHDU     90   1R x 33C   [690A, 23I, 92A, 23J, 23I,
23D, 23E, 23E, 23E, 23E, 2I, 2E, 2E, 2E, 2E, E, E, E, E, E, 23E, 23E, 23E, 23E,
23E, 23E, 23E, 10025E, 10025E, 401E, D, D, 401E]
```

- HISTORY APSTAR: HDU0 = Header only
- HISTORY APSTAR: All image extensions have:
- HISTORY APSTAR: row 1: combined spectrum with individual pixel weighting
- HISTORY APSTAR: row 2: combined spectrum with global weighting
- HISTORY APSTAR: row 3-nvisits+2: individual resampled visit spectra
- HISTORY APSTAR: unless nvisits=1, which only have a single row
- HISTORY APSTAR: All spectra shifted to rest (vacuum) wavelength scale
- HISTORY APSTAR: HDU1 - Flux (10^-17 ergs/s/cm^2/Ang)
- HISTORY APSTAR: HDU2 - Error (10^-17 ergs/s/cm^2/Ang)
- HISTORY APSTAR: HDU3 - Flag mask:
- HISTORY APSTAR: row 1: bitwise OR of all visits
- HISTORY APSTAR: row 2: bitwise AND of all visits
- HISTORY APSTAR: row 3-nvisits+2: individual visit masks
- HISTORY APSTAR: HDU4 - Sky (10^-17 ergs/s/cm^2/Ang)
- HISTORY APSTAR: HDU5 - Sky Error (10^-17 ergs/s/cm^2/Ang)
- HISTORY APSTAR: HDU6 - Telluric
- HISTORY APSTAR: HDU7 - Telluric Error
- HISTORY APSTAR: HDU8 - LSF coefficients
- HISTORY APSTAR: HDU9 - RV and CCF structure


Flux [10^-17 erg/cm^2/s/Å]
Wavelength [Ångströms]
[Ar III] 7135
He II 5411
He I 5876
[O I] 6363
[N II] 6583
[S II] 6716
[S II] 6730
[O III] 4959
[O III] 5007
[N II] 5755
[O I] 5577
He II 4685
[O III] 4363
Hβ
Hγ
K (Ca II 3933)
H (Ca II 3968)
Hε
Hδ
Hγ
Hβ
Mg I 5175
D1 (Na I 5895)
D2 (Na I 5889)
Hα


Flux [10^-17 erg/cm^2/s/Å]
Wavelength [Ångströms]

谢

谢

历元与参考架

- ICRS，国际天球参考系，以太阳系质心为原点
- FK5 动力学参考系，依据FK5星表确定的一个动力学坐标系，主平面是J2000的平赤道面，赤经起点是J2000的平春分点。