Lec18-FITS-RyanSponzilli

October 24, 2024

1 ASTR 310 Lecture 18 - FITS files

1.0.1 Exercise 1: plotting an image from a FITS file

Using NumPy, Matplotlib, and Astropy, read in the file `lmc_ha_b20.fits` and do the following:
- Get the image data from the first HDU.

- Take the common logarithm of the image data.

[4 pts]

[55]:
```
import astropy.io.fits
import astropy.table
import numpy as np
import matplotlib.pyplot as plt
```

[19]:
```
primary_hdu = astropy.io.fits.open("lmc.ha.b20.fits")[0]
data = primary_hdu.data
log_data = np.log(data)
log_data
```

```
/var/folders/41/_gkgvhb94wd4156zplzr4cg00000gn/T/ipykernel_88936/3187382245.py:3
: RuntimeWarning: divide by zero encountered in log
  log_data = np.log(data)
```

[19]:
```
array([[-inf, -inf, -inf, …, -inf, -inf, -inf],
       [-inf, -inf, -inf, …, -inf, -inf, -inf],
       [-inf, -inf, -inf, …, -inf, -inf, -inf],
       …,
       [-inf, -inf, -inf, …, -inf, -inf, -inf],
       [-inf, -inf, -inf, …, -inf, -inf, -inf],
       [-inf, -inf, -inf, …, -inf, -inf, -inf]], dtype=float32)
```

- Get the first HDU's keywords `NAXIS1`, `NAXIS2`, `CDELT1`, and `CDELT2` (the latter two are in degrees/pixel). Use this information to compute the extent of the image in arcseconds. (Use the abs value of `CDELT1`… RA goes backwards.) [4 pts]

[34]:
```
naxis1 = primary_hdu.header["NAXIS1"]
naxis2 = primary_hdu.header["NAXIS2"]
cdelt1 = primary_hdu.header["CDELT1"]
```

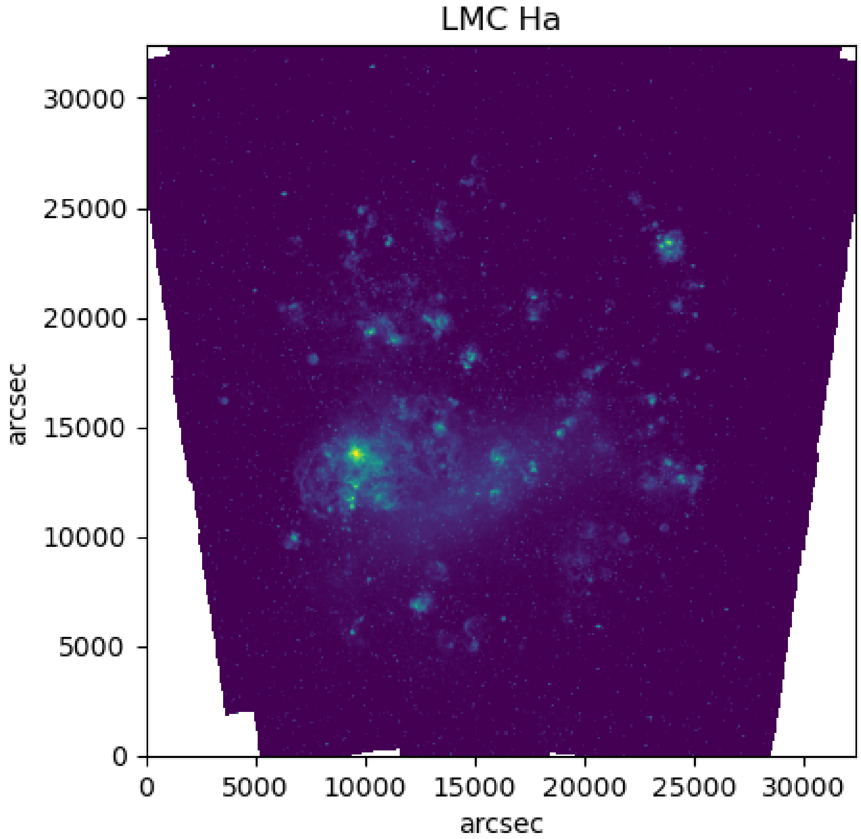
```
cdelt2 = primary_hdu.header["CDELT2"]

axis1_arcseconds = naxis1 * np.abs(cdelt1) * 3600
axis2_arcseconds = naxis2 * cdelt2 * 3600
print(axis1_arcseconds)
print(axis2_arcseconds)
```

```
32399.999517202225
32399.99951720242
```

- Plot the log image using the extent determined by the above keywords and placing the origin at lower left. You will want to adjust the color mapping using the vmin and vmax parameters. Don't forget axis labels and title. [4 pts]

```
[51]: plt.imshow(log_data, extent=[0, axis1_arcseconds, 0, axis2_arcseconds],␣
       ↪origin='lower', vmin=-0.5, vmax=4)
      plt.xlabel("arcsec")
      plt.ylabel("arcsec")
      plt.title("LMC Ha")
```

```
[51]: Text(0.5, 1.0, 'LMC Ha')
```



2 Exercise 2: modifying a FITS table

- The files `table1.fits` and `table2.fits` each contain an empty primary HDU and a binary table extension HDU. The table data are drawn from a sample of nearby clusters of galaxies. The file `table1.fits` contains: cluster name, redshift, RA (J2000) and dec (J2000).
  The file `table2.fits` contains: cluster name, 2-10 keV X-ray luminosity, X-ray temperature in keV.
- Using NumPy and Astropy, read the two files in, merge the tables into one new table, and write the results to a new file `newtable.fits`.

[6 pts]

[72]:

```
table1 = astropy.io.fits.open("table1.fits")[1].data
table2 = astropy.io.fits.open("table2.fits")[1].data

table1 = astropy.table.Table(table1)
table2 = astropy.table.Table(table2)

newtable = astropy.table.join(table1, table2, keys="name", join_type='outer')

newtable.write("newtable.fits", format='fits', overwrite=True)
```

- Double-check that the new table contains all the information you want: read the new table and print its column names. [2 pts]

[82]:

```
table = astropy.io.fits.open('newtable.fits')[1].data
print(table.columns)
print(table)
```

```
ColDefs(
    name = 'name'; format = '10A'
    name = 'redshift'; format = 'E'
    name = 'raj2000'; format = '15A'
    name = 'decj2000'; format = '15A'
    name = 'lx2-10'; format = 'D'
    name = 'tx'; format = 'E'
)
[('0336+09', 0.0349, '03h38m35.3s', '+09d57m55s', 2.49e+44,  3.1)
 ('3C, 129', 0.022 , '04h48m58.2s', '+45d02m01s', 2.02e+44,  5.6)
 ('A119', 0.044 , '00h56m21.4s', '-01d15m47s', 2.58e+44,  5.1)
 ('A1644', 0.0456, '12h57m14.8s', '-17d21m13s', 2.38e+44,  4.1)
 ('A1650', 0.0845, '12h58m46.2s', '-01d45m11s', 6.62e+44,  4.9)
 ('A1736', 0.046 , '13h26m52.1s', '-27d06m33s', 1.58e+44,  5.1)
 ('A2029', 0.0767, '15h10m58.7s', '+05d45m42s', 1.97e+45,  7.1)
 ('A2052', 0.0348, '15h16m45.5s', '+07d00m01s', 1.41e+44,  3.4)
 ('A2063', 0.0337, '15h23m01.8s', '+08d38m22s', 1.31e+44,  4.1)
 ('A2065', 0.0721, '15h22m42.6s', '+27d43m21s', 6.46e+44,  9.2)
 ('A2142', 0.0899, '15h58m16.1s', '+27d13m29s', 2.72e+45, 11. )
 ('A2147', 0.0356, '16h02m17.2s', '+15d53m43s', 1.82e+44,  4.4)
 ('A2244', 0.1024, '17h02m44.0s', '+34d02m48s', 8.09e+44, 10.3)
 ('A2255', 0.0798, '17h12m31.0s', '+64d05m33s', 4.87e+44,  8.4)
 ('A2319', 0.0564, '19h20m45s', '+43d57.7m', 1.70e+45,  9.2)
 ('A262', 0.0164, '01h52m50.4s', '+36d08m46s', 2.74e+43,  2.4)
 ('A3112', 0.0746, '03h17m56.3s', '-44d14m06s', 4.84e+44,  4.1)
 ('A3158', 0.059 , '03h43m00.6s', '-53d38m33s', 4.61e+44,  5.2)
 ('A3266', 0.0594, '04h31m09.7s', '-61d28m40s', 9.21e+44,  7.4)
 ('A3395', 0.0545, '06h27m31.1s', '-54d23m58s', 2.35e+44,  5.9)
 ('A3532', 0.0585, '12h57m19.2s', '-30d22m13s', 2.95e+44,  4.4)
 ('A3558', 0.0478, '13h27m54.8s', '-31d29m32s', 4.23e+44,  3.8)
 ('A3562', 0.0499, '13h33m31.8s', '-31d40m23s', 3.86e+44,  3.8)
 ('A3571', 0.0391, '13h47m29s', '-32d52.0m', 7.67e+44,  7.6)
 ('A399', 0.0715, '02h57m56.4s', '+13d00m59s', 7.76e+44,  6. )
 ('A4059', 0.0478, '23h56m40.7s', '-34d40m18s', 1.89e+44,  3.5)
 ('A478', 0.09  , '04h13m20.7s', '+10d28m35s', 2.41e+45,  6.8)
 ('A496', 0.032 , '04h33m37.1s', '-13d14m46s', 2.54e+44,  4.8)
 ('A576', 0.0381, '07h21m24.1s', '+55d44m20s', 1.09e+44,  3.8)
 ('A644', 0.0704, '08h17m26.4s', '-07d35m21s', 9.15e+44,  8.1)
 ('A85', 0.0518, '00h41m37.8s', '-09d20m33s', 7.54e+44,  6.6)
 ('AWM7', 0.0172, '02h54m32.2s', '+41d35m10s', 1.17e+44,  3.6)
 ('K44', 0.0283, '23h47m41.9s', '-28d08m19s', 8.91e+43,  3.3)
 ('MKW3s', 0.043 , '15h21m50.7s', '+07d42m18s', 1.45e+44,  2.8)
 ('Oph', 0.028 , '17h12m26s', '-23d22.6m', 1.52e+45,  9. )
 ('Tri, Aus', 0.051 , '16h38m20s', '-64d31.0m', 1.26e+45,  8. )]
```