# lab1_report

March 6, 2023

[ ]:
```
import numpy as np
from astropy.io import fits, ascii
from astropy.cosmology import FlatLambdaCDM
import pandas as pd
import matplotlib.pyplot as plt
```

[ ]:
```
quasar_hdu = fits.open('lab1-data.fits')[1]
quasar_data = quasar_hdu.data
```

[ ]:
```
RAs = quasar_data['RA']
DECs = quasar_data['DEC']
Zs = quasar_data['REDSHIFT']
```

[ ]:
```
indx = np.array(range(1000))
ra, dec, redshift = RAs, DECs, Zs
ascii.write([indx,ra,dec], 'lab1-target.txt', names=['name', 'ra', 'dec'],␣
 ↪overwrite=True)
```

[ ]:
```
results = pd.read_csv("result.csv")
u = results['modelMag_u'].values
g = results['modelMag_g'].values
r = results['modelMag_r'].values
i = results['modelMag_i'].values
z = results['modelMag_z'].values
located_indices = results['name']
RAs_located = RAs[located_indices]
DECs_located = DECs[located_indices]
Zs_located = Zs[located_indices]
```

[ ]:
```
cosmo = FlatLambdaCDM(H0=70, Om0=0.3)
lum_dist = cosmo.luminosity_distance(Zs_located)
```

[ ]:
```
dist_mod = 5 * np.log10(lum_dist.value * 1e6 / 10)
```

[ ]:
```
u_abs = u - dist_mod
g_abs = g - dist_mod
i_abs = i - dist_mod
```

```
r_abs = r - dist_mod
z_abs = z - dist_mod
```

[ ]:
```
from astropy.table import Table
```

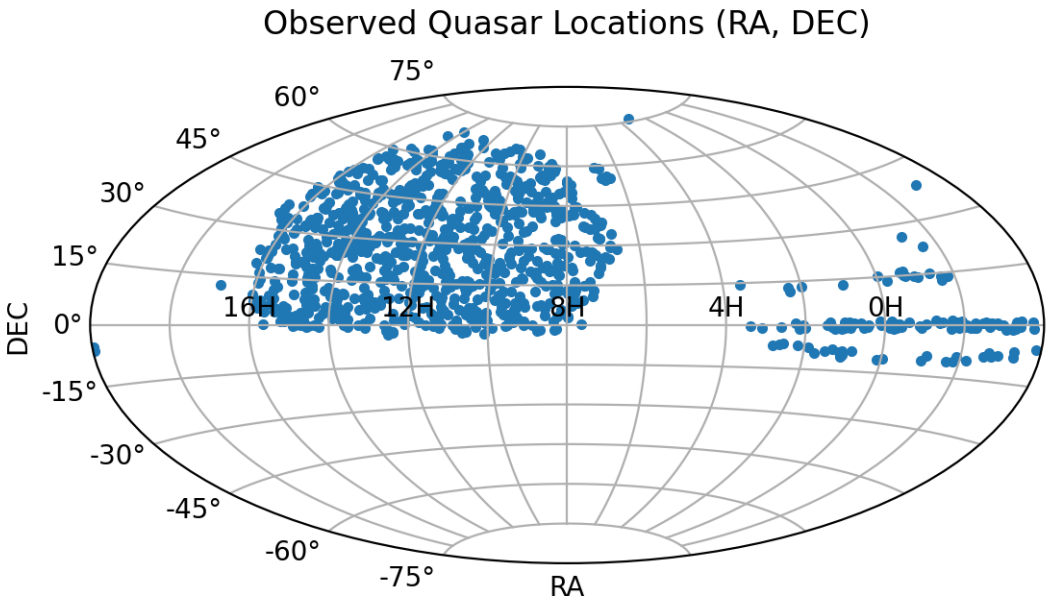
[ ]:
```
df = pd.DataFrame({"RA":RAs_located, "DEC":DECs_located, "Z":Zs_located, "m_u":
 ↪u,
                   'm_g':g, 'm_i':i, 'm_r':r, 'm_z':z, 'M_u':u_abs, 'M_g':
 ↪g_abs, 'M_i':i_abs,
                   'M_r':r_abs, 'M_z':z_abs, "lum_dist":lum_dist, 'dist_mod':
 ↪dist_mod})

t = Table.from_pandas(df)
t.write("lab1-complete-data.fits")
```

[ ]:
```
Table.read("lab1-complete-data.fits")
```

[ ]:
```
<Table length=996>
        RA                 DEC          …      lum_dist           dist_mod
     float64             float64        …      float64            float64
------------------ ------------------- … ------------------ ------------------
117.13700866699219  19.371322631835938 …  3918.174883925851 42.965419082016076
 130.7330322265625  22.575538635253906 …  22530.36209598235  46.76384085764787
   163.66357421875   5.592619895935059 … 3393.6960849312695  42.65336473712821
 123.6034164428711   47.45636749267578 …   9781.19161162182 44.951958833665685
132.97669982910156  2.2425200939178467 …  8920.151199784463  44.75186107943416
 175.0653533935547   30.11508560180664 …   33607.7242170061  47.63219552374884
205.48696899414062  0.5028740167617798 …  8575.948830670137  44.66641090545622
120.83368682861328   25.43400764465332 … 2678.2354600799713  42.13924377920931
231.78990173339844   22.30058479309082 … 10620.366737883976  45.13069756936253
230.36985778808594   21.69563865661621 … 10856.698995149622  45.17848898532715
               …                   … …                …                  …
132.24703979492188   6.256741046905518 …  9448.514351085081  44.87681763521919
194.49789428710938   47.04592514038086 … 1808.7465591172795  41.28688858972241
 149.5789031982422   28.36258316040039 …  31403.21524707842 47.484870579948506
227.40740966796875   54.60527038574219 … 25334.058556685486  47.01852385017246
218.33428955078125   38.14769744873047 … 4054.6179499864197  43.03974969343927
 189.8983612060547  2.5840859413146973 …  9231.969367906702 44.826471774026245
 260.9588928222656   54.38901901245117 … 15015.957327223061  45.88276512712283
140.87411499023438   6.823600769042969 … 12318.563110041718  45.45280026407161
223.72789001464844 -0.0873280018568039 … 10155.794086141323  45.03356943375401
 325.3304748535156  11.473767280578613 … 14484.802683746006  45.80456291746281
```

[ ]:
```
from astropy import units as u
from astropy.coordinates import SkyCoord
c = SkyCoord(ra=(RAs_located+120)*u.degree, dec=DECs_located*u.degree,␣
 ↪frame='icrs')
```

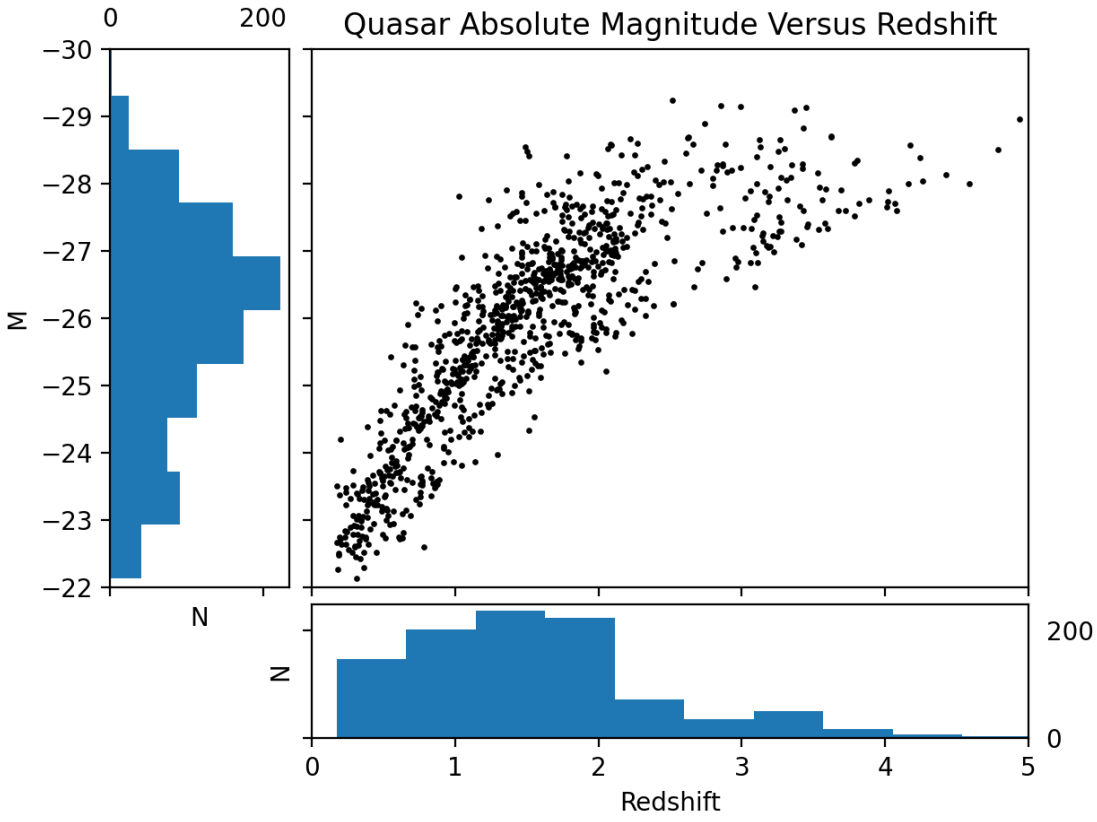
```
ra_rad = c.ra.wrap_at(180 * u.deg).radian
dec_rad = c.dec.radian
plt.figure(dpi=200)
plt.subplot(projection="aitoff")
ax = plt.gca()
ax.plot(ra_rad, dec_rad, 'o', markersize=3, zorder=0)
ax.set_xticklabels(['', '16H', '', '12H', '', '8H', '', '4H', '', '0H', ''],␣
 ↪zorder=10)
plt.title("Observed Quasar Locations (RA, DEC)", pad=20)
plt.xlabel("RA")
plt.ylabel("DEC")
plt.grid(True)
```




[ ]:
```
def scatter_hist(x, y, ax, ax_histx, ax_histy):
    # no labels
    ax_histx.tick_params(axis="x", labelbottom=False)
    ax_histy.tick_params(axis="y", labelleft=False)

    # the scatter plot:
    #ax.scatter(x, y)

    # now determine nice limits by hand:
    binwidth = 0.25
    xymax = max(np.max(np.abs(x)), np.max(np.abs(y)))
    lim = (int(xymax/binwidth) + 1) * binwidth
```

```
    bins = np.arange(-lim, lim + binwidth, binwidth)
    ax_histx.hist(x, bins=bins, orientation = 'horizontal')
    ax_histy.hist(y, bins=bins, orientation='vertical')


fig = plt.figure(dpi = 200)
gs = fig.add_gridspec(2, 2,  width_ratios=(1, 4), height_ratios=(4, 1),
                      left=0.1, right=0.9, bottom=0.1, top=0.9,
                      wspace=0.05, hspace=0.05)

ax = fig.add_subplot(gs[0, 1])
ax.set_ylim(-22, -30)
ax.set_xlim(0, 5)
ax.scatter(Zs_located, i_abs, color = 'black', s =2)
xticklabels = ax.get_xticklabels()
yticklabels = ax.get_yticklabels()
xticks = ax.get_xticks().copy()
yticks = ax.get_yticks().copy()

ax_histy = fig.add_subplot(gs[0, 0], sharey = ax)
ax_histy.hist(i_abs, orientation = 'horizontal')

ax_histy.tick_params(axis="x", labelbottom=False)
ax_histy.tick_params(axis="x", labeltop=True)

ax_histx = fig.add_subplot(gs[1, 1], sharex = ax)
ax_histx.hist(Zs_located)

ax_histx.tick_params(axis="y", labelright=True)
ax_histx.tick_params(axis="y", labelleft=False)



ax_histx.set_xticks(xticks)
ax_histx.set_xticklabels(xticklabels)

# ax_histy.set_yticks(yticks)
# ax_histy.set_yticklabels(yticklabels)
# ax.set_xticklabels([])
# ax.set_yticklabels([])
ax.tick_params(axis="y", labelright=False)
ax.tick_params(axis="y", labelleft=False)
ax.tick_params(axis="y", labelright=False)
ax.tick_params(axis="x", labelbottom=False)

ax_histx.set_xlabel("Redshift")
ax_histx.set_ylabel("N")
```

```
ax_histy.set_ylabel("M")
ax_histy.set_xlabel("N")

ax.set_title("Quasar Absolute Magnitude Versus Redshift")
plt.show()
```




[ ]: