Trombini_QUentin_Week9

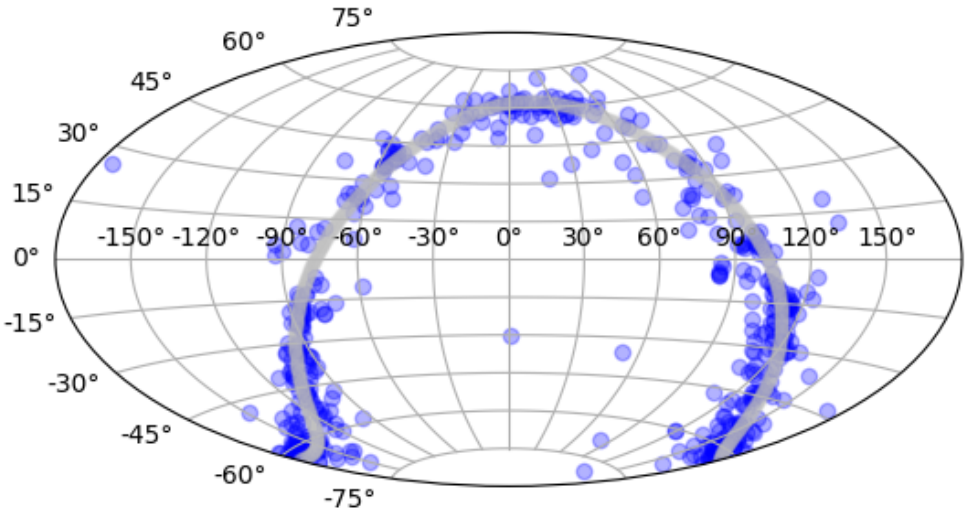
S09_01

S09_01_a

```
import numpy as np
import astropy.units as units
import astropy.coordinates as coord
import matplotlib.figure as fig
import matplotlib.backends.backend_agg as back

u_ha = units.hourangle
u_deg = units.deg

data_ra = np.array([])
data_dec = np.array([])
data_l = np.array([])
data_b = np.array([])

with open("./Astro/S09/dataE1.txt","r") as f:
    lines = f.readlines()

for line in lines:
    splitted = line.split()
    ha = float(splitted[-2])
    deg = float(splitted[-1])

    coordinate = coord.SkyCoord(ha,deg,unit=(u_ha,u_deg))

    ra_rad = coordinate.ra.radian
    dec_rad = coordinate.dec.radian

    l_rad = coordinate.galactic.l.radian
    b_rad = coordinate.galactic.b.radian

    if ra_rad > np.pi:
        ra_rad -= 2.0 * np.pi
    if l_rad > np.pi:
        l_rad -= 2.0 * np.pi

    data_ra = np.append(data_ra, ra_rad)
    data_dec = np.append(data_dec, dec_rad)
    data_l = np.append(data_l, l_rad)
    data_b = np.append(data_b, b_rad)

gal_lon = np.linspace(0.001 , 359.999 , 1000) * u_deg
gal_lat = np.zeros(1000) * u_deg
gal_coord = coord.Galactic(l = gal_lon, b = gal_lat)
gal_ra = gal_coord.transform_to(coord.ICRS).ra.wrap_at(180.0 * u_deg).radian
```

```
gal_dec = gal_coord.transform_to(coord.ICRS).dec.radian

fig = fig.Figure()
canvas = back.FigureCanvasAgg(fig)
ax = fig.add_subplot(111, projection = "aitoff")

ax.grid()

ax.plot(gal_ra, gal_dec, linestyle= "None" ,marker = "o", color = "silver", markersize=5, alpha =0.1)
ax.scatter(data_ra, data_dec, marker = "o", c = "blue", alpha =0.3)

fig.savefig("Astro/S09/E1.png")
```

S09_01_b



S09_01_c

In this code first we will read the file and extract all the data frome the right column

once we get this data we do the $coordinate = coord.SkyCoord(ha, deg, unit = (u\_ha, u\_deg))$ to get a sky coord object out of our data.

Then we will fill our arrays with the data extracted from this object.

we will use

```
gal_ra = gal_coord.transform_to(coord.ICRS).ra.wrap_at(180.0 * u_deg).radian
gal_dec = gal_coord.transform_to(coord.ICRS).dec.radian
```

to translate from classical coordinate to galactic coordinate.

We then plot every point, we can see that everything is on a belt

---

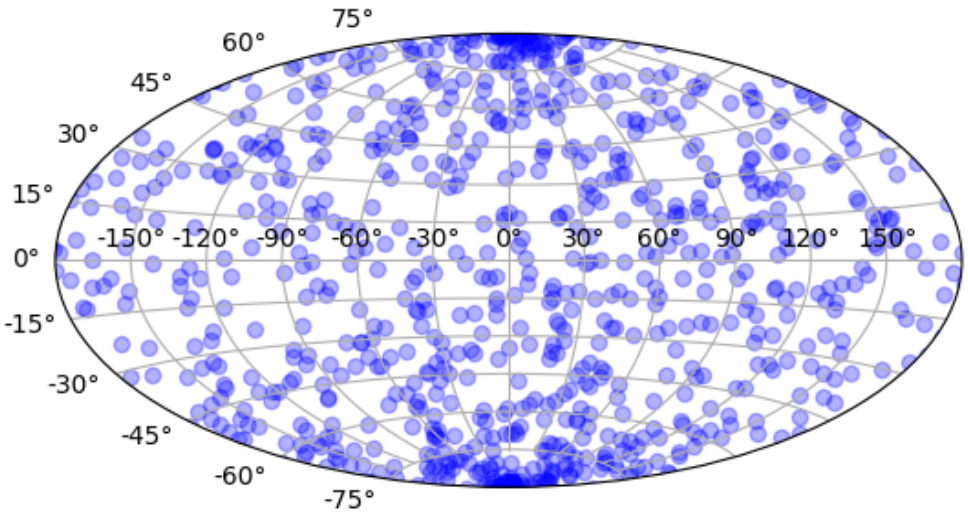
S09_02

S09_02_a

```
import numpy as np
import astropy.units as units
import astropy.coordinates as coord
import matplotlib.figure as fig
import matplotlib.backends.backend_agg as back

u_ha = units.hourangle
u_deg = units.deg

data_ra = np.array([])
data_dec = np.array([])
data_l = np.array([])
data_b = np.array([])

with open("./Astro/S09/dataQ2.txt","r") as f:
    lines = f.readlines()

for line in lines:
    if "#" not in line:
        splitted = line.split()
        try:
            ra_rad = float(splitted[8])
        except:
            continue
        dec_rad = float(splitted[10])

        if ra_rad > np.pi:
```

```
            ra_rad -= 2.0 * np.pi

        data_ra = np.append(data_ra, ra_rad)
        data_dec = np.append(data_dec, dec_rad)

gal_lon = np.linspace(0.001 , 359.999 , 1000) * u_deg
gal_lat = np.zeros(1000) * u_deg
gal_coord = coord.Galactic(l = gal_lon, b = gal_lat)
gal_ra = gal_coord.transform_to(coord.ICRS).ra.wrap_at(180.0 * u_deg).radian
gal_dec = gal_coord.transform_to(coord.ICRS).dec.radian

fig = fig.Figure()
canvas = back.FigureCanvasAgg(fig)
ax = fig.add_subplot(111, projection = "aitoff")

ax.grid()

ax.scatter(data_ra, data_dec, marker = "o", c = "blue", alpha =0.3)

fig.savefig("Astro/S09/E2.png")
```

S09_02_b



S09_02_c

This code is a lot like the one of the first part but instead of having data in hourangle and degree here we directly get it in ra and dec. This allow us to remove a lot a step in our code.

we can see a that there is a special concentration in both pole of the map.