

Spatial Selection using healpy



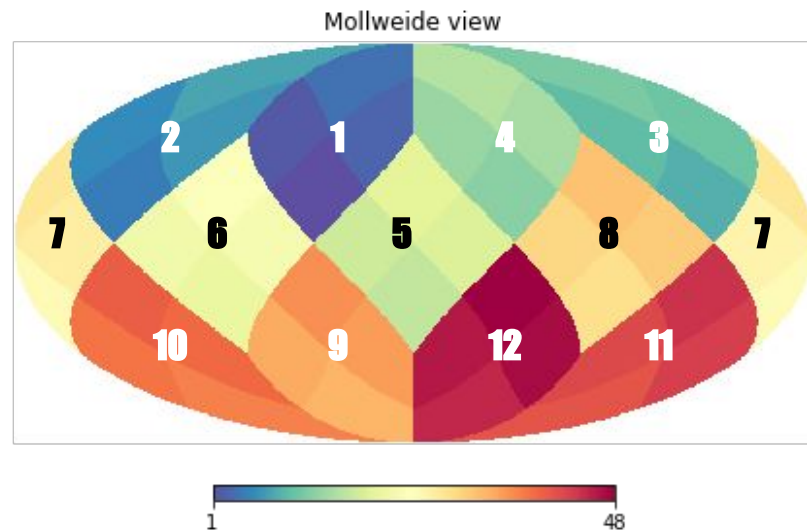
[Notebook](#)

Healpix

Healpix: **H**ierarchical **E**qual **A**rea iso**L**atitude **P**ixelization of a sphere

- Pixels distributed constant latitude
- 12 base pixels $N_{\text{pixels}} = 12 n_{\text{side}}^2$
- Areas of all pixels are equal:

Resolution = $41523 / N_{\text{pixels}}$ (deg²)



Visualize

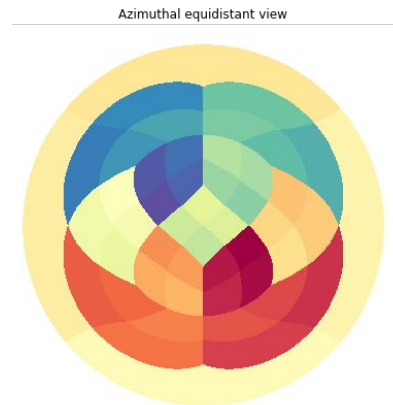
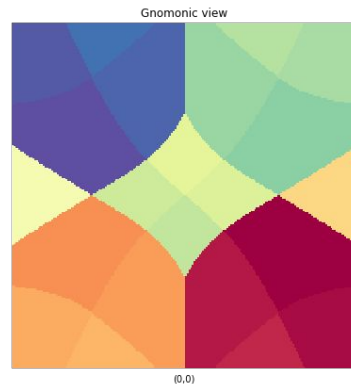
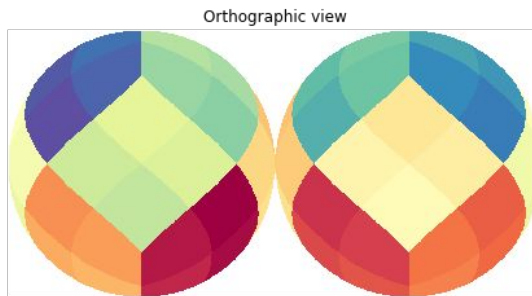
Create a map: 1-D array

```
len(map) = Npixels
```

support masked array

```
fill_value=hp.UNSEEN  
np.ma.masked_array(map, mask)
```

```
hp.mollview(map, fig, coord...)  
hp.gnomview(map, ...)  
hp.cartview(map, ...)  
hp.orthview(map, ...)  
hp.azeqview(map, ...)
```



Query polygon

define vertices

```
ra = [0, 60, 30, 330, 300]  
dec = [0, -15, -45, -45, -15]
```

convert to 3D xyz coordinates

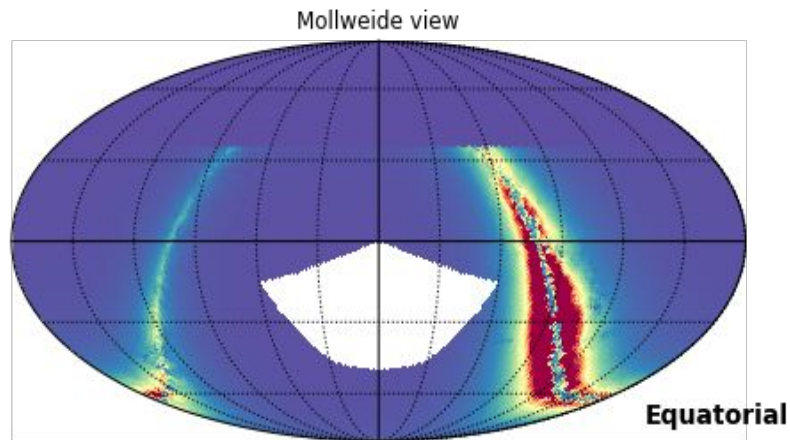
```
xyz = radec2xyz(ra, dec)
```

get pixels inside

```
ipix_poly = hp.query_polygon(NSIDE, xyz)
```

create a masked map

```
mask = np.isin(pixes, ipix_poly)
```



Query disc

define disk center

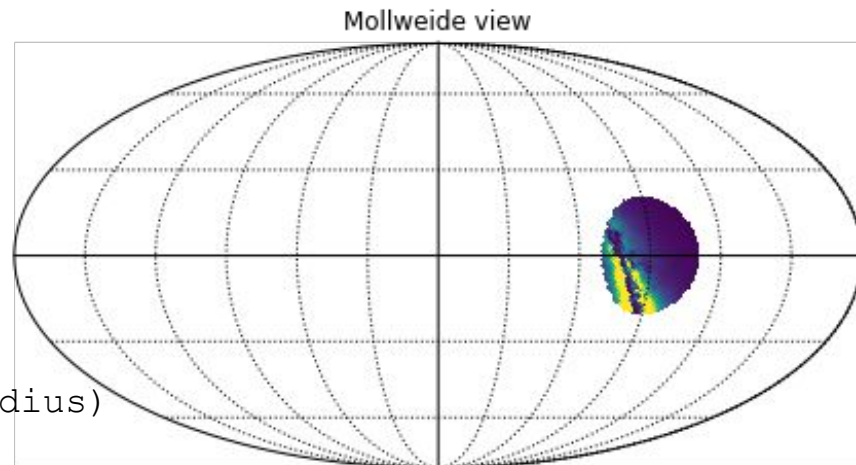
```
vec = radec2xyz(ra=270, dec=0)  
radius = np.deg2rad(20)
```

get pixels inside

```
ipix_disc = hp.query_disc(NSIDE, vec, radius)
```

create a masked map

```
mask = np.isin(pixes, ipix_disc)
```



Query ring

define two disks

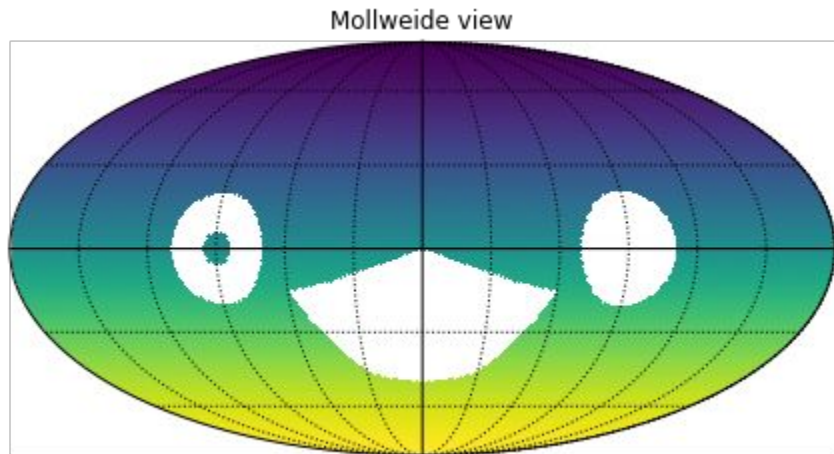
```
vec = radec2xyz(ra=90, dec=0)
radius1 = np.deg2rad(6)
radius2 = np.deg2rad(20)
disc1 = hp.query_disc(NSIDE, vec, radius1)
disc2 = hp.query_disc(NSIDE, center,
radius2)
```

get pixels inside ring

```
ipix_ring = np.setdiff1d(disc2, disc1)
```

create a masked map

```
mask = np.isin(pixes, ipix_ring)
```



Galactic avoidance region

define the shape of galactic plane

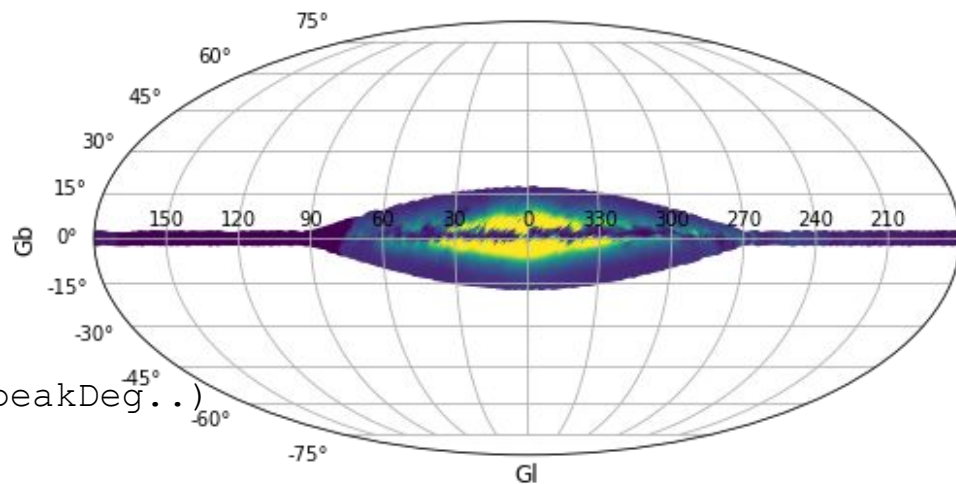
```
peakDeg = 15.  
taperDeg = 90.  
constWidth = .5
```

get pixels inside

```
ipix_galaxy = get_avoidance_pixel(peakDeg..)
```

create a masked map

```
mask = np.isin(pixes, ipix_galaxy)
```



Irregular shape

Select pixels by Interactive tools

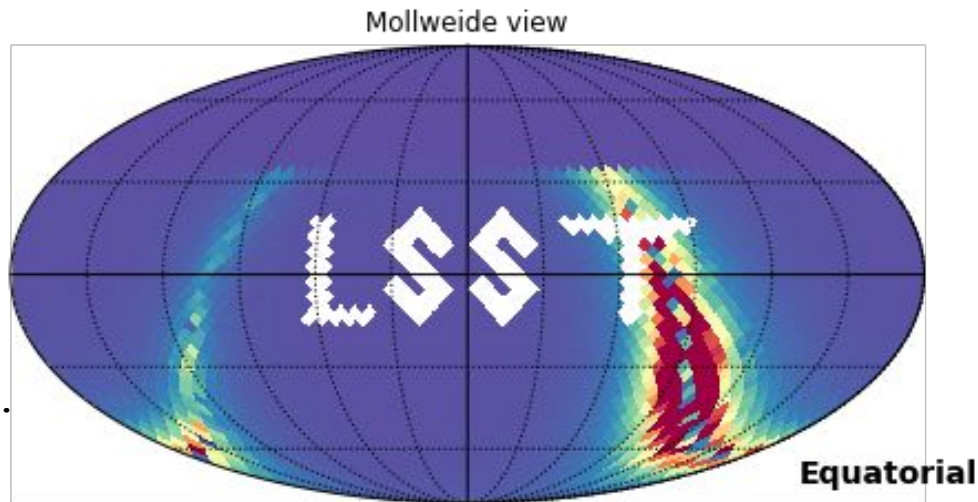
Bokeh, plotly, ... [Link](#)

```
# get pixId
```

```
ipix_lsst = [1067,1066,1130,1194, .
```

```
# create a mask
```

```
mask = np.isin(pixes, ipix_lsst)
```



Summary

Steps: Define the shape → get pixels inside → create a masked map

Query polygon: `hp.query_poly(nside, vec, ...)`

Query disc: `hp.query_disc(nside, vec, ...)`

Galactic avoidance region: defined by size or by star density

Irregular shape: use interactive tools

Link to [Notebook\(https://git.io/JJ6Nq\)](https://git.io/JJ6Nq)