



# Create Maps, Cube, ModelCube, or Image Object

```
from marvin.tools import Maps
maps = Maps('8485-1901')

from marvin.tools import Cube
cube = Cube('1-209232')

from marvin.tools import ModelCube
modelcube = ModelCube('manga-8485-1901-LOGCUBE-HYB10-GAU-MILESHC.fits.gz')

from marvin.tools import Image
image = Image('8553-12702')
Specify plateifu, mangaid, or path to file.
```

## MaNGA Data Analysis with Marvin

[Docs](#)

[Cite Marvin](#)

# Downloading

```
from marvin import config
config.download = True
Global switch to download Maps, Cube, and ModelCube files.

cube = Cube('8485-1901', download=True)
Download single object (also works for Maps and ModelCube).

from marvin.utils.general import downloadList
galaxies = ['8485-1901', '7443-12701']
downloadList(galaxies, dltype='cube')
Batch download cubes, images, maps, or RSS files.
```

# Maps & Map

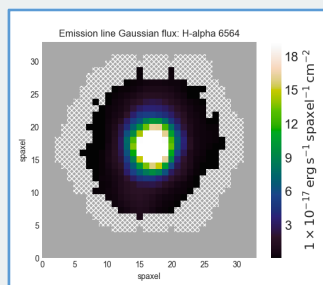
```
maps.datamodel
List all properties (+ channels) in a Maps.

ha = maps.emline_gflux_ha_6564
nii = maps['emline_gflux_nii_6585']
Get a Map with dotted or key syntax.

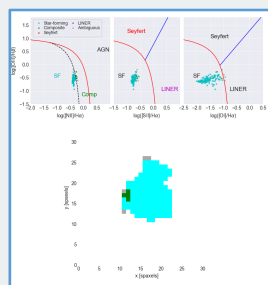
ha.value
ha.ivar
Get map values or inverse variances.

np.log10(nii) / ha**2
Do map arithmetic (+, -, *, /, or **) or logs.

maps.getCube()
maps.getModelCube()
Get Cube or ModelCube from a Maps.
```



ha.plot()



maps.get\_bpt()

# Cube & ModelCube

```
cube.flux.value
cube.flux.ivar
Flux and inverse variance of spectral cube.

modelcube.binned_flux.value
modelcube.binned_flux.ivar
Binned flux and inverse variance of spectra fit by DAP.

modelcube.full_fit.value
Get model fit.

cube.getMaps()
modelcube.getMaps()
modelcube.getCube()
Get a Maps or Cube.
```

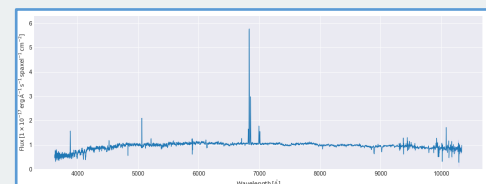
# Spaxel

```
maps[1, 2]
Get spaxel (y=1, x=2). Also works for Cube and ModelCube.

sp = maps.getSpaxel(y=1, x=2, xyorig='lower', cube=True,
                    modelcube=True)
Get spaxel (y=1, x=2) with model fits.

sp.flux.value
sp.flux.ivar
Spectrum flux and inverse variance arrays.

sp.full_fit.value
Get model fits.
```



sp.flux.plot()

# Maskbits

```
ha.pixmask.schema
ha.pixmask.get_mask('NOCOV')
ha.pixmask.values_to_labels()[1][2]
Flags for spaxel (y=1, x=2).

ha.target_flags
Show targeting masks.

Cube and ModelCube also have pixmask and target_flags.
```

# Image

```
images = Image.from_list(['8485-1901', '7443-12701'])
Download list of images.

image.show()
image.plot()
Show an image or plot with Matplotlib.
```



# Query & Results

```
from marvin.tools.query import Query
sf = 'nsa.z < 0.1 and sfr_lre > 0.5'
rp = ['stellar_sigma_lre']
q = Query(search_filter=sf,
          return_params=rp)

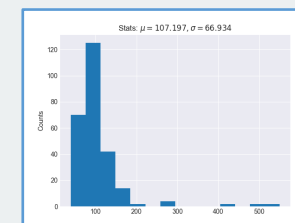
r = q.run()
Build and run query.

r.extendSet()
r.loop()
r.getAll()
Extend results set.

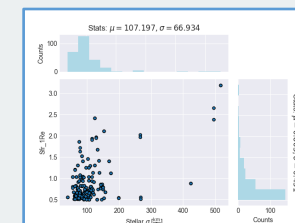
r.getNext()
r.getPrevious()
Cycle through results.

r.download()
Download query results.

r.toTable()
r.toDF()
r.convertToTool('maps')
Convert to astropy Table, pandas DataFrame, or Marvin objects.
```



r.hist('stellar\_sigma\_lre')



r.plot('stellar\_sigma\_lre', 'sfr\_lre')