SUPERMOCK SYNTHETIC SKY CATALOG

Based on:

- LastJourney Dark matter simulation products
- Semi-empirical modeling for dust, metallicity
- Heuristic modeling for galaxy-halo connection
- Stellar population synthesis

Contains

- RA-Dec coordinates, redshifts, velocities
- Simulation quantities such as star formation histories
- SEDs, Bandpasses from multiple surveys (SPHEREx, SDSS, COSMOS...) etc.

Base simulation: HACC Gravity-only Simulation LastJourney: [Paper]

Cosmology Parameters	Simulation Volume Parameters
$$\Omega_m = 0.30964$$	$$L = 3400 \; h^{-1}\mathrm{Mpc}$$
$$\Omega_b = 0.04897$$	$$N_P = 10752$$
$$\Omega_\Lambda = 0.69036$$	$$m_p = 2.7174e+09 \; h^{-1}M_\odot$$
$$\sigma_8 = 0.8102$$	$$z_\mathrm{start} = 200.0$$
	$$n_\mathrm{steps} = 500$$

In [ ]:
```
# import h5py
import numpy as np
import matplotlib.pylab as plt
import matplotlib.colors as mcolors
import pandas as pd
from mpl_toolkits.basemap import Basemap
# plt.style.use('dark_background')
from astropy import units as u
from astropy.coordinates import SkyCoord
import glob

colorList = np.array(list(mcolors.TABLEAU_COLORS.items()))[:, 1]
```

In [ ]:
```
from load_supermock import *
from sample_analysis_utils import *
```

Download the catalog from https://portal.nersc.gov/project/hacc/spherex_sky/

(or directly from NERSC-Perlmutter:/global/cfs/cdirs/hacc/www/spherex_sky, if you have access)

Catalog entries

The following are the mock observation catalog entries

- **galaxy_id**: *Galaxy ID*
- **ra_true**: *Right ascension coordinates of the galaxies on the full sky*
- **dec_true**: *Declination coordinates of the galaxies on the full sky*
- **redshift_true**: *Redshift of the galaxies*
- **SED**: *Spectral energy distributions in [Jy]*
- **SED_wavelength**: *Wavelength of the spectra in [A]*
- **mag_\<n>_spherex**: *SPHEREx bandpasses in n:0-101*
- **mag_\<n>_cosmos**: *COSMOS magnitudes in n:0-30*
- **mag_\<x>_lsst**: *LSST magnitudes for x: {u, g, r, i, z, Y}*
- **mag_\<n>_wise**: *WISE magnitudes in n:0-6*
- **mag_\<n>_mass2**: *MASS2 magnitudes in n:0-2*
- **mag_\<n>_ls**: *DECam Legacy Survey magnitudes in n:0-7*
- **mag_\<n>_f784**: *f784 magnitude*

The following are the simulation-only entries

- **halo_mass**: *Total mass of the host halo in [Msun/h]*
- **stellar_mass**: *Stellar mass of the galaxy in [Msun/h]*
- **position_x**: *x-coordinate of the host core in [Mpc/h]*
- **position_y**: *y-coordinate of the host core in [Mpc/h]*
- **position_z**: *z-coordinate of the host core in [Mpc/h]*
- **velocity_x**: *x-coordinate of the host core in [km/s]*
- **velocity_y**: *y-coordinate of the host core in [km/s]*
- **velocity_z**: *z-coordinate of the host core in [km/s]*
- **is_central**: *Type of the host core: Central=1, Satellite=0*
- **SFH**: *Star formation history of the galaxy*
- **time_bins_SFH**: *Time bins for the SFH in [Gyr]*
- **halo_id**: *Host core ID*

Listing all the available catalog files

In [ ]:
```
all_available_catalog_files = sorted(glob.glob('SuperMocks/*.hdf5'), key=extra
print('Available catalog files\n' + 10*'==')
print(*all_available_catalog_files, sep='\n')
```

```
Available catalog files
====================
SuperMocks/SuperMock_v3_core_5_every_20.hdf5
SuperMocks/SuperMock_v3_core_5_every_10.hdf5
SuperMocks/SuperMock_v3_core_45_every_50.hdf5
SuperMocks/SuperMock_v3_core_48_every_20.hdf5
SuperMocks/SuperMock_v3_core_75_every_20.hdf5
SuperMocks/SuperMock_v3_core_119_every_10.hdf5
SuperMocks/SuperMock_v3_core_142_every_50.hdf5
SuperMocks/SuperMock_v3_core_144_every_100.hdf5
SuperMocks/SuperMock_v3_core_180_every_10.hdf5
SuperMocks/SuperMock_v3_core_181_every_20.hdf5
SuperMocks/SuperMock_v3_core_182_every_20.hdf5
```

Reading catalog files -- reading single file

Reading all attributes from the catalog

Also performing simple selections (such as removing invalid magnitude values)

In [ ]:

```
# Testing the updated function on the provided file
catalog_name = all_available_catalog_files[4]

Vol = 5025**3
test_data_single, removed_test_data_single, test_items_single = load_and_clean_
```

```
Catalog: SuperMocks/SuperMock_v3_core_75_every_20.hdf5
Total number of original galaxies: 343768
Total number of cleaned galaxies: 203196
Total number of removed galaxies: 140572
=--==--==--==--==--==--==--==--==--==--=
```

Features of the SuperMock catalog files

- ##### The catalogs files do not randomly sample the entire sky.
- ##### Instead they sample either parts of the RA-dec-redshift space (for instance, load core: 180 or core: 181)
- ##### This coverage depends entirely on how the core histories intersect with the lightcones.

Reading catalog files -- reading multiple files

This can be slow and memory intensive.

In [ ]:

```
# test_data, test_items = load_all_available_catalogs(dirIn = 'SuperMocks/', ex
test_data, test_items = load_all_available_catalogs(dirIn = 'SuperMocks/') # Re
```

```
Catalog: SuperMocks/SuperMock_v3_core_5_every_20.hdf5
Total number of original galaxies: 350262
Total number of cleaned galaxies: 215658
Total number of removed galaxies: 134604
========================================
Catalog: SuperMocks/SuperMock_v3_core_5_every_10.hdf5
Total number of original galaxies: 700524
Total number of cleaned galaxies: 431377
Total number of removed galaxies: 269147
========================================
Catalog: SuperMocks/SuperMock_v3_core_45_every_50.hdf5
Total number of original galaxies: 140038
Total number of cleaned galaxies: 86637
Total number of removed galaxies: 53401
========================================
Catalog: SuperMocks/SuperMock_v3_core_48_every_20.hdf5
Total number of original galaxies: 355540
Total number of cleaned galaxies: 209552
Total number of removed galaxies: 145988
========================================
Catalog: SuperMocks/SuperMock_v3_core_75_every_20.hdf5
Total number of original galaxies: 343768
Total number of cleaned galaxies: 203196
Total number of removed galaxies: 140572
========================================
Catalog: SuperMocks/SuperMock_v3_core_119_every_10.hdf5
Total number of original galaxies: 685037
Total number of cleaned galaxies: 405917
Total number of removed galaxies: 279120
========================================
Catalog: SuperMocks/SuperMock_v3_core_142_every_50.hdf5
Total number of original galaxies: 135385
Total number of cleaned galaxies: 85725
Total number of removed galaxies: 49660
========================================
Catalog: SuperMocks/SuperMock_v3_core_144_every_100.hdf5
Total number of original galaxies: 3121
Total number of cleaned galaxies: 3102
Total number of removed galaxies: 19
========================================
Catalog: SuperMocks/SuperMock_v3_core_180_every_10.hdf5
Total number of original galaxies: 689115
Total number of cleaned galaxies: 396246
Total number of removed galaxies: 292869
========================================
Catalog: SuperMocks/SuperMock_v3_core_181_every_20.hdf5
Total number of original galaxies: 336349
Total number of cleaned galaxies: 201133
Total number of removed galaxies: 135216
========================================
Catalog: SuperMocks/SuperMock_v3_core_182_every_20.hdf5
Total number of original galaxies: 335765
Total number of cleaned galaxies: 201072
Total number of removed galaxies: 134693
========================================
Grand total number of cleaned galaxies: 2439615
```

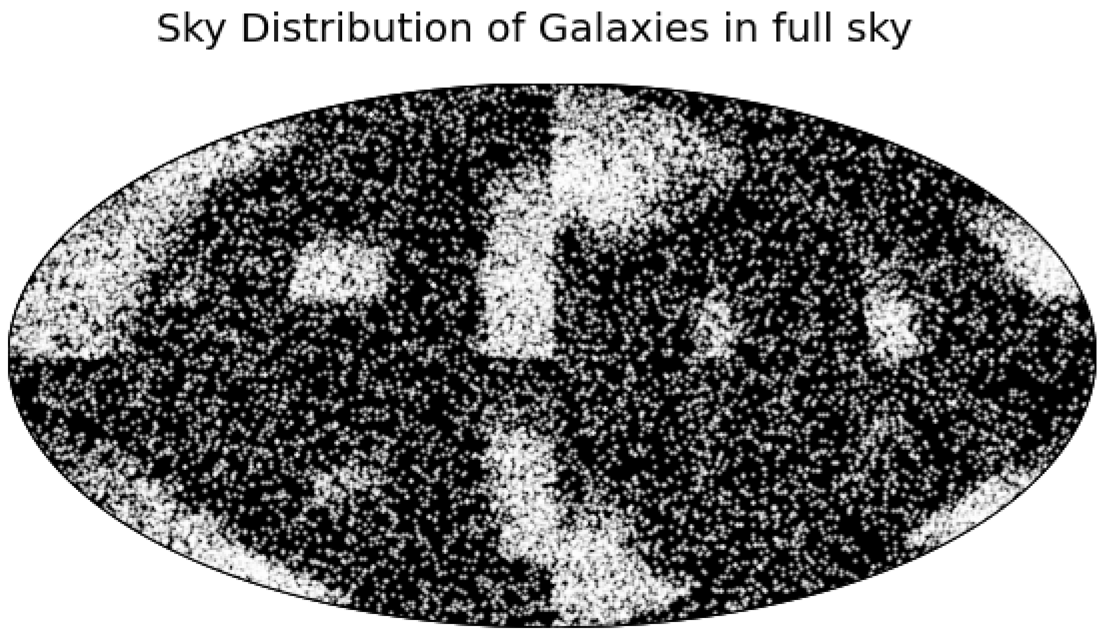
Accessing desired catalog entries

In [ ]:
```
ra_full = np.array(test_data['ra_true'])
dec_full = np.array(test_data['dec_true'])
redshift = np.array(test_data['redshift_true'])
halo_mass = np.array(test_data['halo_mass'])
stellar_mass = np.array(test_data['stellar_mass'])
is_central = np.array(test_data['is_central'])

wavelength = np.array(test_data['SED_wavelength']) #this is rest frame waveleng
SEDs = np.array(test_data['SED'])
luminosity = np.array(test_data['luminosity']) #incorrect calculation

mag_u = np.array(test_data['mag_u_sdss'])
mag_g = np.array(test_data['mag_g_sdss'])
mag_r = np.array(test_data['mag_r_sdss'])
mag_i = np.array(test_data['mag_i_sdss'])
mag_z = np.array(test_data['mag_z_sdss'])
mag_Y = np.array(test_data['mag_Y_sdss'])
```

Sample analyses using the catalog

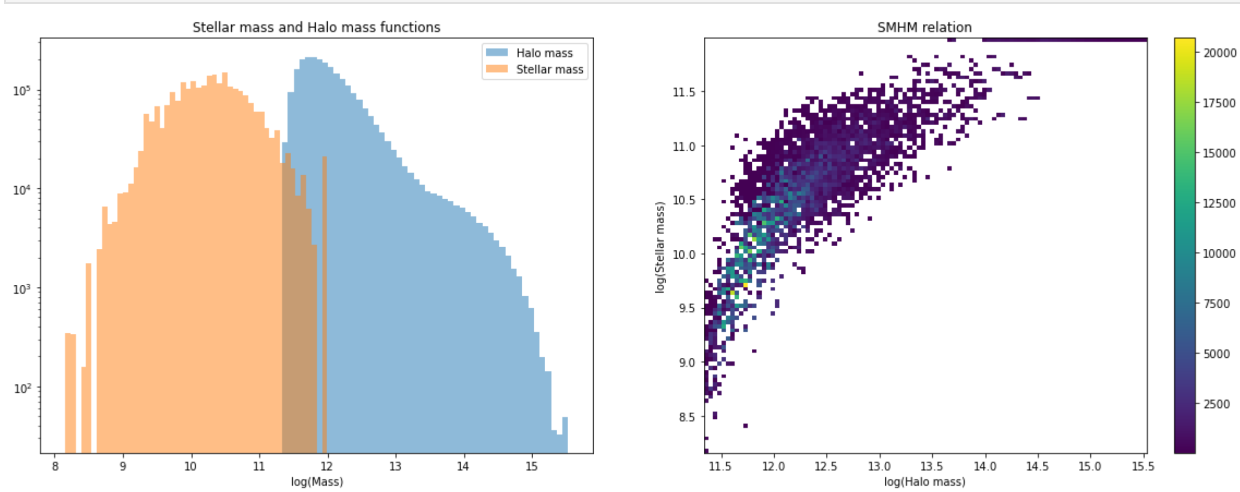
1. Sky distribution using the mollweide projection

In [ ]:
```
fig = plt.figure(figsize=(10, 5))

##########################
ax = fig.add_subplot(111)

# Only plotting first 10,000 galaxies
n_gal = 20000
random_gal_indices = np.random.randint(low=0, high=ra_full.shape[0], size=n_ga

ra_octant = np.array(test_data['ra_true'][random_gal_indices])
dec_octant = np.array(test_data['dec_true'][random_gal_indices])

# Define the orthographic projection centered on the equator and prime meridia
m = Basemap(projection='moll', lat_0=-60, lon_0=90, resolution='c')
# Convert RA, Dec to x, y coordinates for plotting
x, y = m(ra_octant, dec_octant)

# Plot the sky distribution
m.scatter(x, y, s=2, c='white', alpha=0.5, edgecolors='w', linewidth=1)

# Draw parallels and meridians
# m.drawparallels(np.arange(-90.,90.,22.5), color='yellow', textcolor='yellow',
# m.drawmeridians(np.arange(0.,360.,22.5), color='yellow', textcolor='yellow',
m.drawmapboundary(fill_color='black')
# m.drawcoastlines(color='black', linewidth=0.5)

plt.suptitle('Sky Distribution of Galaxies in full sky', fontsize=20)
plt.show()
```



2. Halo mass function, stellar mass function and Stellar mass-to-halo mass relationship

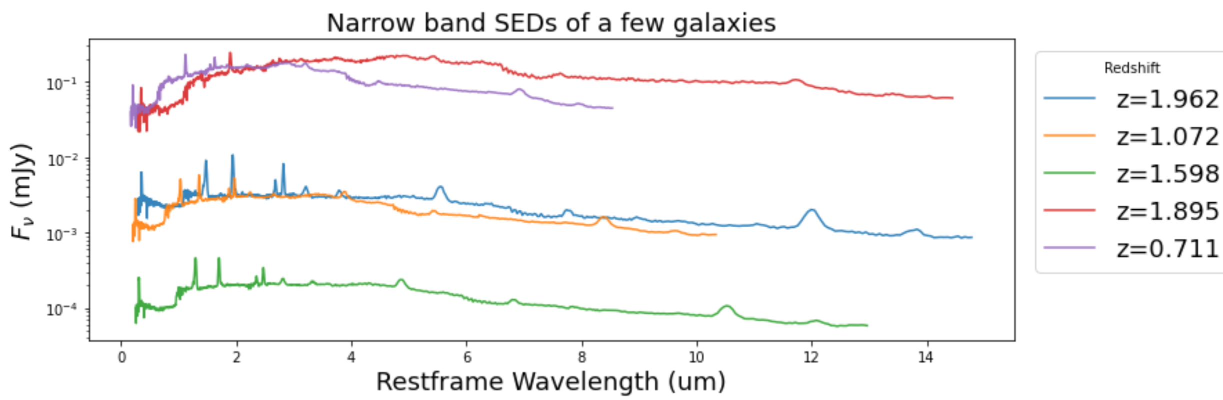
In [ ]:
```
fig = plt.figure(figsize=(20, 7))
ax = fig.add_subplot(121)

ax.set_title('Stellar mass and Halo mass functions')
ax.hist(np.log10(halo_mass), bins=50, alpha=0.5, label='Halo mass');
ax.hist(np.log10(stellar_mass), bins=50, alpha=0.5, label='Stellar mass');
ax.set_yscale('log')
ax.set_xlabel('log(Mass)')
ax.legend()

ax = fig.add_subplot(122)
_, _, _, im = ax.hist2d(np.log10(halo_mass), np.log10(stellar_mass), bins=(100

ax.set_title('SMHM relation')
ax.set_xlabel('log(Halo mass)')
ax.set_ylabel('log(Stellar mass)')
plt.colorbar(im)

plt.show()
```



3. SEDs of randomly chosen galaxies

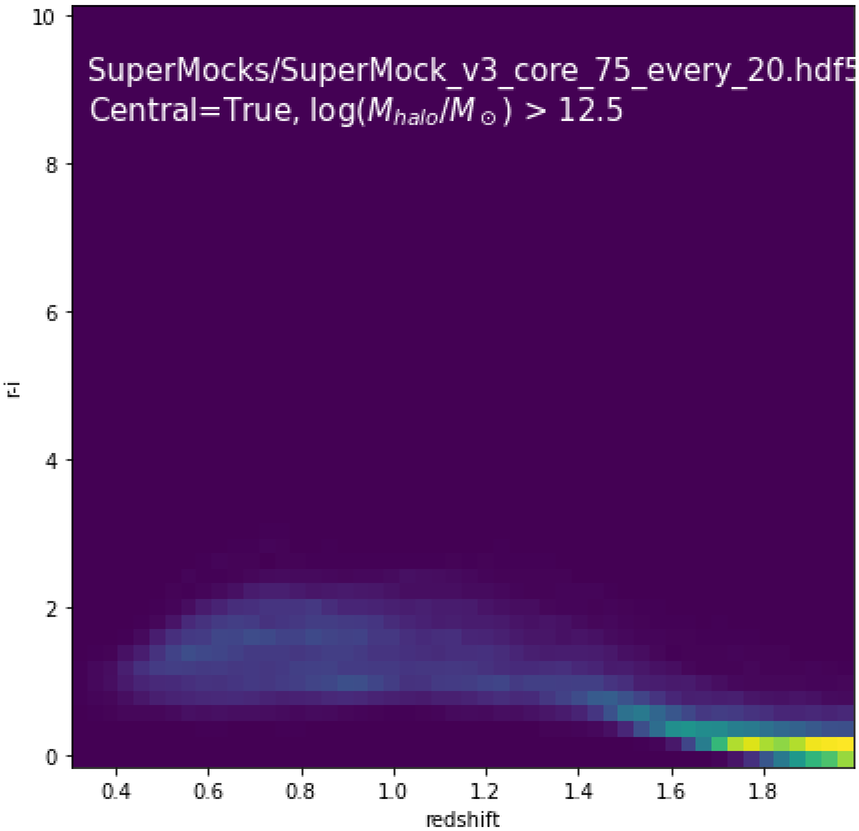
In [ ]:
```
np.random.seed(8)
galID_arr = np.random.randint(low=0, high=SEDs.shape[0], size=5)

plt.figure(figsize=(12,4))

for idx, galID in enumerate(galID_arr):

    obs_frame_wave = wavelength*(1 + redshift[galID])
    plt.plot(obs_frame_wave/1e4 ,
             SEDs[galID]*1e3,
             label='z=%.3f'%redshift[galID],
             color=colorList[idx],
             alpha=0.9);

plt.ylabel(r'$F_\nu$ (mJy)', fontsize=18)
plt.xlabel(' Restframe Wavelength (um) ', fontsize=18)
plt.yscale('log')
plt.legend(ncol=1, title='Redshift', fontsize=18, bbox_to_anchor=(1.01, 1.0),
plt.title('Narrow band SEDs of a few galaxies', fontsize=18)
```

Out[ ]:
```
Text(0.5, 1.0, 'Narrow band SEDs of a few galaxies')
```



4. Color-redshift relationship for Central galaxies (SDSS-r and SDSS-i bands)

In [ ]:
```
color_ri = mag_r - mag_i
mass_cut = 12.5
cond = np.where( (is_central==True) & ( np.log10(halo_mass) > mass_cut) )

redshift_select = redshift[cond]
ri_select = color_ri[cond]

f, ax = plt.subplots(1,1, figsize =(7, 7))
plt.hist2d(x=redshift_select, y=ri_select, bins=50);
plt.xlabel('redshift')
plt.ylabel('r-i')
plt.text(x=0.02, y=0.85, s='%s \nCentral=True, log($M_{halo}/M_\odot$) > %.1f'
         transform=ax.transAxes, fontsize=15)
plt.show()
```



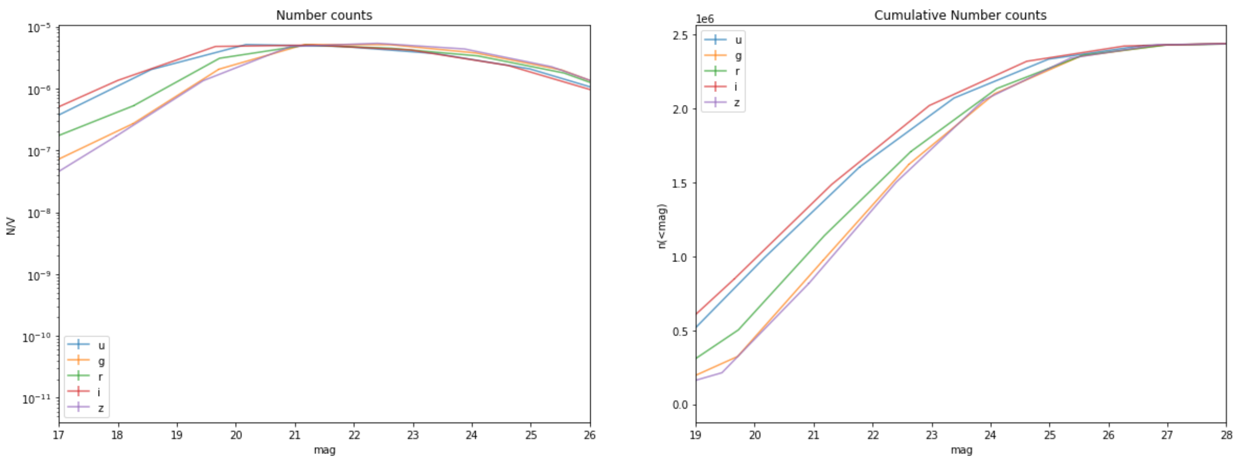
5. SDSS magnitude distributions

In [ ]:
```
def plt_errorbar(x, label):
    y, bin_edges = np.histogram(x, bins = 50)
    bin_centers = 0.5*(bin_edges[1:] + bin_edges[:-1])
    plt.errorbar(bin_centers, y/Vol, yerr = (y**0.5)/Vol, label=label, alpha=0

def plt_cumulative(x, label):
    y, bin_edges = np.histogram(x, bins = 50)
    bin_centers = 0.5*(bin_edges[1:] + bin_edges[:-1])
    y_cumulative = np.cumsum(y)
    plt.errorbar(bin_centers, y_cumulative, yerr = (y_cumulative**0.5), label=
    plt.xlim(19, 28)


fig = plt.figure(figsize=(20, 7))
ax = fig.add_subplot(121)


plt_errorbar(mag_u, 'u')
plt_errorbar(mag_g, 'g')
plt_errorbar(mag_r, 'r')
plt_errorbar(mag_i, 'i')
plt_errorbar(mag_z, 'z')
plt.title('Number counts')
plt.xlabel('mag')
plt.ylabel('N/V')
plt.xlim(17, 26)
# plt.ylim(1e-5, 1e-2)
plt.yscale('log')
plt.legend()
```

```
ax = fig.add_subplot(122)

# plt.figure(figsize=(9, 5))
plt_cumulative(mag_u, 'u')
plt_cumulative(mag_g, 'g')
plt_cumulative(mag_r, 'r')
plt_cumulative(mag_i, 'i')
plt_cumulative(mag_z, 'z')
plt.title('Cumulative Number counts')
plt.xlabel('mag')
plt.ylabel('n(<mag)')
plt.xlim(19, 28)
plt.legend()
```

Out[ ]:

```
<matplotlib.legend.Legend>
```



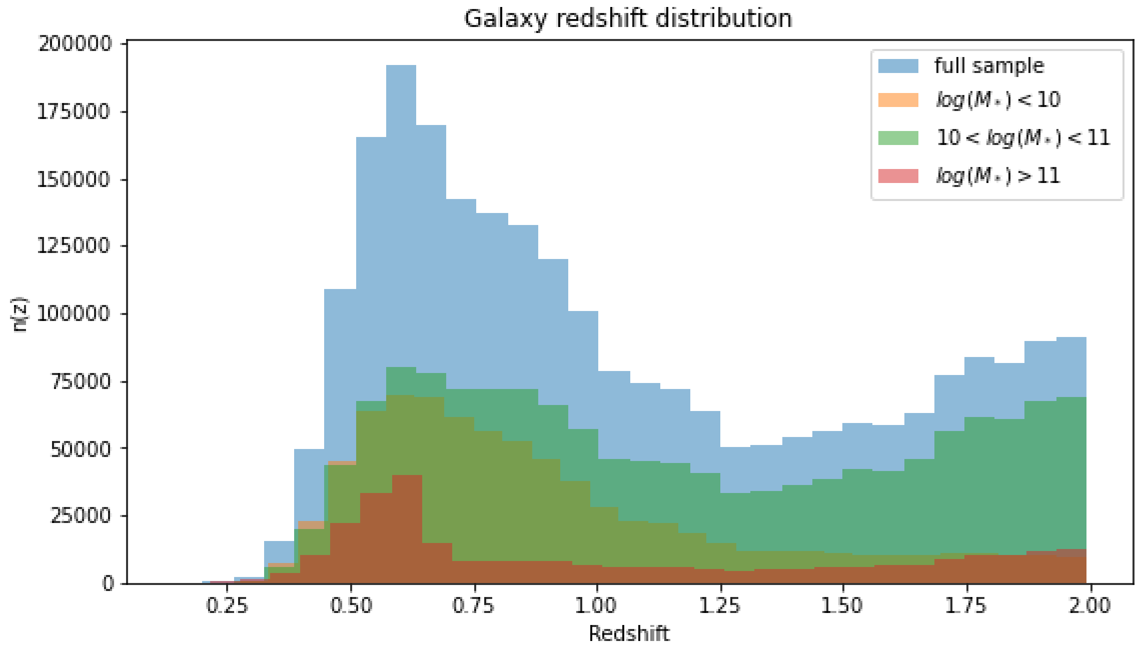
6. Color-redshift relationship for Central galaxies (SDSS-r and SDSS-i bands)

In [ ]:

```
plt.figure(figsize=(9, 5))
plt.hist( redshift, bins = 30, label='full sample', alpha=0.5);
plt.hist( redshift[np.where( np.log10(stellar_mass) < 10) ], bins = 30, label=
plt.hist( redshift[np.where( (np.log10(stellar_mass) > 10) & (np.log10(stellar_
plt.hist( redshift[np.where( np.log10(stellar_mass) > 11) ], bins = 30, label=


plt.title('Galaxy redshift distribution')
plt.xlabel('Redshift')
plt.ylabel('n(z)')
# plt.xlim(0, 0.52)
plt.legend()
```

Out[ ]:

```
<matplotlib.legend.Legend>
```



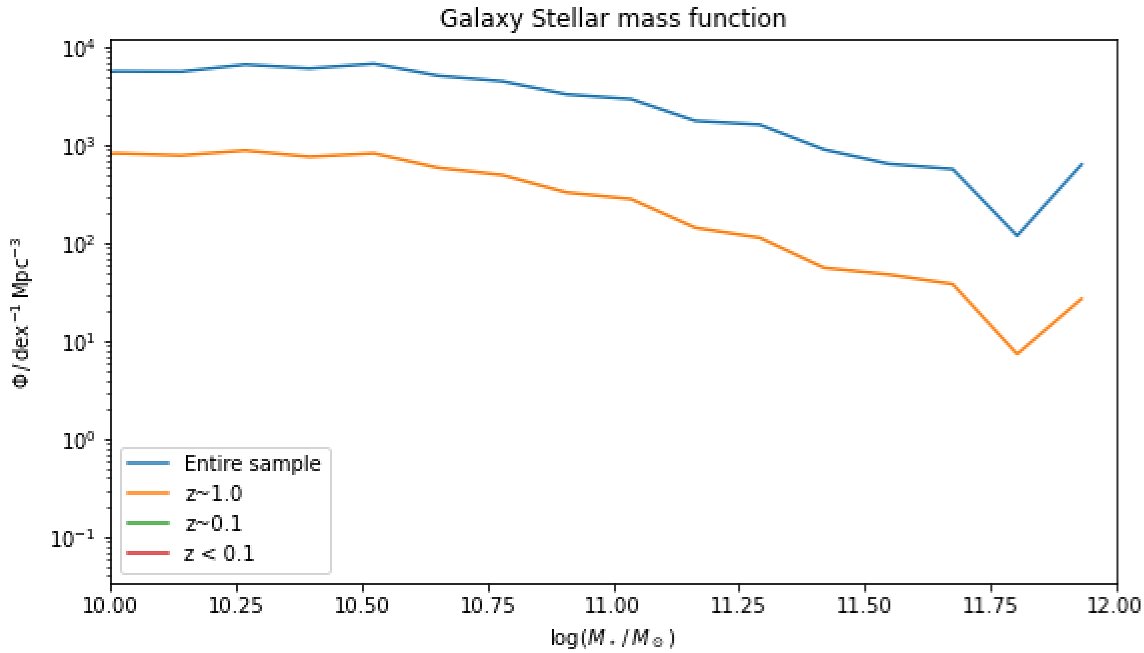
7. Galaxy stellar mass functions

In [ ]:

```
Max, Phi = GSMF(stellar_mass[stellar_mass > 9])
Max0_2, Phi0_2 = GSMF(stellar_mass[ (stellar_mass > 9)&(redshift < 1.1) &(redsl
Max0_1, Phi0_1 = GSMF(stellar_mass[ (stellar_mass > 9)&(redshift < 0.11) &(red:
Max0_05, Phi0_05 = GSMF(stellar_mass[ (stellar_mass > 9)&(redshift < 0.1) &(re(


plt.figure(figsize=(9, 5))
plt.title('Galaxy Stellar mass function')
plt.yscale('log')
plt.xlabel(r'$\log(M_\star\,/\,M_\odot)$')
plt.ylabel(r'$\Phi\,/\,\mathrm{dex}^{-1}\,\mathrm{Mpc}^{-3}$')

plt.plot( Max, Phi , ls='-', label='Entire sample')
plt.plot( Max0_2, Phi0_2 , ls='-', label='z~1.0')
plt.plot( Max0_1, Phi0_1 , ls='-', label='z~0.1')
plt.plot( Max0_05, Phi0_05 , ls='-', label='z < 0.1')

plt.xlim(10, 12)

plt.legend()
```

Out[ ]:

```
<matplotlib.legend.Legend>
```

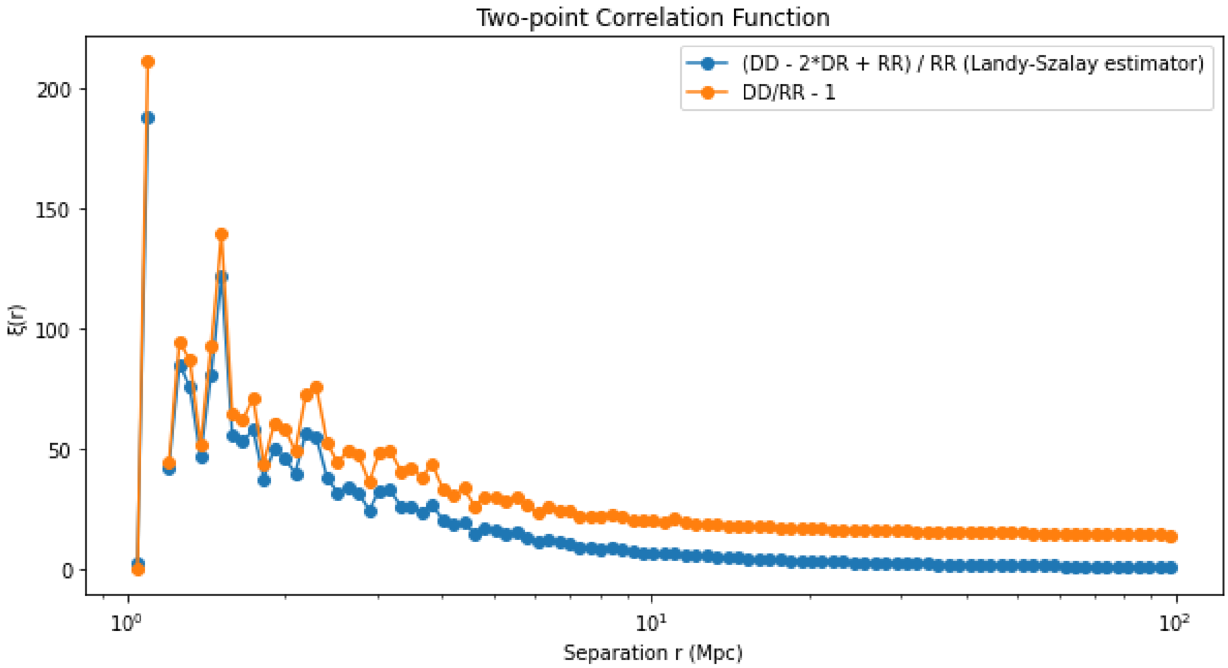


8. Two-point correlation function (may not be accurate)

In [ ]:
```
x_subset, y_subset, z_subset = radec_to_cartesian(ra_full, dec_full, redshift)
bin_centers, xi, xi_Landy_Szalay = xyz_to_xi(x_subset, y_subset, z_subset)
```

```
/lcrc/project/cosmo_ai/nramachandra/Projects/SPHEREx/MAH/HACCnPaint/SyntheticC
atalog/sample_analysis_utils.py:46: RuntimeWarning: divide by zero encountered
in divide
  xi = DD / RR - 1
/lcrc/project/cosmo_ai/nramachandra/Projects/SPHEREx/MAH/HACCnPaint/SyntheticC
atalog/sample_analysis_utils.py:55: RuntimeWarning: divide by zero encountered
in divide
  xi_Landy_Szalay = (DD - 2*DR + RR) / RR
```

In [ ]:
```
# Plot the two-point correlation function
plt.figure(figsize=(9, 5))
plt.plot(bin_centers, xi_Landy_Szalay, marker='o', linestyle='-', label='(DD -
plt.plot(bin_centers, xi, marker='o', linestyle='-', label='DD/RR - 1')

plt.xscale('log')
# plt.yscale('log')
plt.xlabel('Separation r (Mpc)')
plt.ylabel('ξ(r)')
plt.title('Two-point Correlation Function')
# plt.grid(True, which="both", ls="--")
plt.legend()
plt.tight_layout()
plt.show()
```



We also provide auxilary pickle files for information about survey bandpasses

In [ ]:
```
central_wavelengths, bandpass_wavs, bandpass_vals, bandpass_names = load_survey
```

In [ ]:
```
bandpass_names[::10]
```

Out[ ]:
```
['SPHEREx_band1_ch16',
 'SPHEREx_band1_ch2',
 'SPHEREx_band2_ch13',
 'SPHEREx_band2_ch16',
 'SPHEREx_band3_ch9',
 'SPHEREx_band3_ch12',
 'SPHEREx_band4_ch5',
 'SPHEREx_band5_ch9',
 'SPHEREx_band5_ch12',
 'SPHEREx_band6_ch17',
 'SPHEREx_band6_ch4']
```

9. Reading spherex bandpasses, plotting number densities in (color vs redshift) plot

In [ ]:
```
mag_spherex_0 = np.array(test_data['mag_0_spherex'])
mag_spherex_1 = np.array(test_data['mag_1_spherex'])
mag_spherex_2 = np.array(test_data['mag_2_spherex'])
mag_spherex_3 = np.array(test_data['mag_3_spherex'])
mag_spherex_4 = np.array(test_data['mag_4_spherex'])
mag_spherex_5 = np.array(test_data['mag_5_spherex'])
mag_spherex_6 = np.array(test_data['mag_6_spherex'])
mag_spherex_7 = np.array(test_data['mag_7_spherex'])
mag_spherex_8 = np.array(test_data['mag_8_spherex'])

s1 = np.array([mag_spherex_0, mag_spherex_1, mag_spherex_2, mag_spherex_3, mag_
allLabels = [ 'Mag(' +str(i+1)+ ') - Mag(' +str(i)+ ')' for i in range(s1.shape
```

In [ ]:
```
ncols = 2
nrows = 2

f, a = plt.subplots(ncols= ncols, nrows = nrows, sharex = True, figsize=(ncols
f.subplots_adjust(hspace=0.2, wspace=0.2, left=0.01, right=0.99)

for col_idx in range(ncols):
    for row_idx in range(nrows):

        band_idx = row_idx*nrows + col_idx
        hb = a[col_idx, row_idx].hexbin(s1[:, -1], s1[:, band_idx], gridsize= 
        a[col_idx, row_idx].set_xlabel(r'$z$', fontsize = "xx-large")
        a[col_idx, row_idx].set_ylabel(allLabels[band_idx], fontsize = "xx-larg
        #a[col_idx, row_idx].text(0.2, 0.9, allLabels[band_idx], horizontalali
        # a[col_idx, row_idx].text(0.05, 0.9, 'channel i (μm): ' + str(lambda_
        a[col_idx, row_idx].text(0.05, 0.95,
                                 r'Channel i: $\lambda_{cen} [\mu m]$: '
                                 + str( round(central_wavelengths[band_idx], 3
                                 horizontalalignment='left', verticalalignment


        a[col_idx, row_idx].text(0.05, 0.9,
                                 r'Channel j: $\lambda_{cen} [\mu m]$: '
                                 + str( round(central_wavelengths[band_idx + 1
                                 horizontalalignment='left', verticalalignment
        #a[col_idx, row_idx].set(aspect='equal')
        cb = f.colorbar(hb, ax=a[col_idx, row_idx])
        cb.set_label(r'$N_{\rm galaxies}$', fontsize = "xx-large")
        a[col_idx, row_idx].set_ylim(10, 30)
```