The WaZP cluster finder on DC2 & The ClEvaR package

Michel Aguena

27/05/2021

Cluster Detection Validation

Cluster **Eva**luation **R**esources

ClEvaR

Legacy Survey of Space and Time

- Goal

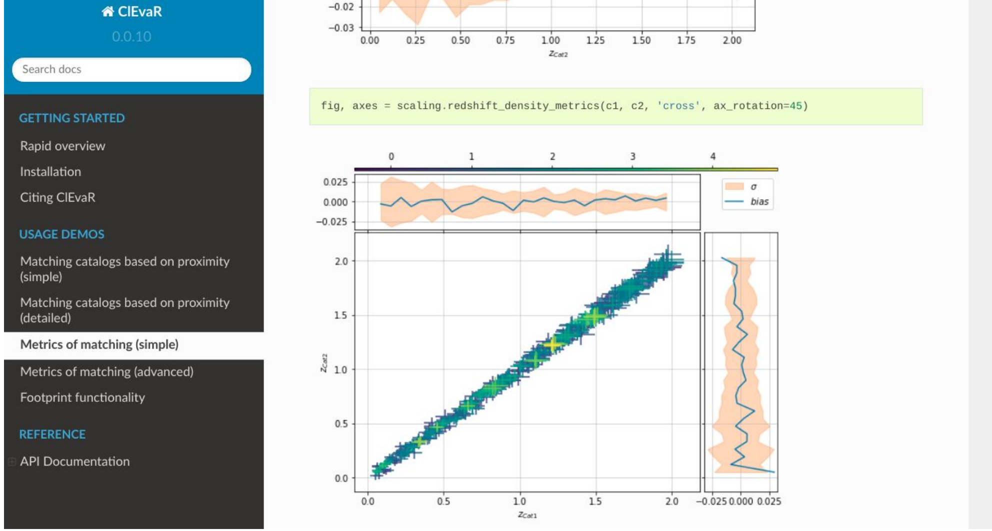
  Compare and validate cluster catalogs

- Functionality
  - Self consistency checks of catalog properties
  - Easy matching with other catalogs (cluster/halos)
  - Metrics of matched catalogs (selection function) & scaling relations (mass proxy, size, orientation, redshift)

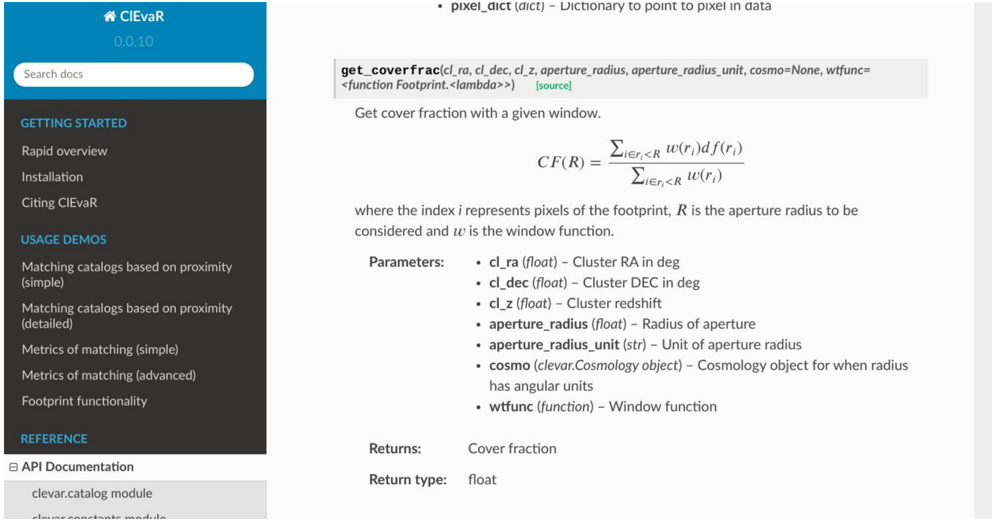
- Objectives
  - Code development inside the DESC pipeline framework (documentation, versioning, unit tests)
  - Modular structure to allow for integration with other libraries
  - Robust executables based on configuration files for automatic runs of pipeline

Demonstration and Documentation

- Code on DESC github: https://github.com/LSSTDESC/clevar
- Notebooks: https://github.com/LSSTDESC/clevar/tree/master/examples
- Documentation in code
- ClEvaR Doc, API and Demos on DESC: http://lsstdesc.org/clevar/



```
fig, axes = scaling.redshift_density_metrics(c1, c2, 'cross', ax_rotation=45)
```



- **pixel_dict** (*dict*) – Dictionary to point to pixel in data

get_coverfrac(*cl_ra, cl_dec, cl_z, aperture_radius, aperture_radius_unit, cosmo=None, wtfunc=<function Footprint.<lambda>>*) [source]

Get cover fraction with a given window.

$$CF(R) = \frac{\sum_{i \in r_i < R} w(r_i) df(r_i)}{\sum_{i \in r_i < R} w(r_i)}$$

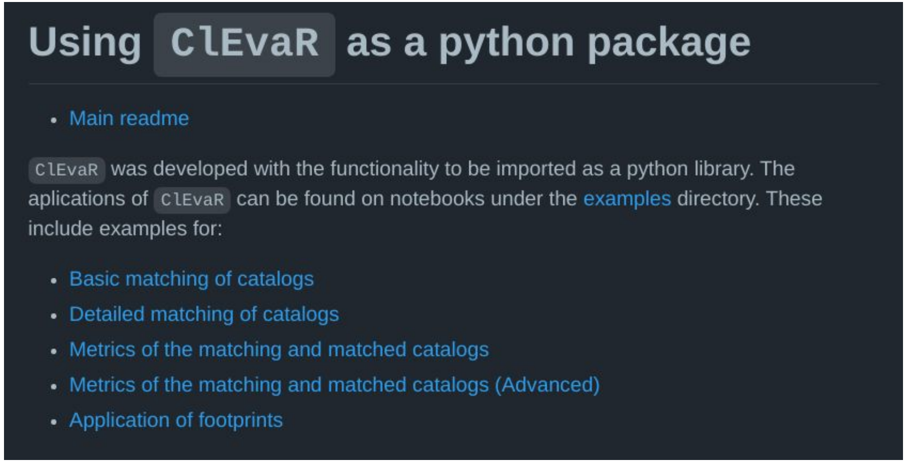
where the index i represents pixels of the footprint, R is the aperture radius to be considered and w is the window function.

Parameters:
- **cl_ra** (*float*) – Cluster RA in deg
- **cl_dec** (*float*) – Cluster DEC in deg
- **cl_z** (*float*) – Cluster redshift
- **aperture_radius** (*float*) – Radius of aperture
- **aperture_radius_unit** (*str*) – Unit of aperture radius
- **cosmo** (*clevar.Cosmology object*) – Cosmology object for when radius has angular units
- **wtfunc** (*function*) – Window function

Returns: Cover fraction

Return type: float

Modes for running

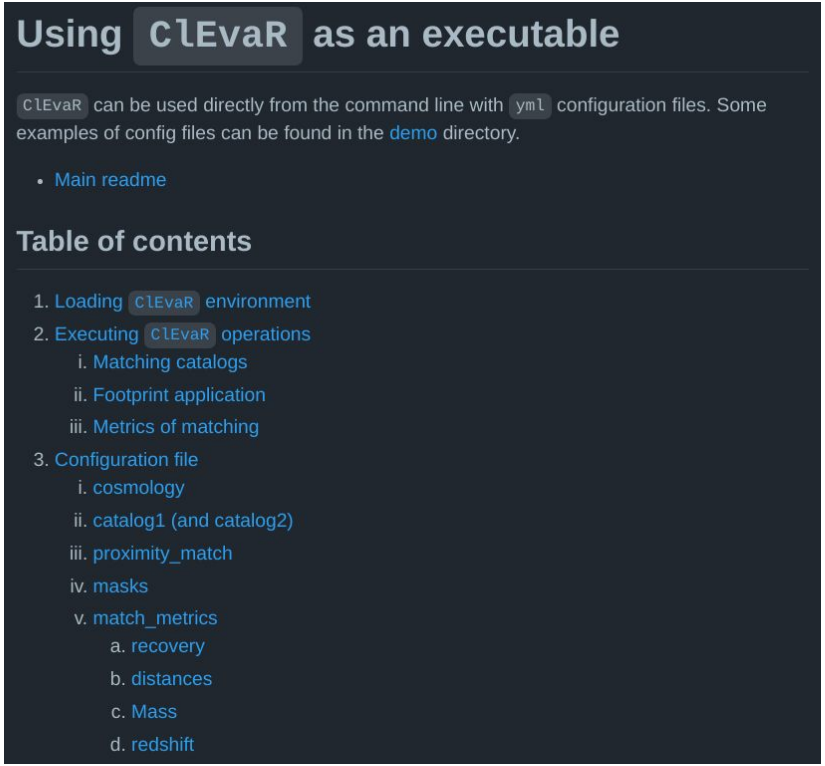


Using `ClEvaR` as a python package

- Main readme

`ClEvaR` was developed with the functionality to be imported as a python library. The aplications of `ClEvaR` can be found on notebooks under the examples directory. These include examples for:

- Basic matching of catalogs
- Detailed matching of catalogs
- Metrics of the matching and matched catalogs
- Metrics of the matching and matched catalogs (Advanced)
- Application of footprints



Using `ClEvaR` as an executable

`ClEvaR` can be used directly from the command line with `yml` configuration files. Some examples of config files can be found in the demo directory.

- Main readme

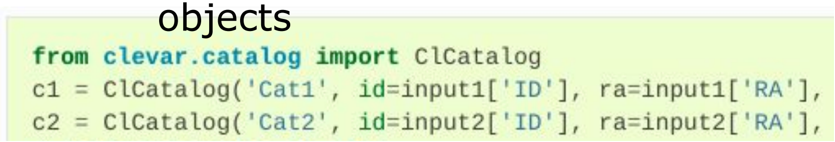
Table of contents

1. Loading `ClEvaR` environment
2. Executing `ClEvaR` operations
   1. Matching catalogs
   2. Footprint application
   3. Metrics of matching
3. Configuration file
   1. cosmology
   2. catalog1 (and catalog2)
   3. proximity_match
   4. masks
   5. match_metrics
      1. recovery
      2. distances
      3. Mass
      4. redshift

Modes for running

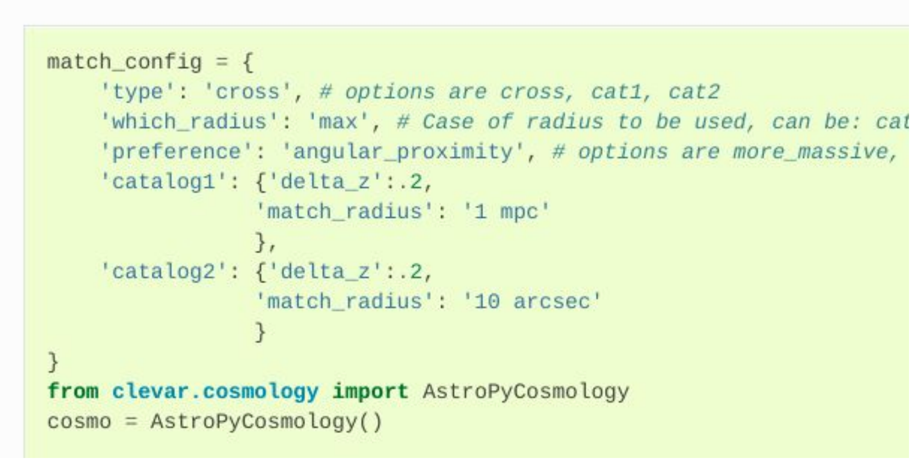
Using ClEvaR as a python package

1. Add catalogs to ClCatalogs objects



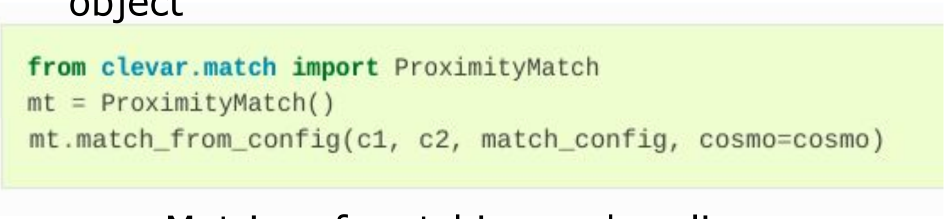
```
from clevar.catalog import ClCatalog
c1 = ClCatalog('Cat1', id=input1['ID'], ra=input1['RA'],
c2 = ClCatalog('Cat2', id=input2['ID'], ra=input2['RA'],
```

2. Prepare config for matching



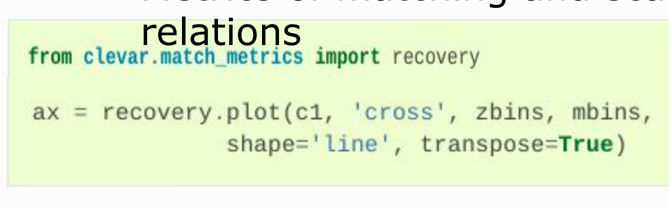
```
match_config = {
    'type': 'cross', # options are cross, cat1, cat2
    'which_radius': 'max', # Case of radius to be used, can be: cat
    'preference': 'angular_proximity', # options are more_massive,
    'catalog1': {'delta_z':.2,
                 'match_radius': '1 mpc'
                 },
    'catalog2': {'delta_z':.2,
                 'match_radius': '10 arcsec'
                 }
}
from clevar.cosmology import AstroPyCosmology
cosmo = AstroPyCosmology()
```

3. Import and Run matching object

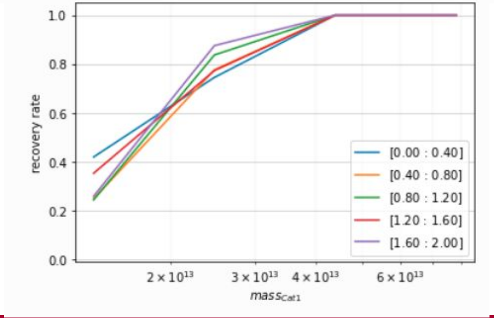


```
from clevar.match import ProximityMatch
mt = ProximityMatch()
mt.match_from_config(c1, c2, match_config, cosmo=cosmo)
```

+ Metrics of matching and scaling relations



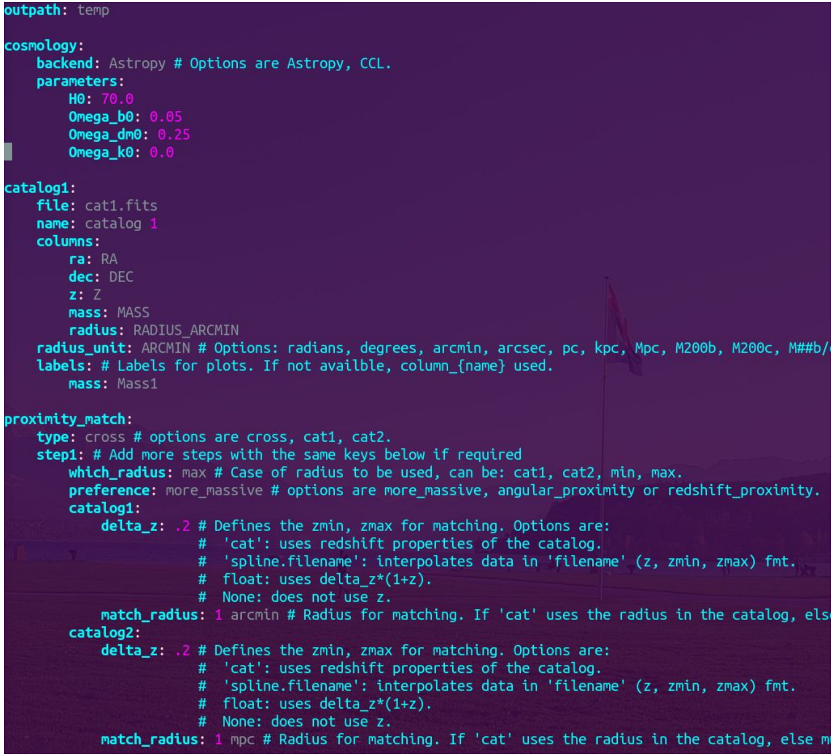
```
from clevar.match_metrics import recovery

ax = recovery.plot(c1, 'cross', zbins, mbins,
                   shape='line', transpose=True)
```



Modes for running

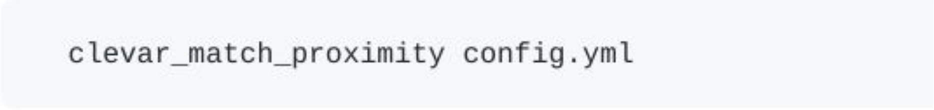
Using ClEvaR as an executable

Configuration (yaml) file

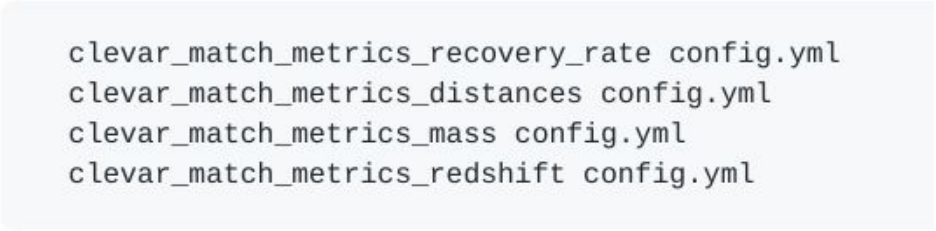


```yaml
outpath: temp

cosmology:
    backend: Astropy # Options are Astropy, CCL.
    parameters:
        H0: 70.0
        Omega_b0: 0.05
        Omega_dm0: 0.25
        Omega_k0: 0.0

catalog1:
    file: cat1.fits
    name: catalog 1
    columns:
        ra: RA
        dec: DEC
        z: Z
        mass: MASS
        radius: RADIUS_ARCMIN
    radius_unit: ARCMIN # Options: radians, degrees, arcmin, arcsec, pc, kpc, Mpc, M200b, M200c, M##b/
    labels: # Labels for plots. If not availble, column_{name} used.
        mass: Mass1

proximity_match:
    type: cross # options are cross, cat1, cat2.
    step1: # Add more steps with the same keys below if required
        which_radius: max # Case of radius to be used, can be: cat1, cat2, min, max.
        preference: more_massive # options are more_massive, angular_proximity or redshift_proximity.
        catalog1:
            delta_z: .2 # Defines the zmin, zmax for matching. Options are:
                        #  'cat': uses redshift properties of the catalog.
                        #  'spline.filename': interpolates data in 'filename' (z, zmin, zmax) fmt.
                        #  float: uses delta_z*(1+z).
                        #  None: does not use z.
            match_radius: 1 arcmin # Radius for matching. If 'cat' uses the radius in the catalog, els
        catalog2:
            delta_z: .2 # Defines the zmin, zmax for matching. Options are:
                        #  'cat': uses redshift properties of the catalog.
                        #  'spline.filename': interpolates data in 'filename' (z, zmin, zmax) fmt.
                        #  float: uses delta_z*(1+z).
                        #  None: does not use z.
            match_radius: 1 mpc # Radius for matching. If 'cat' uses the radius in the catalog, else m
```

Executable commands:



```
clevar_match_proximity config.yml
```



```
clevar_match_metrics_recovery_rate config.yml
clevar_match_metrics_distances config.yml
clevar_match_metrics_mass config.yml
clevar_match_metrics_redshift config.yml
```

Legacy Survey of Space and Time

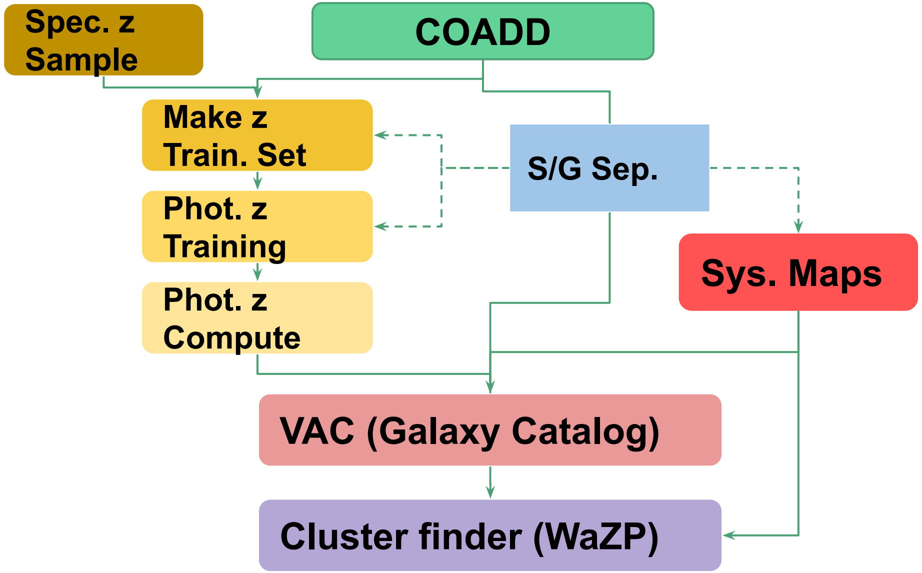
The WaZP cluster finder on DC2

Michel Aguena, Dominique Boutigny, Thibault Guillemin +
Brazil CWG (LIneA)

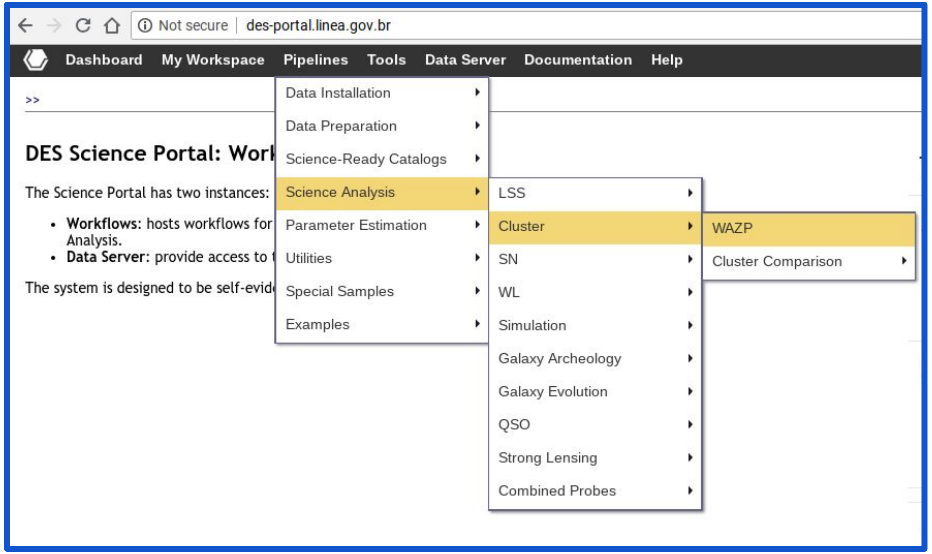
DESC Project (EC approved)

Producing Catalogs

Data workflow



LIneA Science Portal



DES Science Portal: Work

The Science Portal has two instances:

- **Workflows:** hosts workflows for Analysis.
- **Data Server:** provide access to t

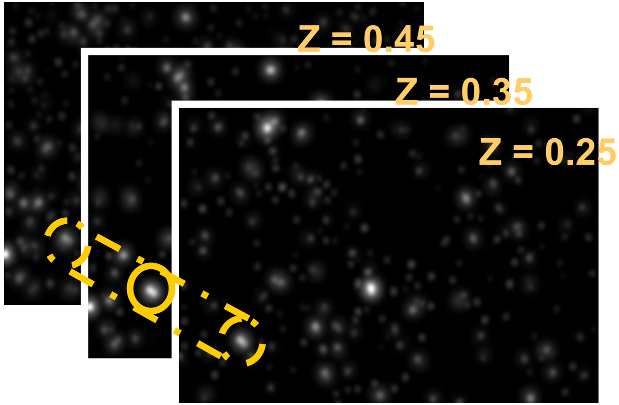
The system is designed to be self-evid

WaZP Cluster Finder on DC2

Wavelet Z-Photometric (WaZP)

Developed by C. Benoist

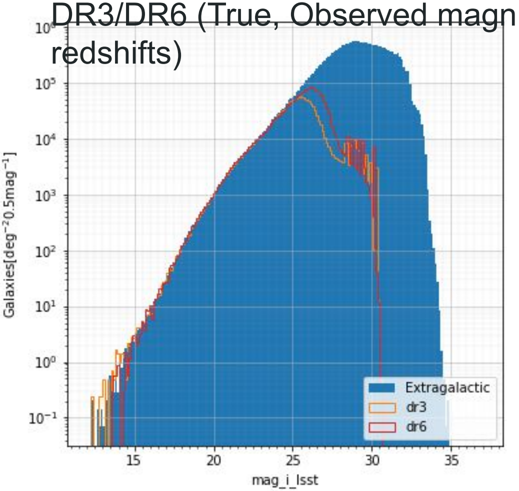
- Galaxies are selected in redshift slices based on PDZ's from photo-z algorithms
- Clusters are detected as overdensities in wavelet based density maps
- No assumption on the galaxy populations of clusters (e.g. red sequence)
- Produces cluster membership probabilities for galaxies



A 3 deg² tile

DC2 Catalogs - cosmoDC2 v1.1.4

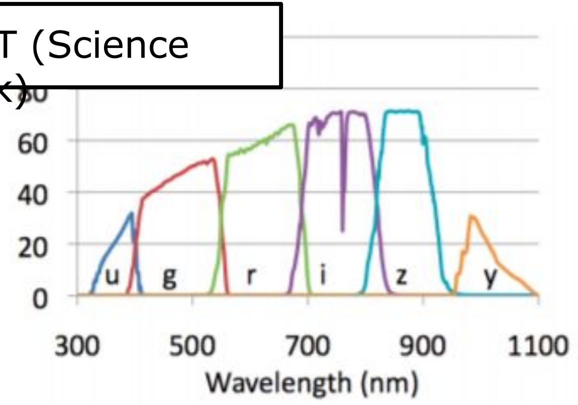
- True catalog:
  - extragalactic galaxy catalog (True, Observed magnitudes and redshifts)
- Observations (run 2.2i):
  - DR3/DR6 (True, Observed magnitudes and redshifts)



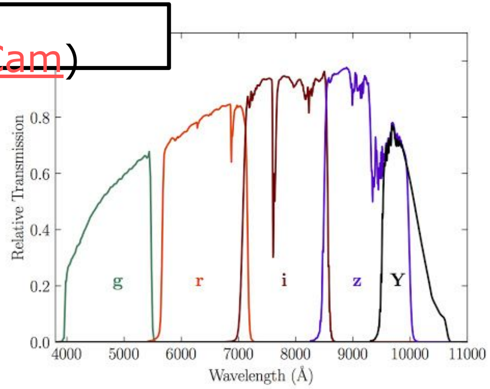
WaZP Cluster Finder on DC2

- Internal calibration has to be updated for LSST magnitudes

LSST (Science book)

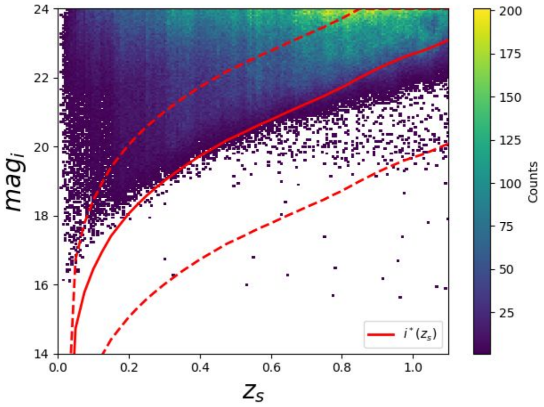


DES (DECam)

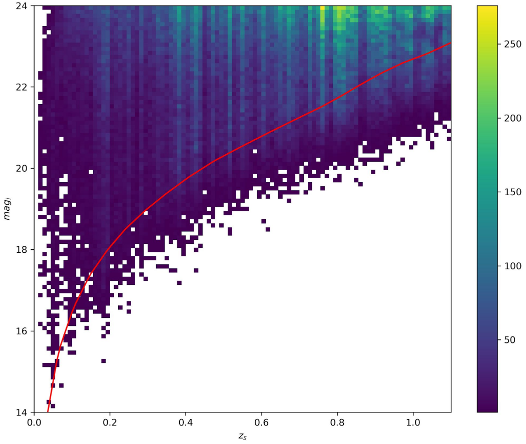


- Effect of magnitude calibration on detection

DC2 run (DR3 - PSF Mag)



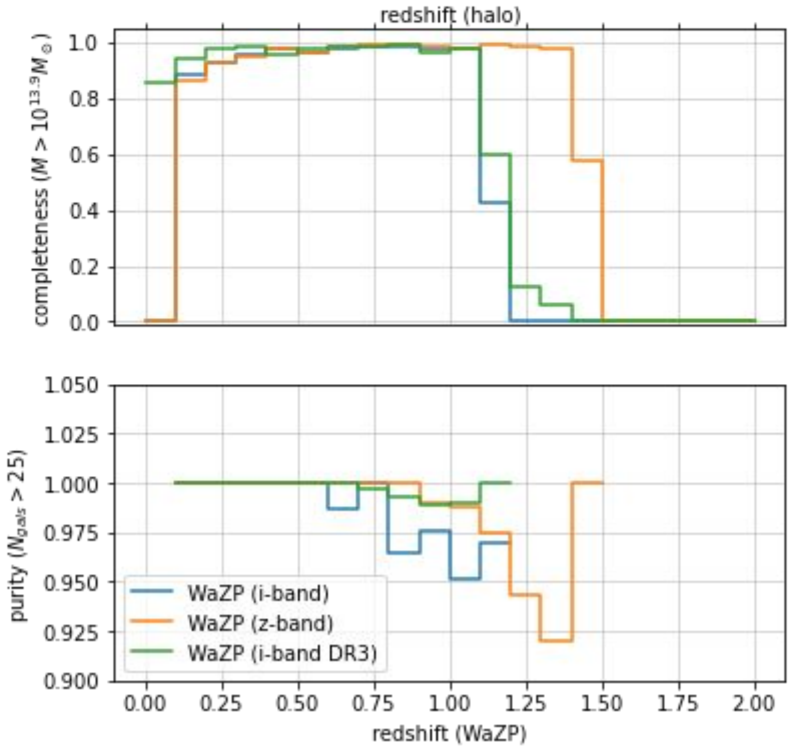
DC2 run (DR3 - True Mag)



WaZP Cluster Finder on DC2

Current status

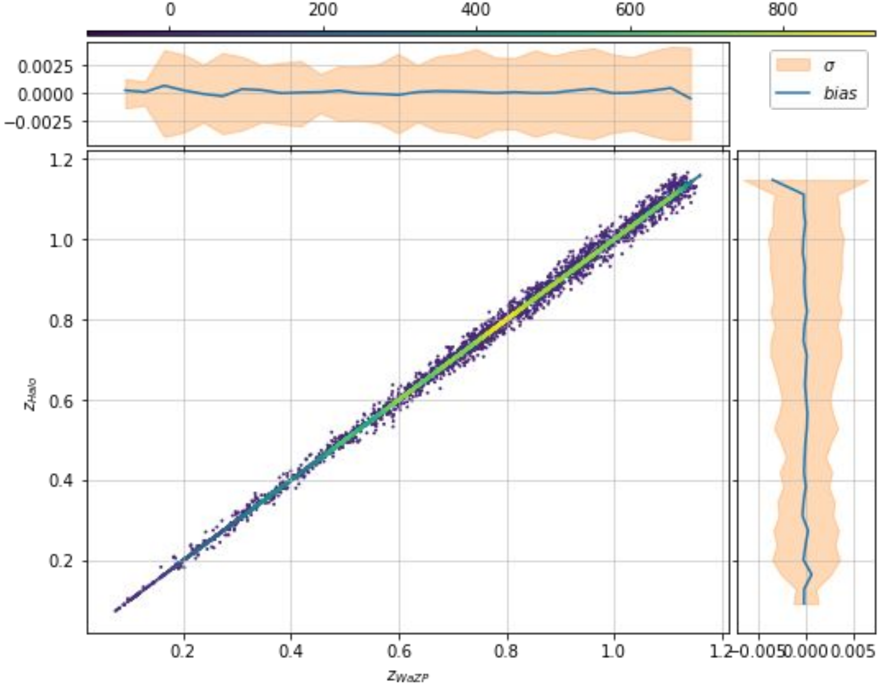
- Pipelines adapted for DC2
- Extragalactic (True)
  - i-band detection
  - z-band detection
- DR3 catalogs
  - i-band detection (preliminary)

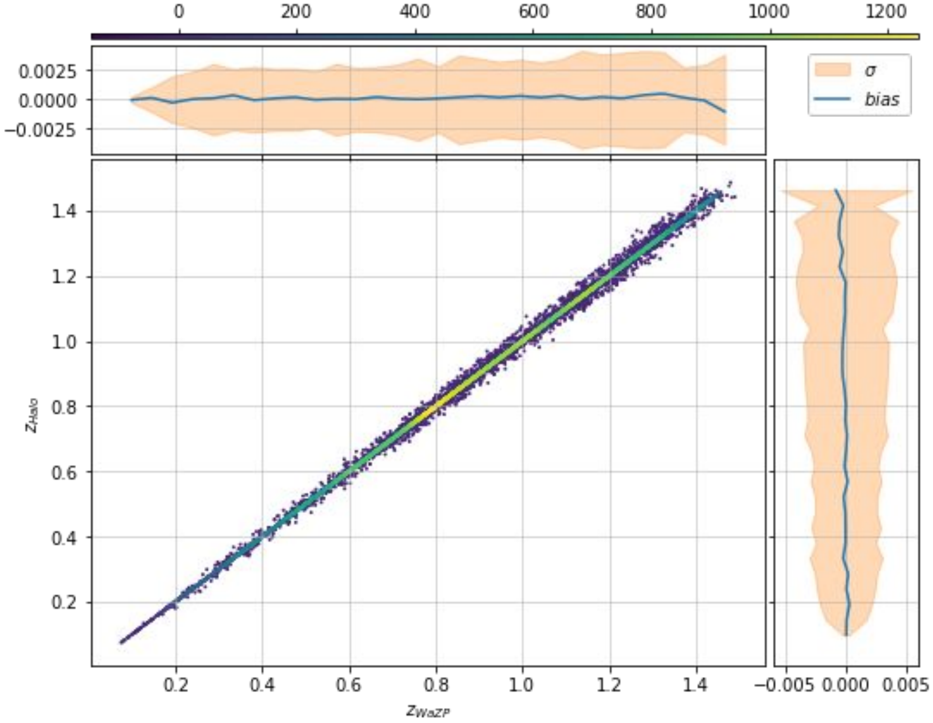


WaZP Cluster Finder on DC2

Redshift relation

i-band
z-band



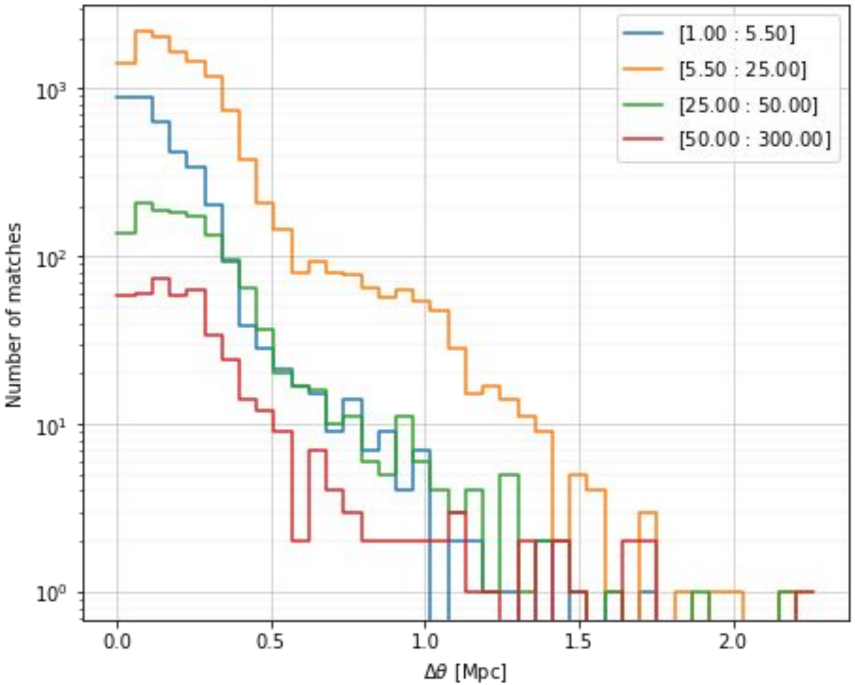


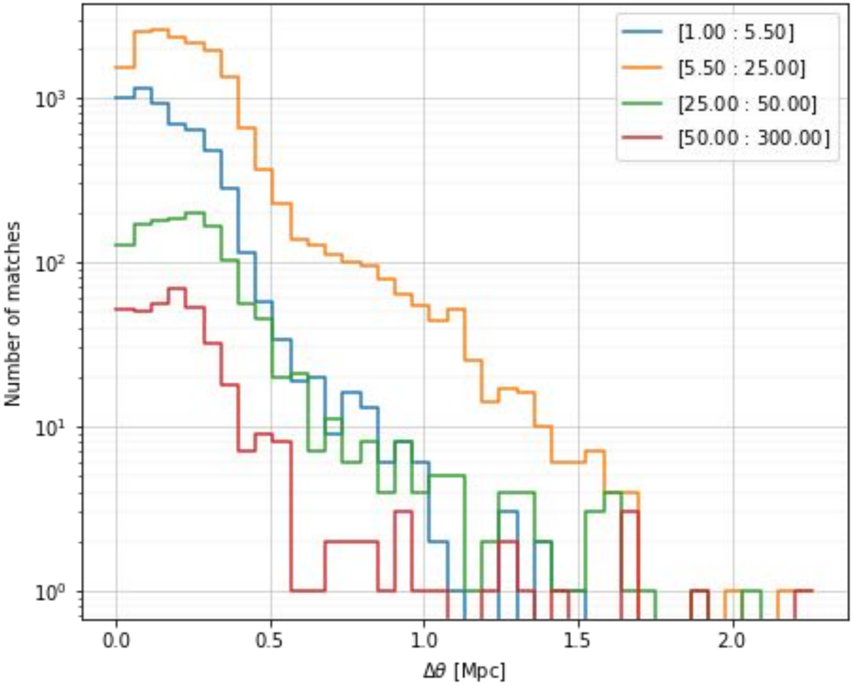
WaZP Cluster Finder on DC2

Redshift relation

Legacy Survey of Space and Time

i-band
z-band

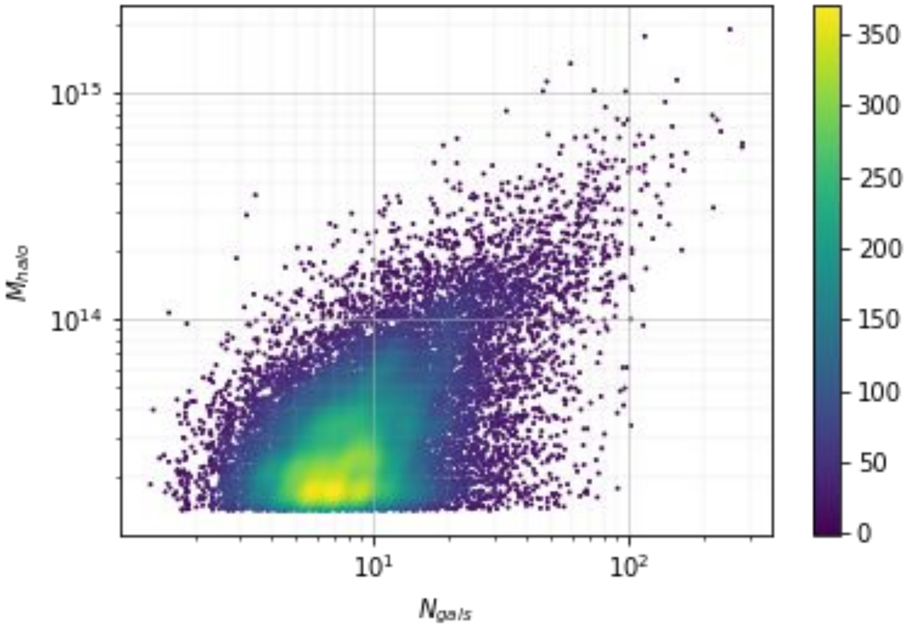


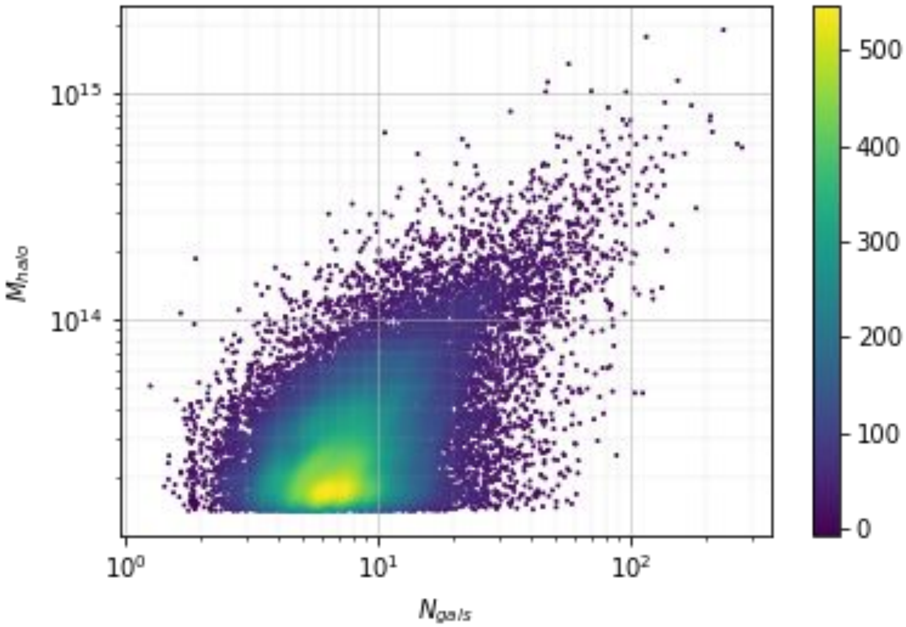


WaZP Cluster Finder on DC2

Mass-richness relation

i-band
z-band



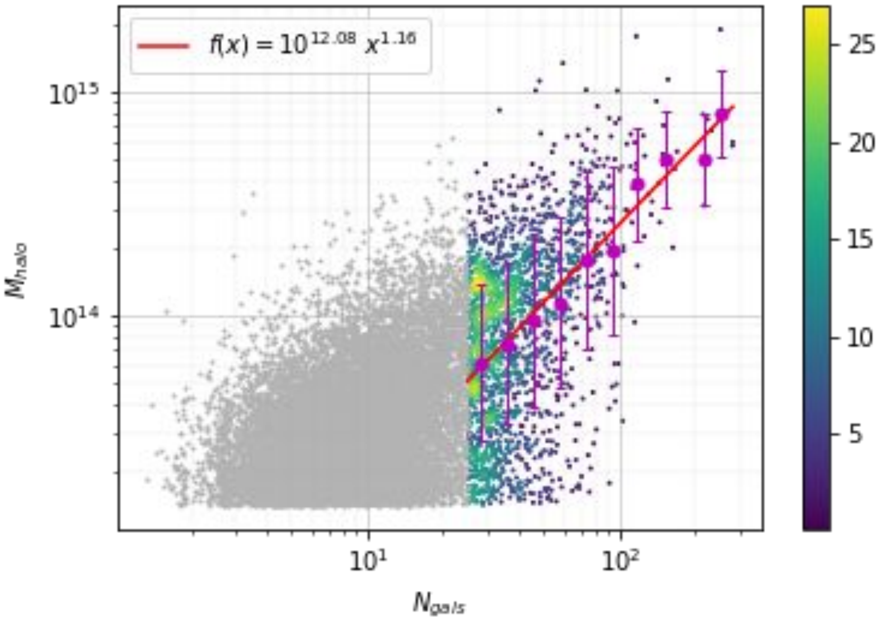


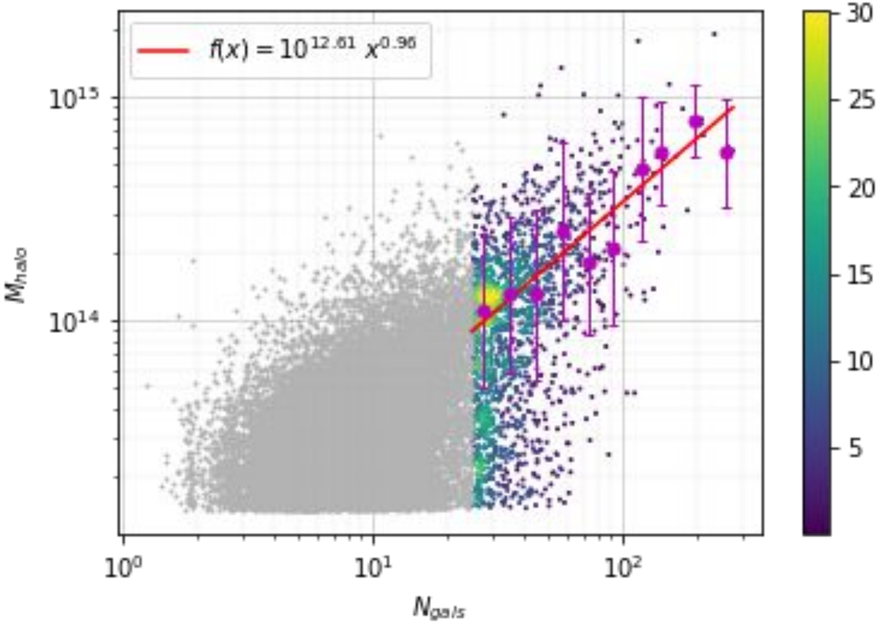
WaZP Cluster Finder on DC2

Mass-richness relation

Legacy Survey of Space and Time

i-band
z-band



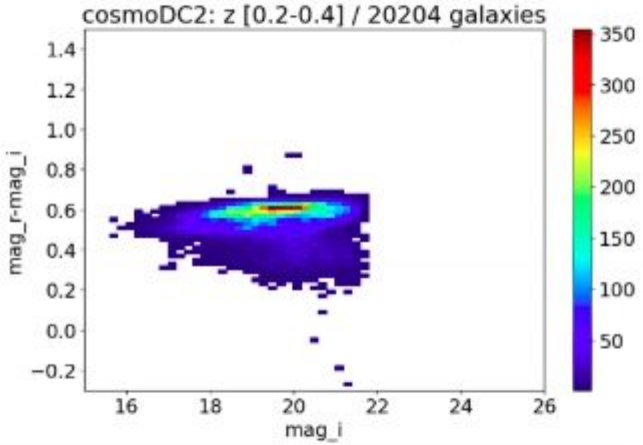


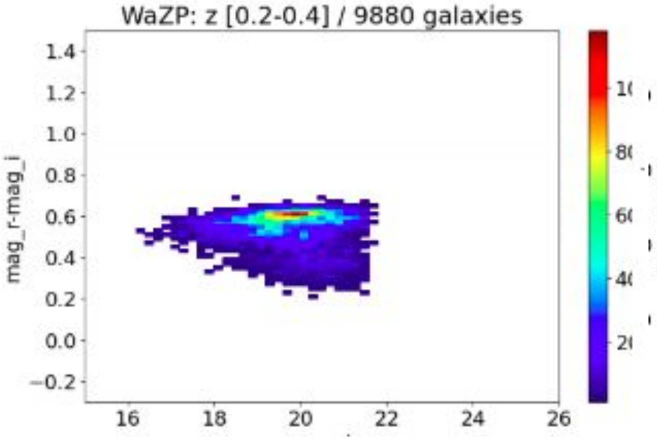
WaZP Cluster Finder on DC2

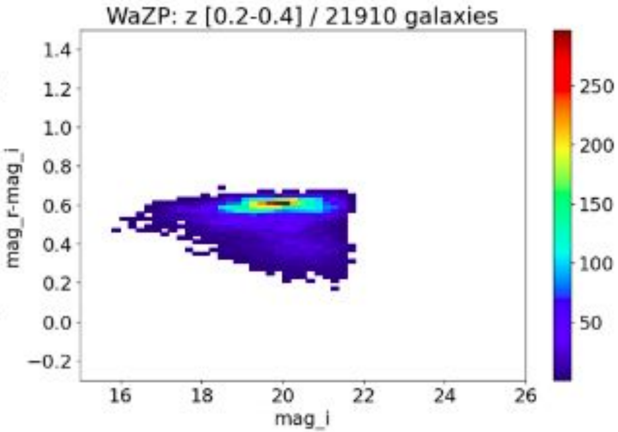
Colors of cluster members (Thibault Guillemin)

DM Halos
WaZP (DR3)

WaZP (Extragalactic)







WaZP Cluster Finder on DC2

Legacy Survey of Space and Time

- Future plans
  - Run on extragalactic with observed magnitudes and DESC photo-z
  - Run on DR3 with true magnitudes and DESC photo-z
  - Run on DR3 with observed magnitude and DESC photo-z
  - Run with photo-z computed by LIneA
  - Use full PDF for runs
  - Evaluate colors of cluster members - red sequence evolution (Thibault Guillemin)