

1.1_AKARI-NEP-OptNIR

March 8, 2018

1 AKARI-NEP master catalogue

1.1 Preparation of AKARI-NEP-OptNIR data

This product contains the 8-band (u^* , g' , r' , i' , z' , Y, J, Ks) optical to near-infrared catalogue from Oi et al., 2014.

- The identifier (it's unique in the catalogue);
- The position;
- The stellarity;
- The total magnitude for each band (u^* , g' , r' , i' , z' , Y, J, Ks).
- There are no aperture mags so we put empty columns in the masterlist.

We don't know when the maps have been observed. We will use the year of the reference paper.

This notebook was run with `herschelhelp_internal` version:
0246c5d (Thu Jan 25 17:01:47 2018 +0000) [with local modifications]

This notebook was executed on:
2018-02-15 20:31:07.859038

1.2 I - Column selection

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/ipykernel/__main__.py:8: R
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10
Check the NumPy 1.11 release notes for more information.
    ma.MaskedArray.__setitem__(self, index, value)
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/ipykernel/__main__.py:9: R
```

Out[6]: <IPython.core.display.HTML object>

1.3 II - Removal of duplicated sources

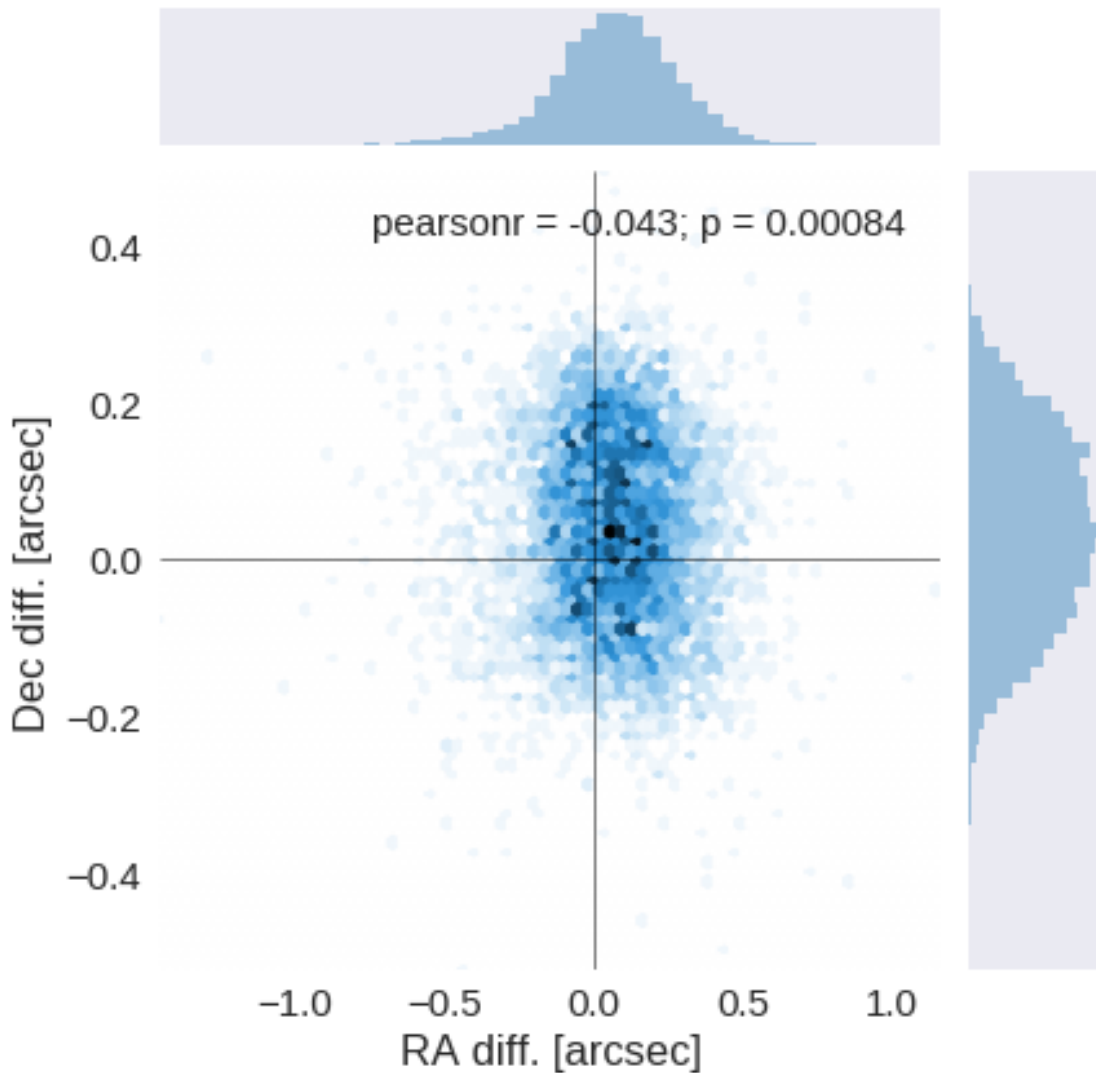
We remove duplicated objects from the input catalogues.

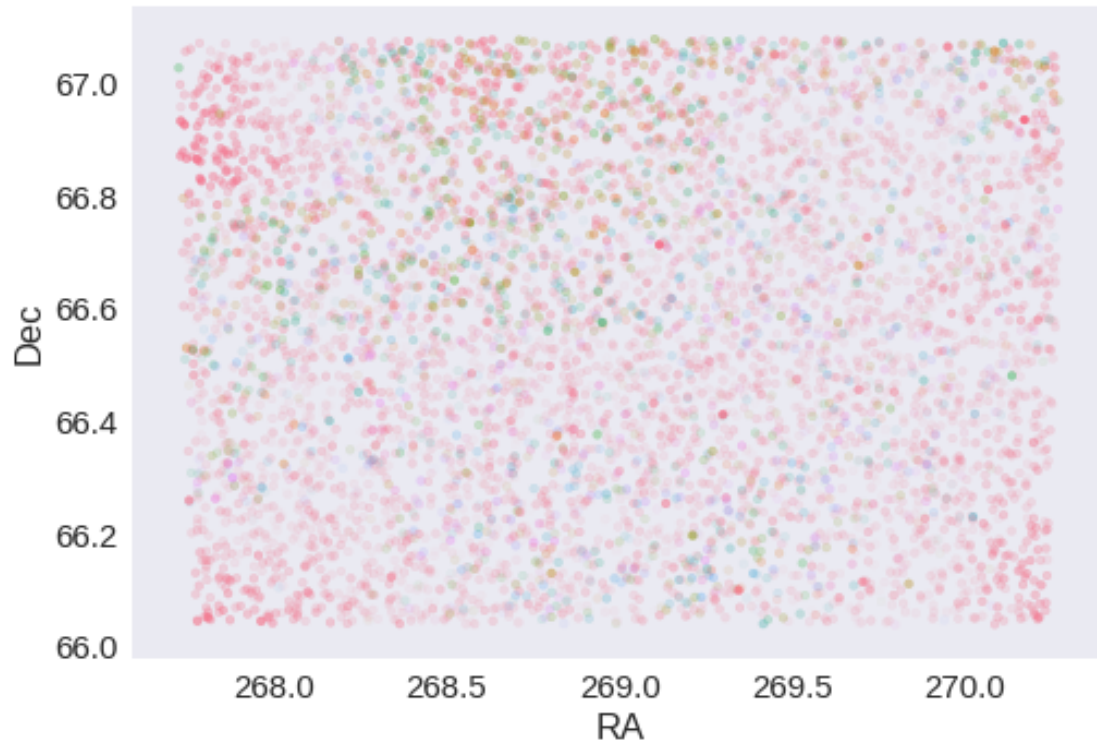
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10
Check the NumPy 1.11 release notes for more information.
ma.MaskedArray.__setitem__(self, index, value)
```

The initial catalogue had 85797 sources.
The cleaned catalogue has 85797 sources (0 removed).
The cleaned catalogue has 0 sources flagged as having been cleaned

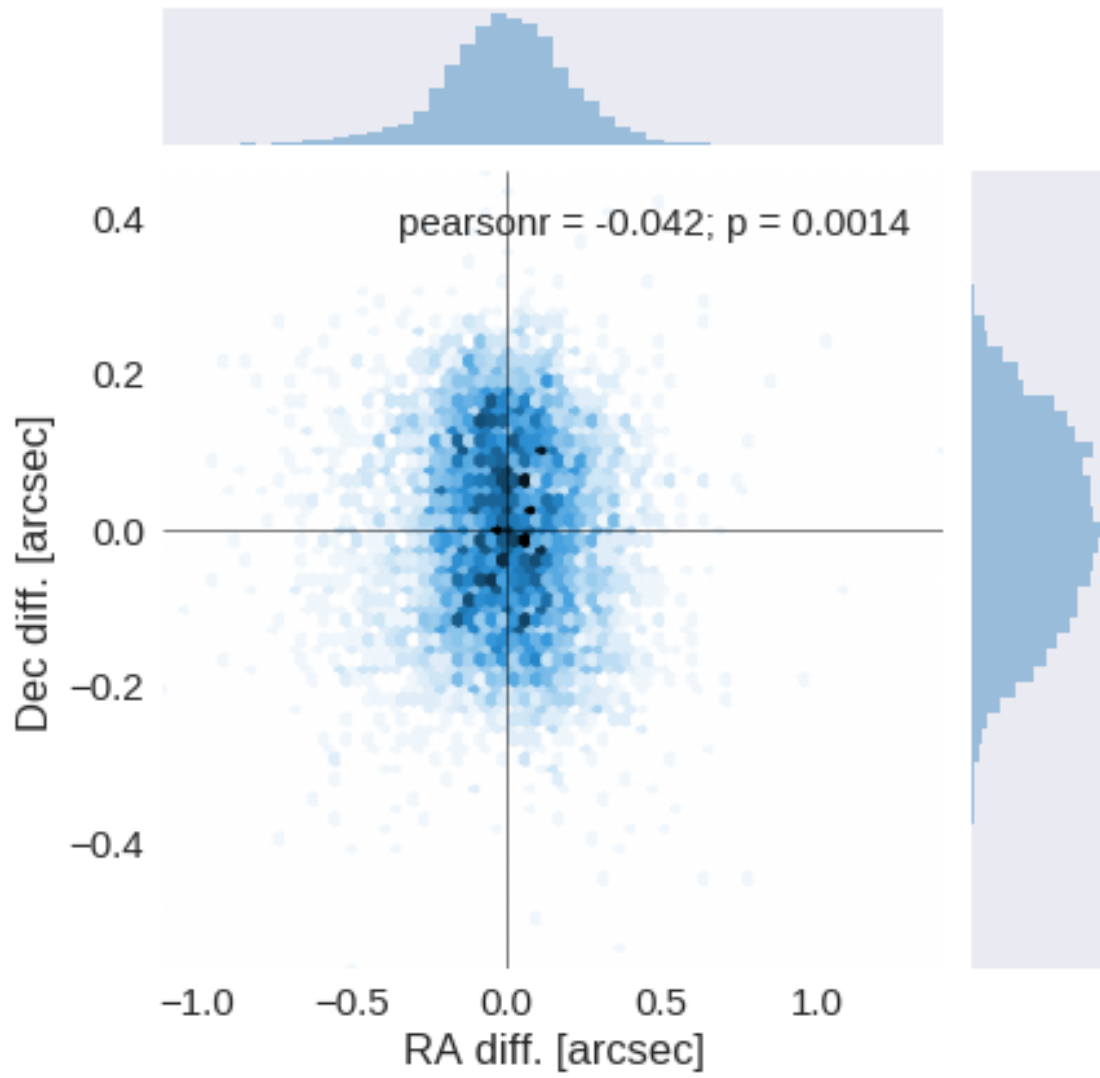
1.4 III - Astrometry correction

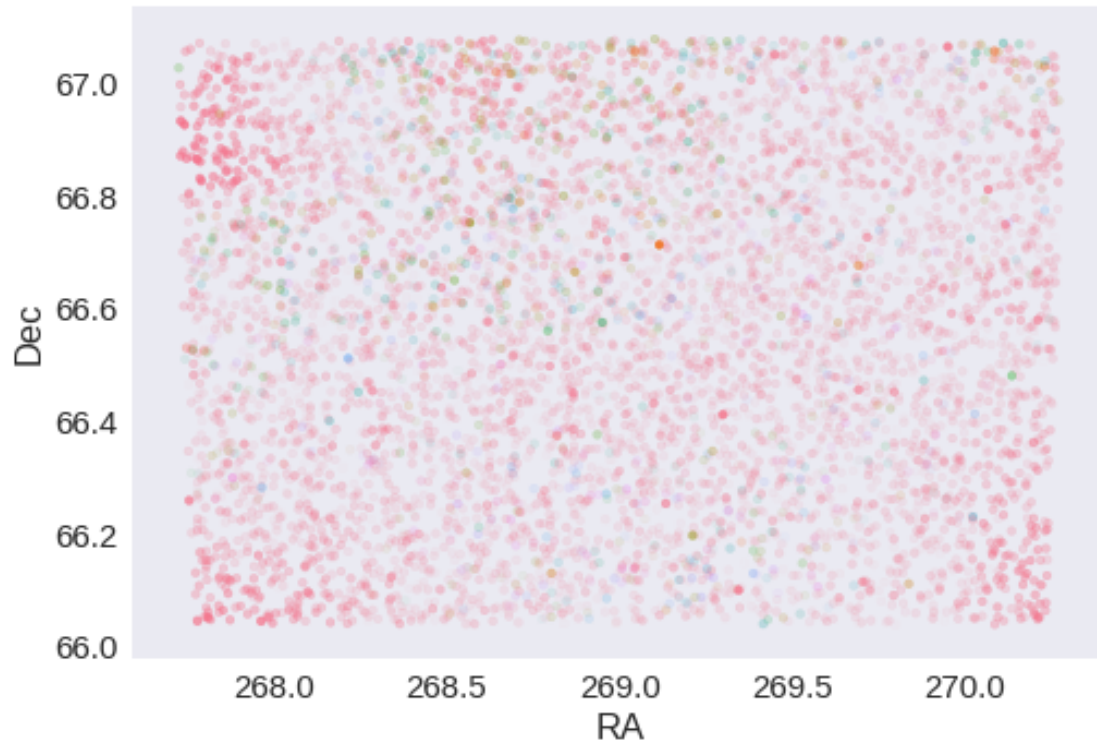
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.





RA correction: -0.07950606818667438 arcsec
Dec correction: -0.038150143166149064 arcsec





1.5 IV - Flagging Gaia objects

5953 sources flagged.

2 V - Saving to disk

1.2_PanSTARRS-3SS

March 8, 2018

1 AKARI-NEP master catalogue

1.1 Preparation of Pan-STARRS1 - 3pi Steradian Survey (3SS) data

This catalogue comes from `dmu0_PanSTARRS1-3SS`.

In the catalogue, we keep:

- The `uniquePspSTid` as unique object identifier;
- The r-band position which is given for all the sources;
- The grizy `<band>FApMag` aperture magnitude (see below);
- The grizy `<band>FKronMag` as total magnitude.

The Pan-STARRS1-3SS catalogue provides for each band an aperture magnitude defined as “In PS1, an ‘optimal’ aperture radius is determined based on the local PSF. The wings of the same analytic PSF are then used to extrapolate the flux measured inside this aperture to a ‘total’ flux.”

The observations used for the catalogue were done between 2010 and 2015 ([ref](#)).

TODO: Check if the detection flag can be used to know in which bands an object was detected to construct the coverage maps.

TODO: Check for stellarity.

This notebook was run with `herschelhelp_internal` version:
33f5ec7 (Wed Dec 6 16:56:17 2017 +0000)

1.2 I - Column selection

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10  
Check the NumPy 1.11 release notes for more information.  
ma.MaskedArray.__setitem__(self, index, value)
```

Out[6]: <IPython.core.display.HTML object>

1.3 II - Removal of duplicated sources

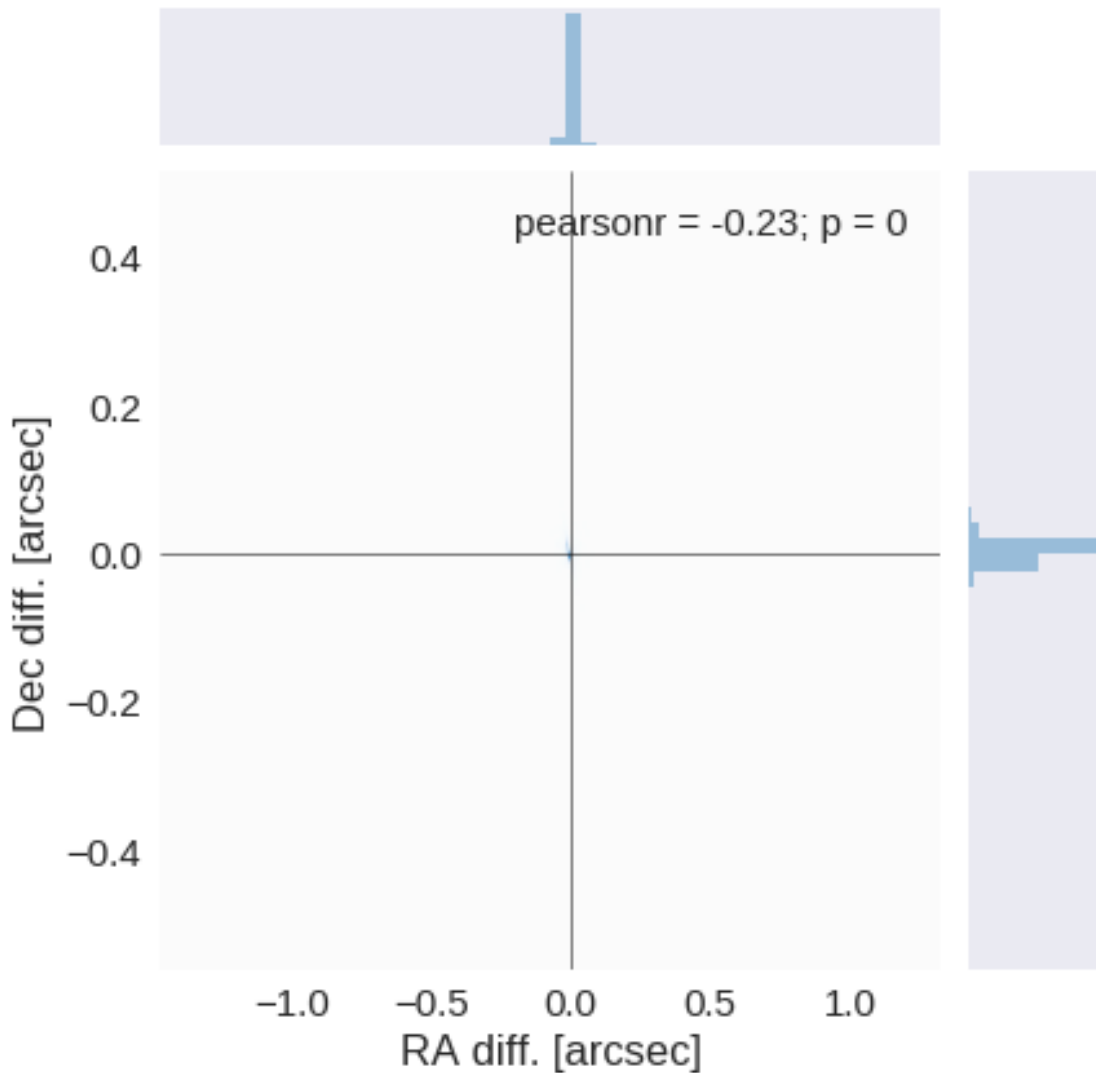
We remove duplicated objects from the input catalogues.

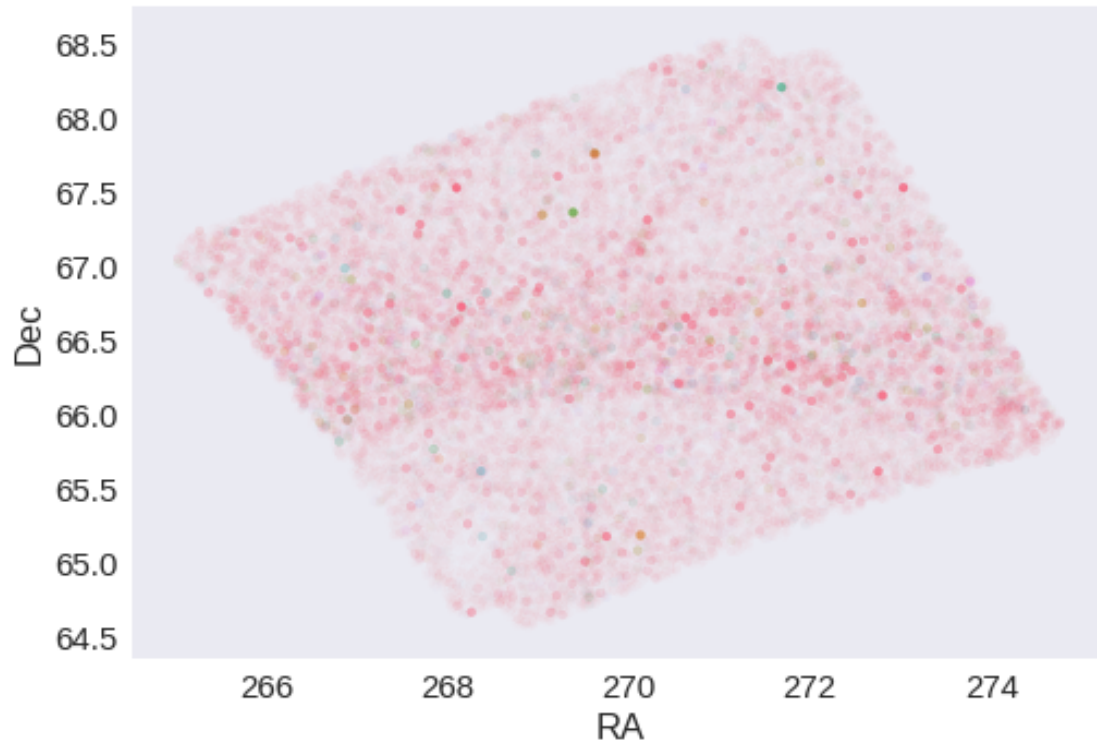
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10
Check the NumPy 1.11 release notes for more information.
ma.MaskedArray.__setitem__(self, index, value)
```

The initial catalogue had 221353 sources.
The cleaned catalogue has 221306 sources (47 removed).
The cleaned catalogue has 44 sources flagged as having been cleaned

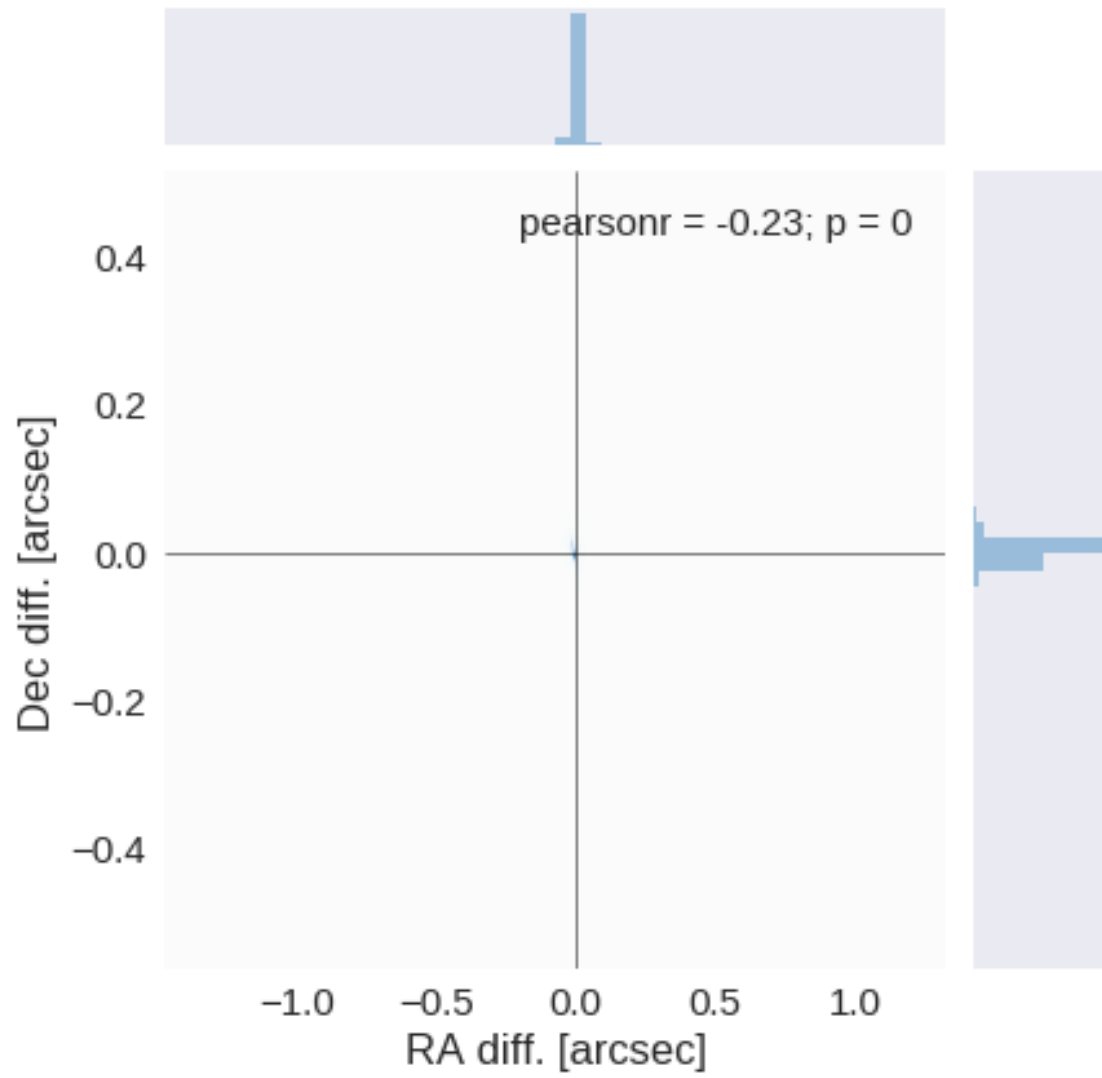
1.4 III - Astrometry correction

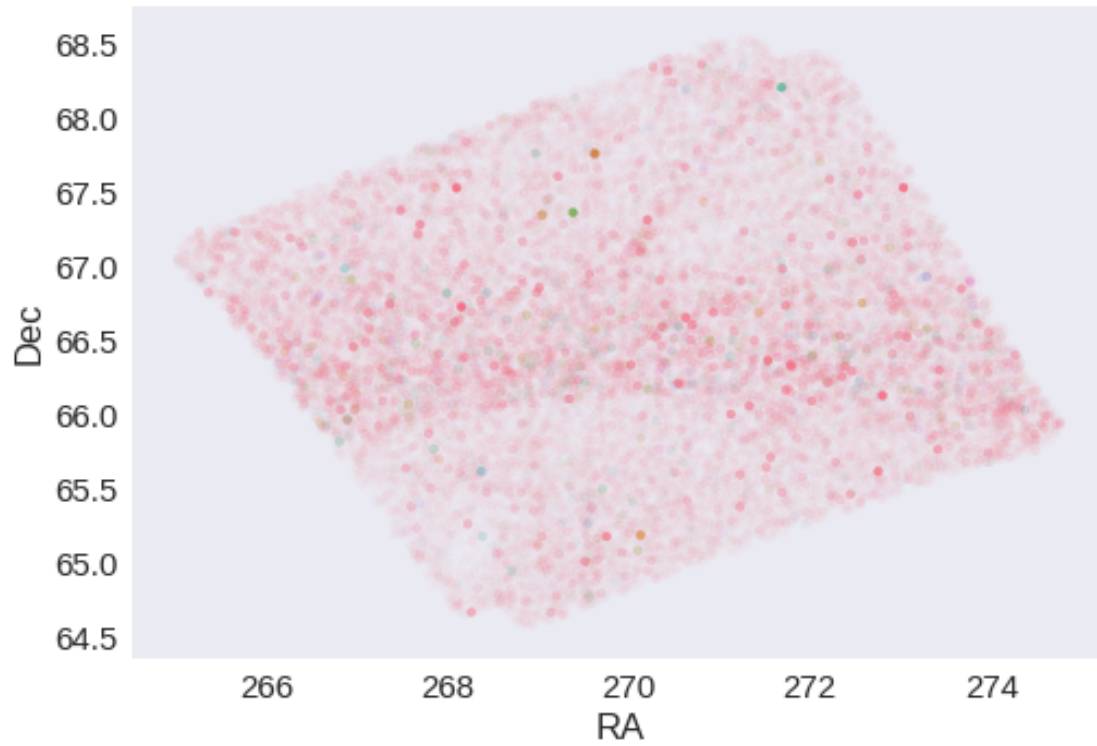
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.





RA correction: 0.00029263319447636604 arcsec
Dec correction: -0.001360161365937529 arcsec





1.5 IV - Flagging Gaia objects

61878 sources flagged.

2 V - Saving to disk

1.3_NEP-Spitzer

March 8, 2018

1 AKARI-NEP master catalogue

1.1 Preparation of Spitzer datafusion SERVS data

The Spitzer catalogues are available in `dmu0_NEP-Spitzer`.
In the catalogue, we keep:

- The internal identifier (this one is only in HeDaM data);
- The position;
- The fluxes in aperture 2 (1.9 arcsec); CHECK!
- The “auto” flux (which seems to be the Kron flux);
- The stellarity in each band

This notebook was run with `herschelhelp_internal` version:
33f5ec7 (Wed Dec 6 16:56:17 2017 +0000)

1.2 I - Column selection

`Out [6]: <IPython.core.display.HTML object>`

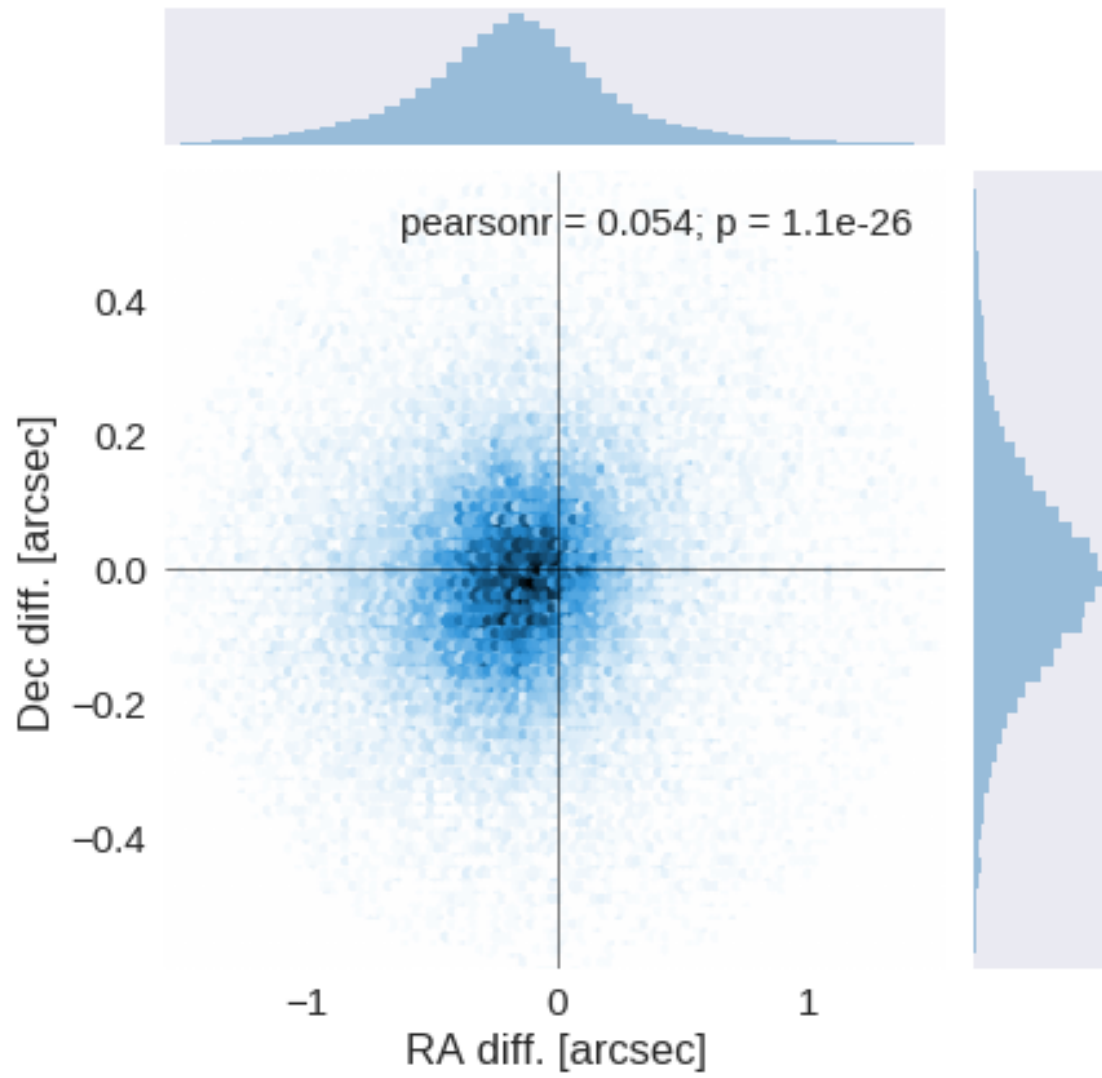
1.3 II - Removal of duplicated sources

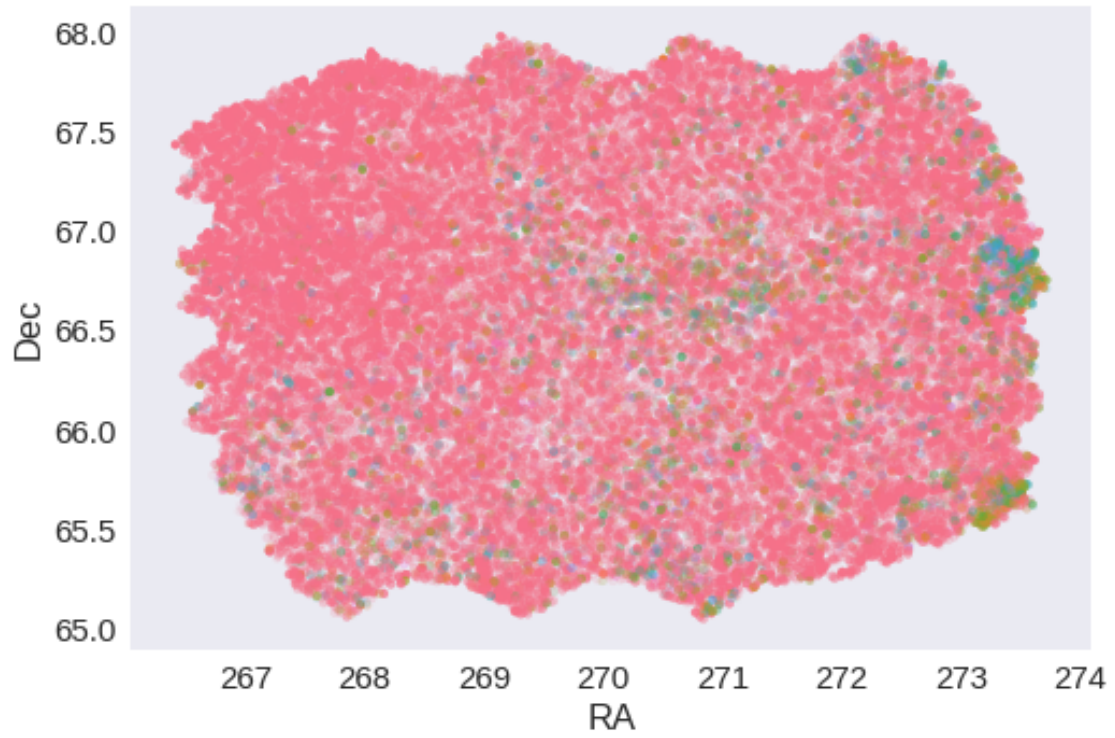
We remove duplicated objects from the input catalogues.

The initial catalogue had 380858 sources.
The cleaned catalogue has 380858 sources (0 removed).
The cleaned catalogue has 0 sources flagged as having been cleaned

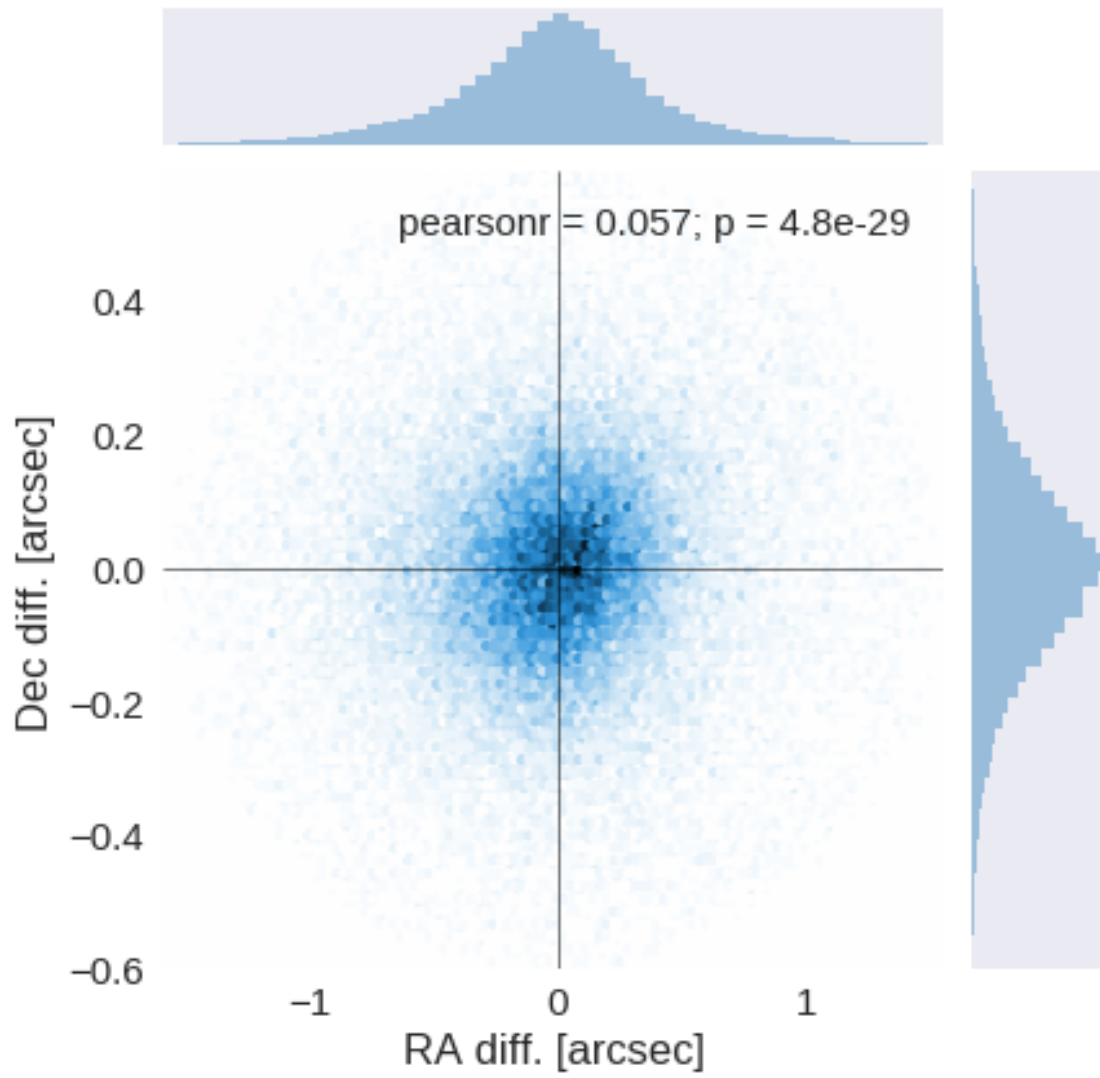
1.4 III - Astrometry correction

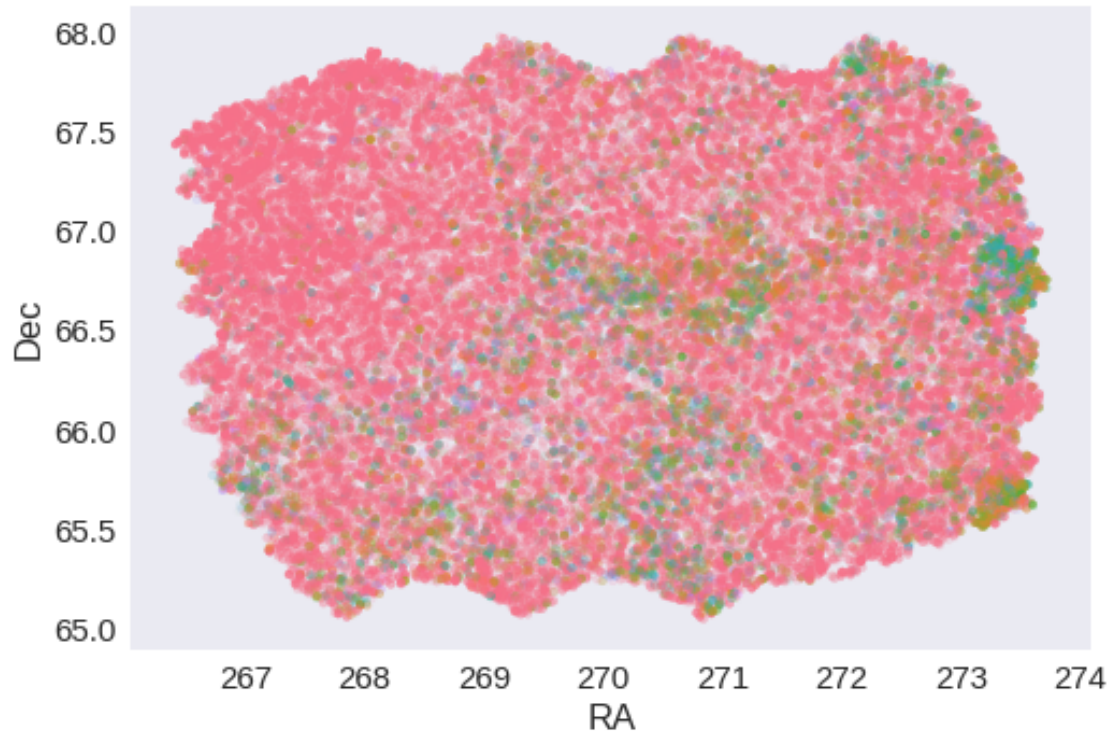
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.





RA correction: 0.17224358761041003 arcsec
Dec correction: 0.017100100612310598 arcsec





1.5 IV - Flagging Gaia objects

44067 sources flagged.

1.6 V - Saving to disk

2_Merging

March 8, 2018

1 AKARI-NEP master catalogue

This notebook presents the merge of the various pristine catalogues to produce HELP mater catalogue on AKARI-NEP.

```
This notebook was run with herschelhelp_internal version:  
0246c5d (Thu Jan 25 17:01:47 2018 +0000) [with local modifications]  
This notebook was executed on:  
2018-02-15 20:31:42.157732
```

1.1 I - Reading the prepared pristine catalogues

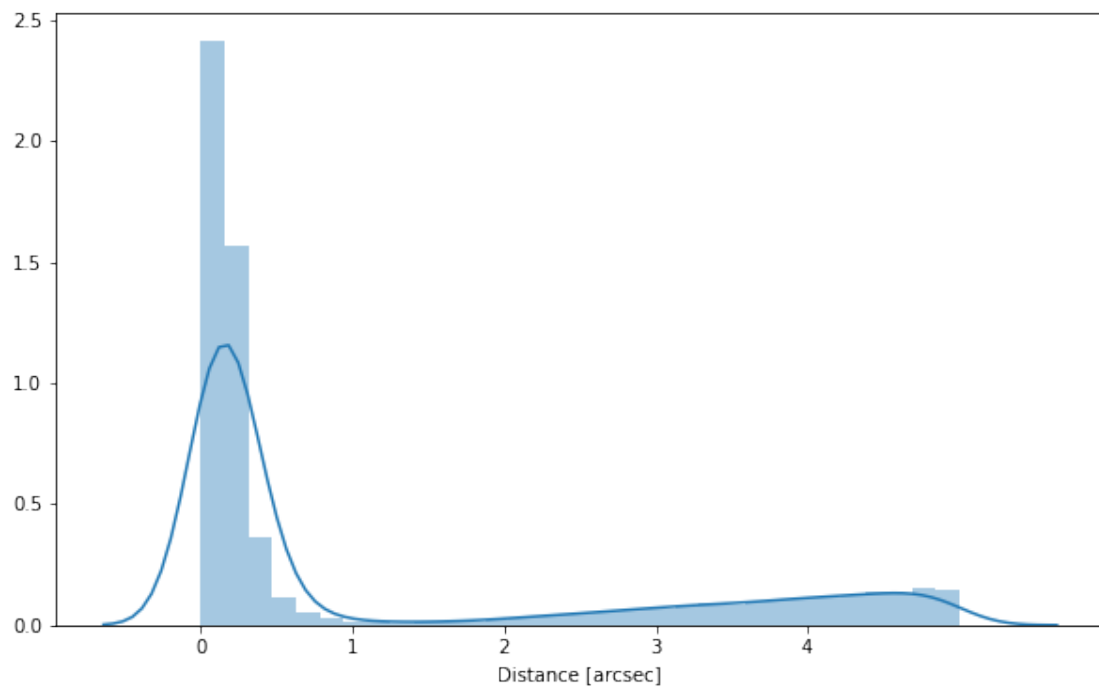
1.2 II - Merging tables

We first merge the optical catalogues and then add the infrared ones: WFC, DXS, SpARCS, HSC, PS1, SERVS, SWIRE.

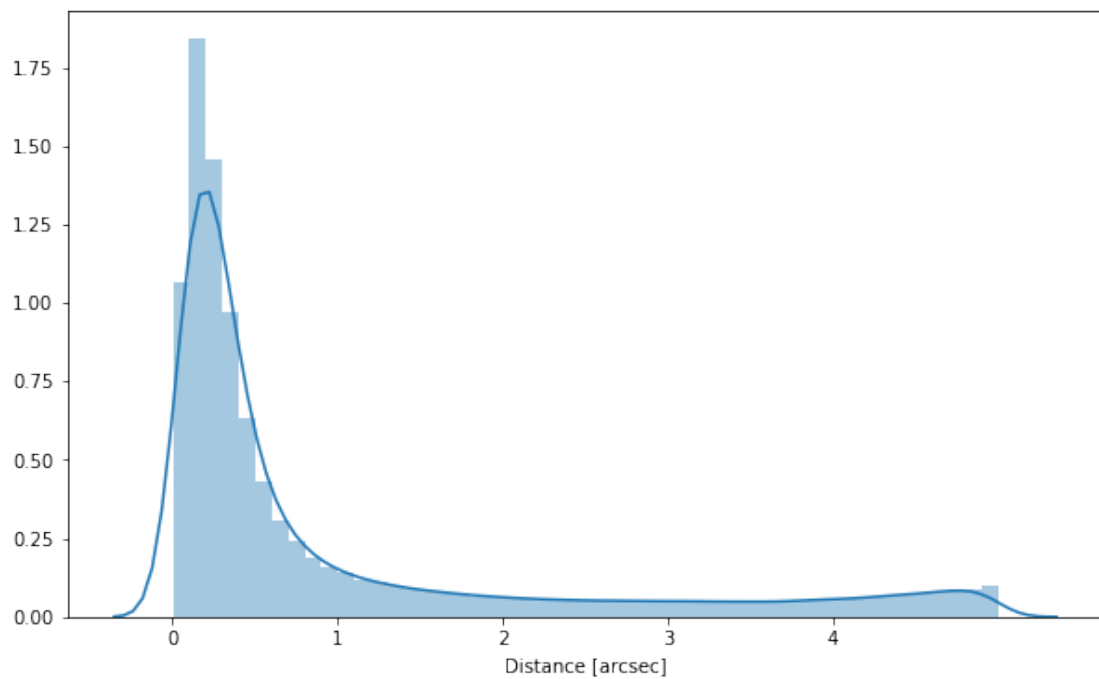
At every step, we look at the distribution of the distances to the nearest source in the merged catalogue to determine the best crossmatching radius.

1.2.1 PanSTARRS

1.2.2 Add AKARI



1.2.3 Add NEP-Spitzer



1.2.4 Cleaning

When we merge the catalogues, astropy masks the non-existent values (e.g. when a row comes only from a catalogue and has no counterparts in the other, the columns from the latest are masked for that row). We indicate to use NaN for masked values for floats columns, False for flag columns and -1 for ID columns.

```
Out[11]: <IPython.core.display.HTML object>
```

1.3 III - Merging flags and stellerity

Each pristine catalogue contains a flag indicating if the source was associated to a another nearby source that was removed during the cleaning process. We merge these flags in a single one.

Each pristine catalogue contains a flag indicating the probability of a source being a Gaia object (0: not a Gaia object, 1: possibly, 2: probably, 3: definitely). We merge these flags taking the highest value.

Each prisitine catalogue may contain one or several stellerity columns indicating the probability (0 to 1) of each source being a star. We merge these columns taking the highest value. We keep trace of the origin of the stellerity.

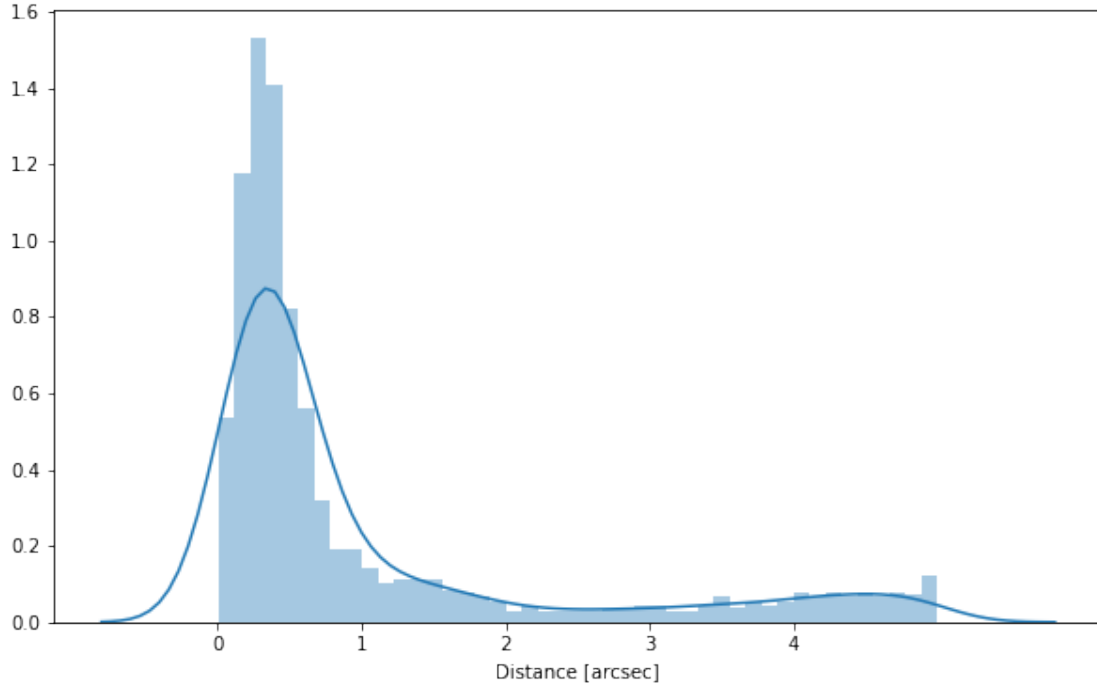
```
akari_stellariry, irac_stellariry
```

1.4 IV - Adding E(B-V) column

1.5 V.a - Adding HELP unique identifiers and field columns

OK!

1.6 V.b - Add Specz



1.7 VIII.a Wavelength domain coverage

We add a binary flag `flag_optnir_obs` indicating that a source was observed in a given wavelength domain:

- 1 for observation in optical;
- 2 for observation in near-infrared;
- 4 for observation in mid-infrared (IRAC).

It's an integer binary flag, so a source observed both in optical and near-infrared but not in mid-infrared would have this flag at $1 + 2 = 3$.

Note 1: The observation flag is based on the creation of multi-order coverage maps from the catalogues, this may not be accurate, especially on the edges of the coverage.

Note 2: Being on the observation coverage does not mean having fluxes in that wavelength domain. For sources observed in one domain but having no flux in it, one must take into consideration the different depths in the catalogue we are using.

1.8 VIII.b Wavelength domain detection

We add a binary flag `flag_optnir_det` indicating that a source was detected in a given wavelength domain:

- 1 for detection in optical;

- 2 for detection in near-infrared;
- 4 for detection in mid-infrared (IRAC).

It's an integer binary flag, so a source detected both in optical and near-infrared by not in mid-infrared would have this flag at $1 + 2 = 3$.

Note 1: We use the total flux columns to know if the source has flux, in some catalogues, we may have aperture flux and no total flux.

To get rid of artefacts (chip edges, star flares, etc.) we consider that a source is detected in one wavelength domain when it has a flux value in **at least two bands**. That means that good sources will be excluded from this flag when they are on the coverage of only one band.

1.9 IX - Cross-identification table

We are producing a table associating to each HELP identifier, the identifiers of the sources in the pristine catalogues. This can be used to easily get additional information from them.

For convenience, we also cross-match the master list with the SDSS catalogue and add the objID associated with each source, if any. **TODO: should we correct the astrometry with respect to Gaia positions?**

33 master list rows had multiple associations.

```
['ps1_id', 'akari_id', 'nep_id', 'help_id', 'specz_id', 'sdss_id']
```

1.10 X - Adding HEALPix index

We are adding a column with a HEALPix index at order 13 associated with each source.

1.11 XI - Saving the catalogue

Missing columns: {'flag_megacam_u', 'flag_megacam_i', 'flag_wircam_ks', 'flag_gpc1_z', 'flag_wir

3_Checks_and_diagnostics

March 8, 2018

1 AKARI-NEP master catalogue

1.1 Checks and diagnostics

This notebook was run with `herschelhelp_internal` version:
0246c5d (Thu Jan 25 17:01:47 2018 +0000) [with local modifications]
This notebook was executed on:
2018-02-15 20:56:51.894212

Diagnostics done using: `master_catalogue_akari-nep_20180215.fits`

1.2 0 - Quick checks

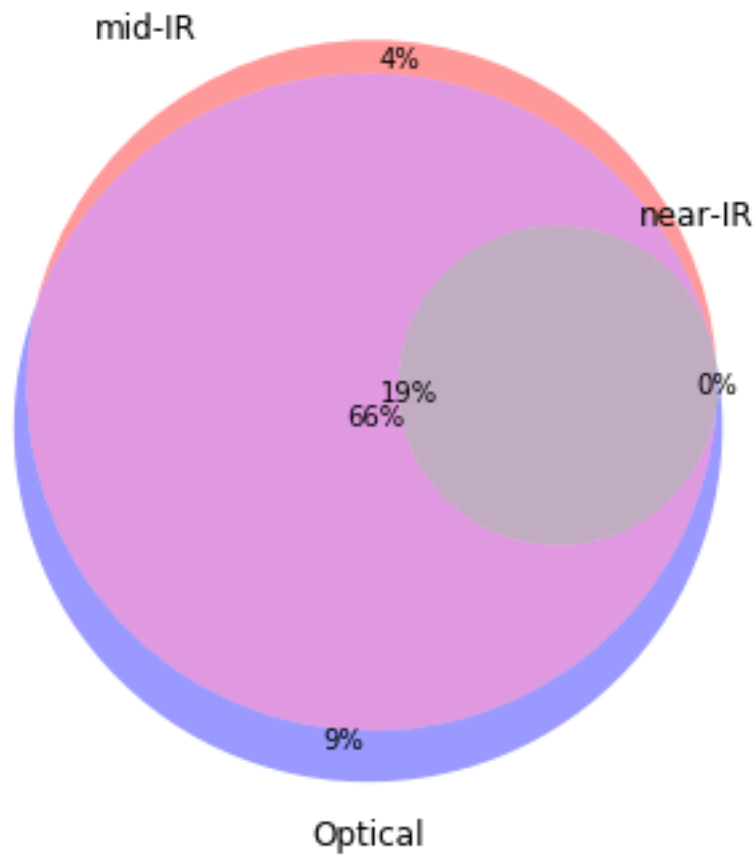
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/core/numeric.py:301:
  format(shape, fill_value, array(fill_value).dtype), FutureWarning)
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/core/numeric.py:301:
  format(shape, fill_value, array(fill_value).dtype), FutureWarning)
```

Table shows only problematic columns.

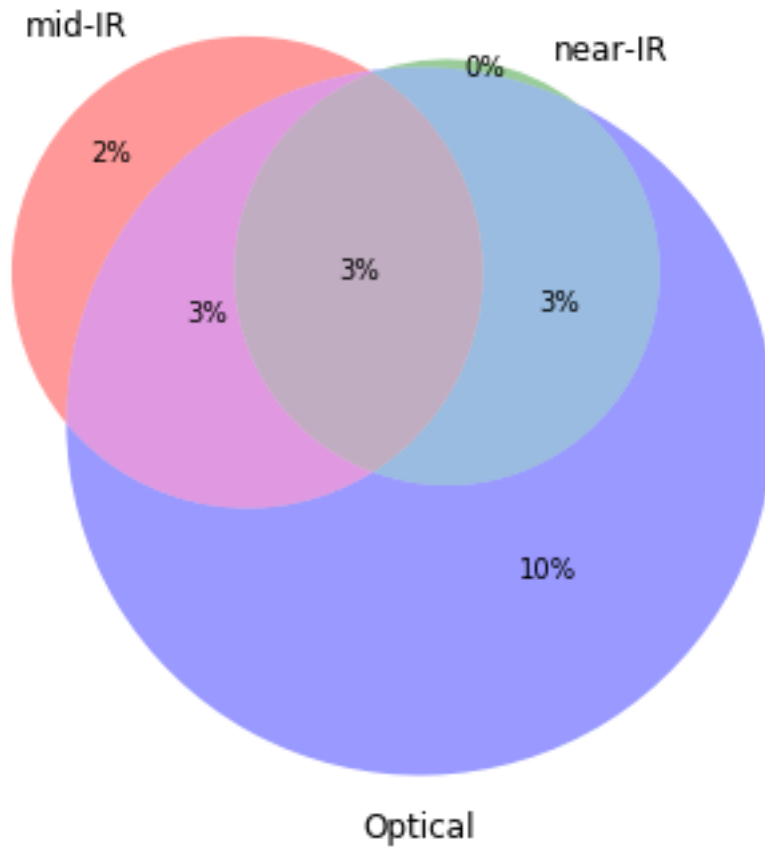
`Out[4]:` <IPython.core.display.HTML object>

1.3 I - Summary of wavelength domains

Wavelength domain observations



Detection of the 418,429 sources detected
in any wavelength domains (among 531,746 sources)

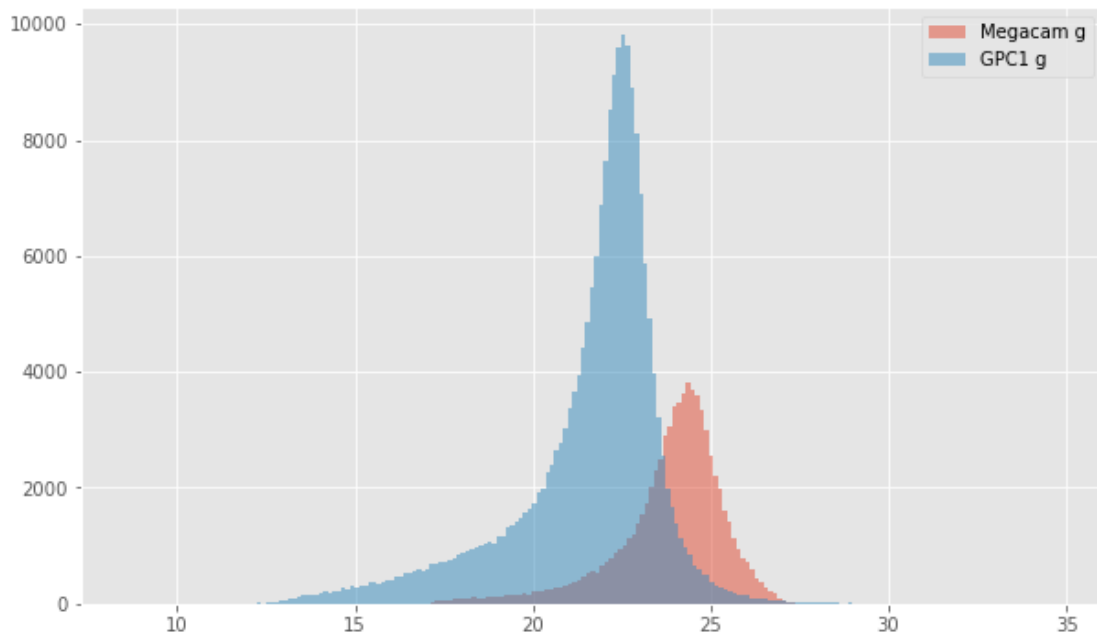
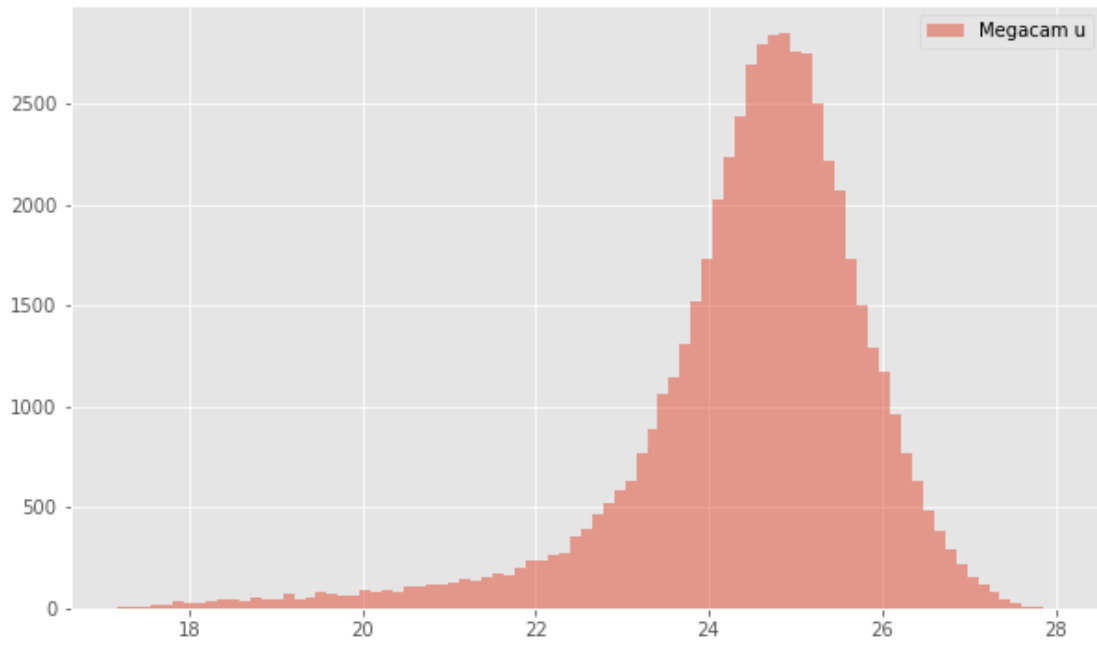


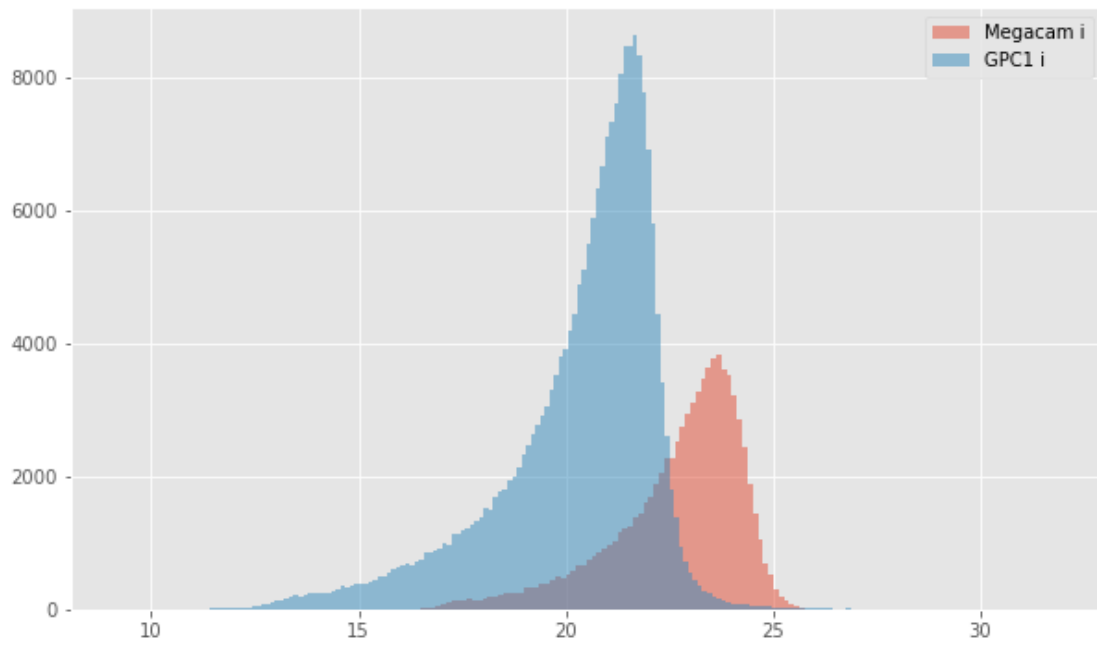
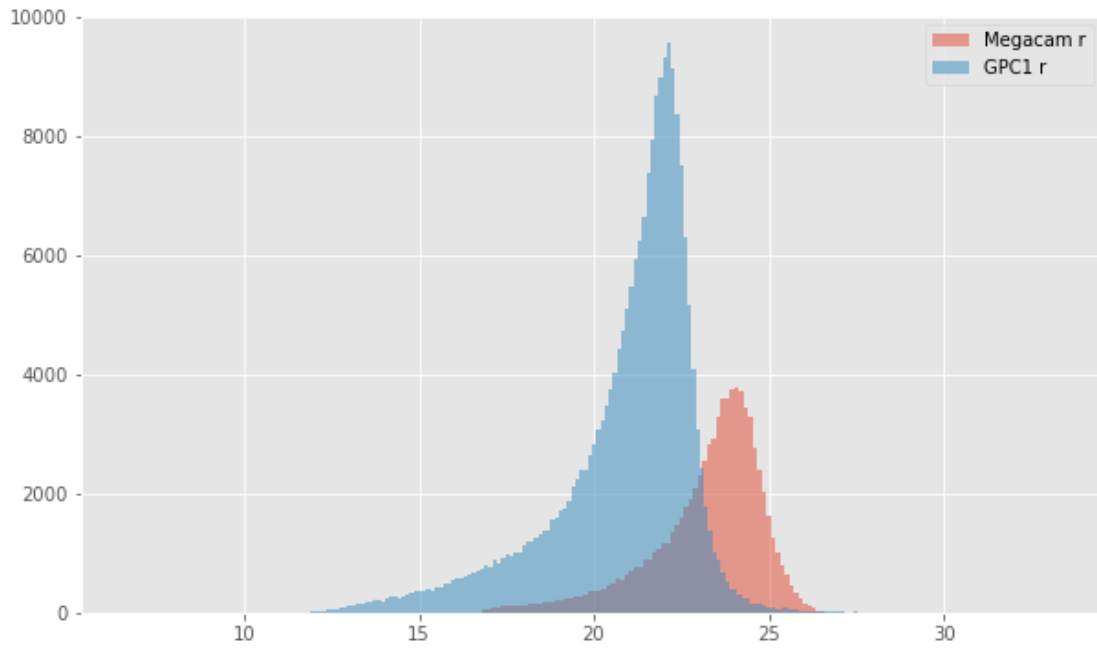
1.4 II - Comparing magnitudes in similar filters

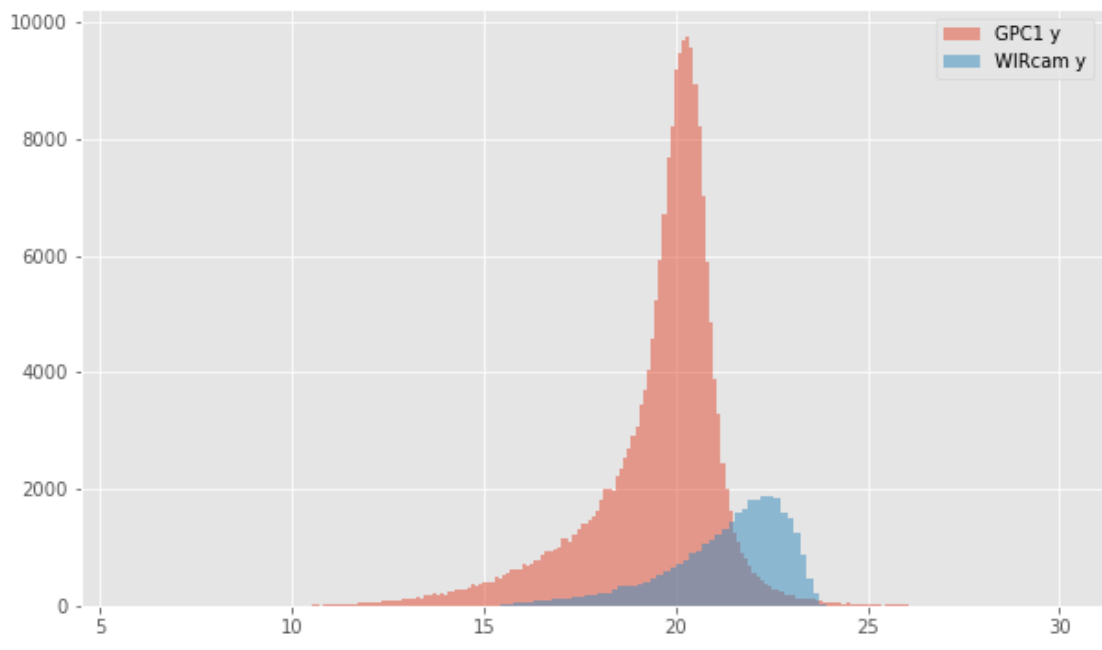
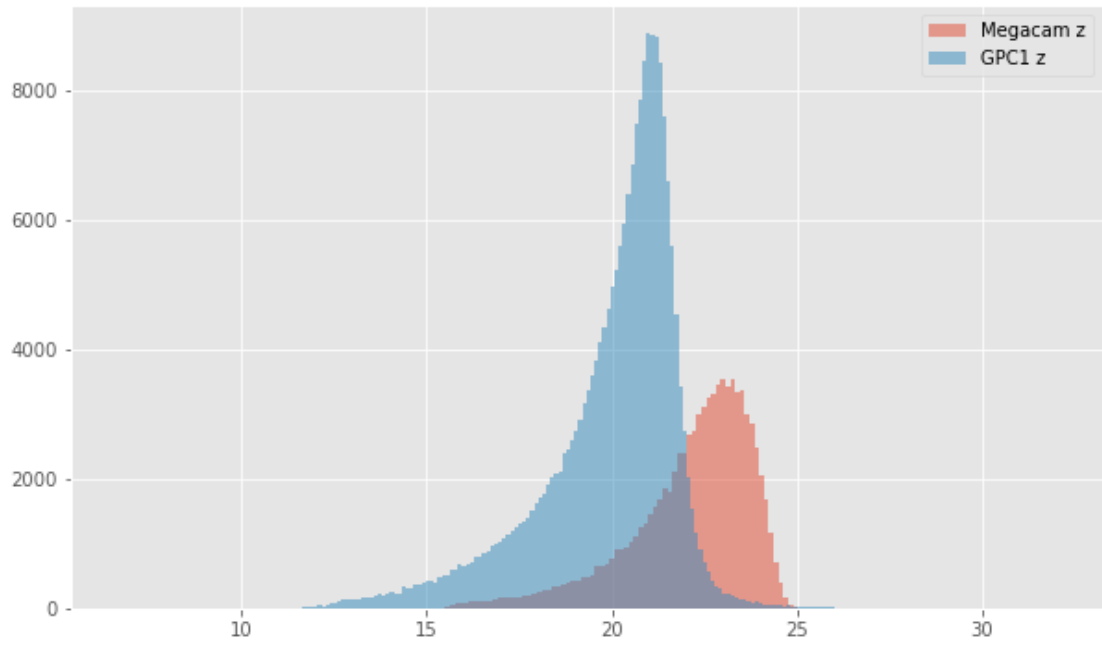
The master list is composed of several catalogues containing magnitudes in similar filters on different instruments. We are comparing the magnitudes in these corresponding filters.

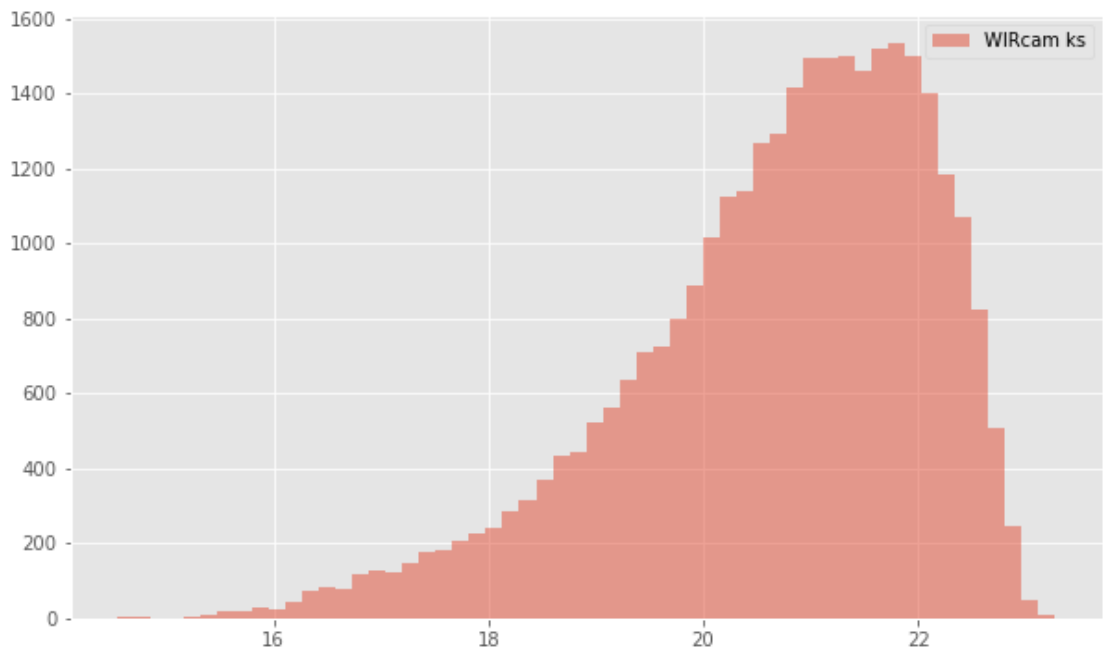
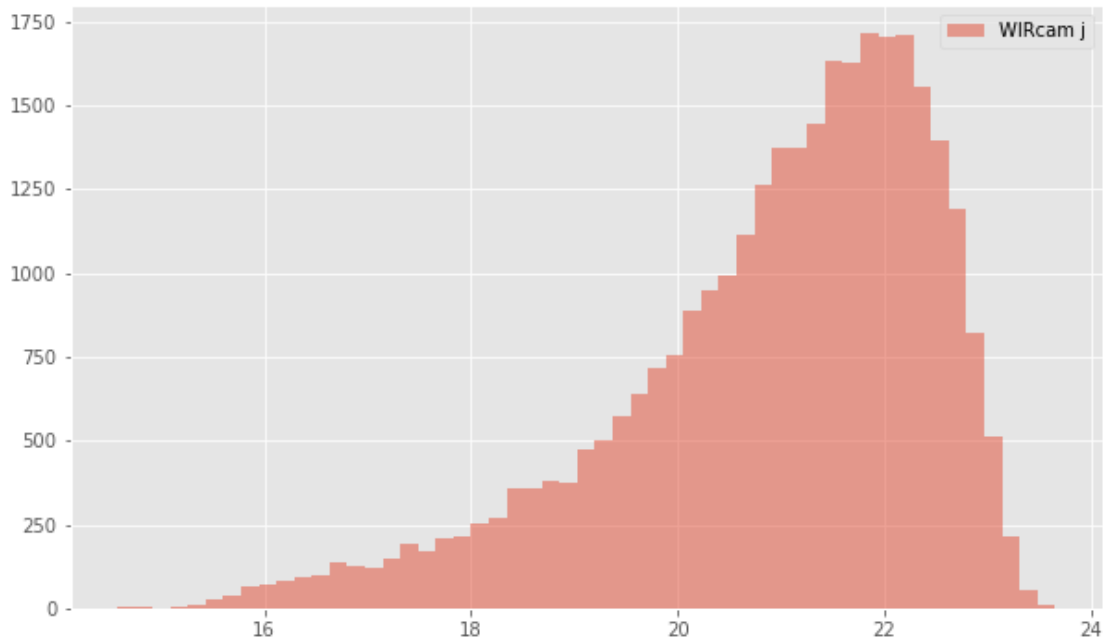
1.4.1 II.a - Comparing depths

We compare the histograms of the total aperture magnitudes of similar bands.







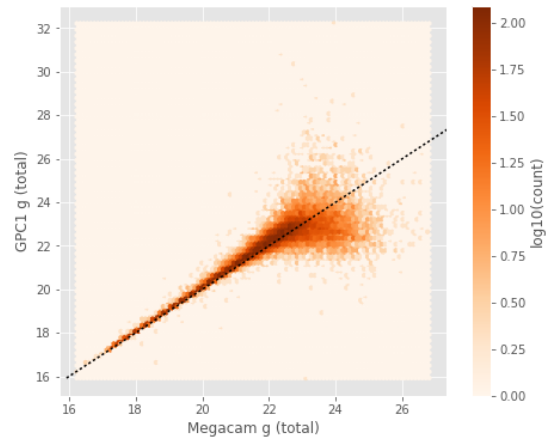
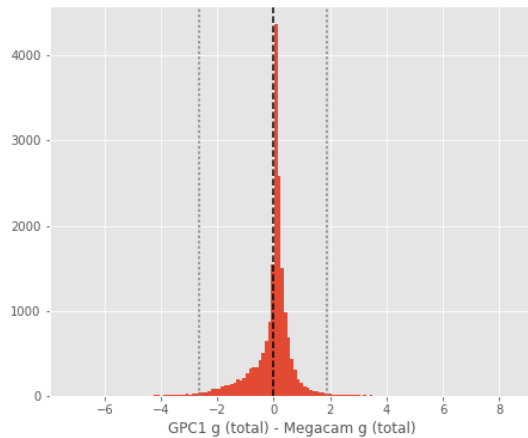


1.4.2 II.b - Comparing magnitudes

We compare one to one each magnitude in similar bands.

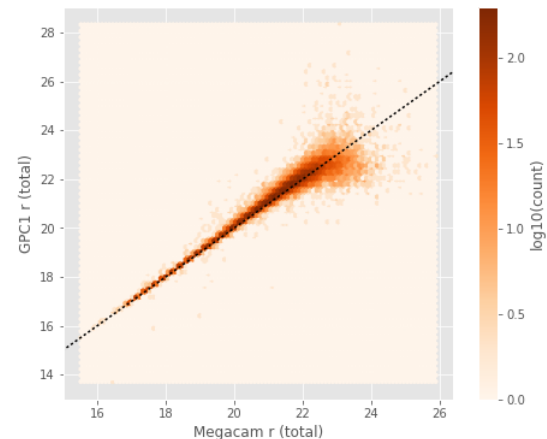
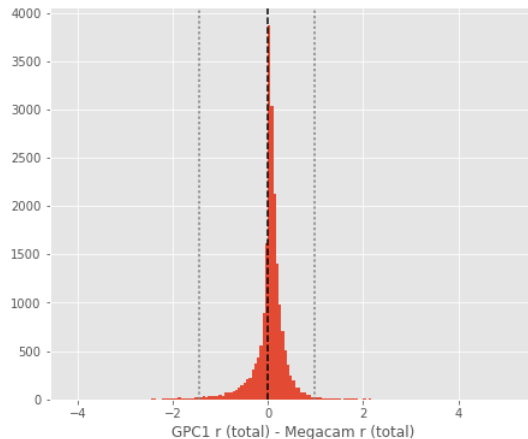
GPC1 g (total) - Megacam g (total):

- Median: 0.08
- Median Absolute Deviation: 0.21
- 1% percentile: -2.629887466430664
- 99% percentile: 1.8692850494384783



GPC1 r (total) - Megacam r (total):

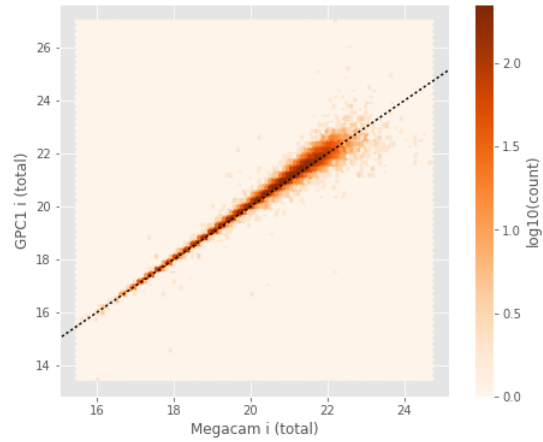
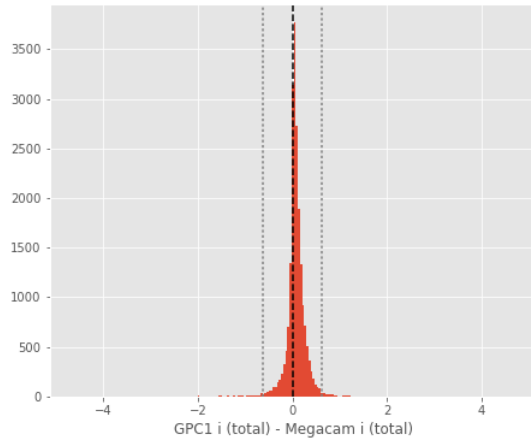
- Median: 0.06
- Median Absolute Deviation: 0.10
- 1% percentile: -1.4476290702819823
- 99% percentile: 0.9900200843811018



GPC1 i (total) - Megacam i (total):

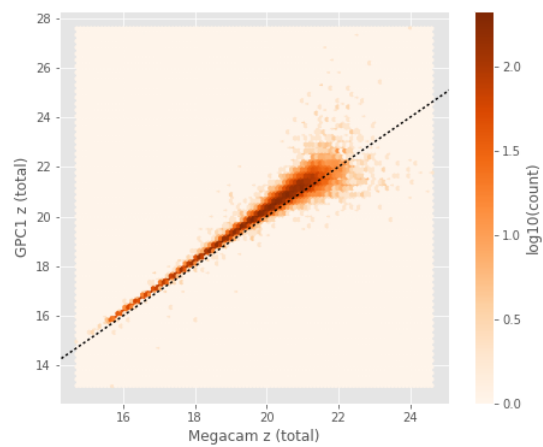
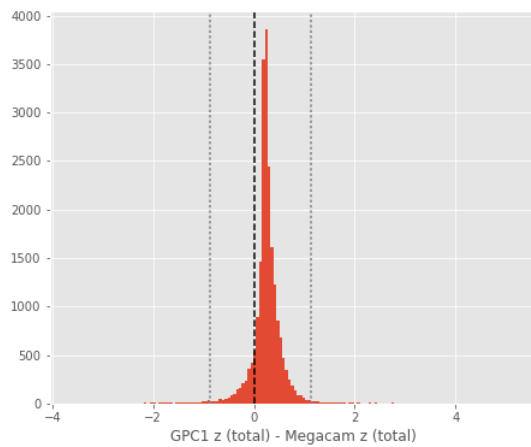
- Median: 0.06

- Median Absolute Deviation: 0.07
- 1% percentile: -0.6291459655761719
- 99% percentile: 0.6203815078735347



GPC1 z (total) - Megacam z (total):

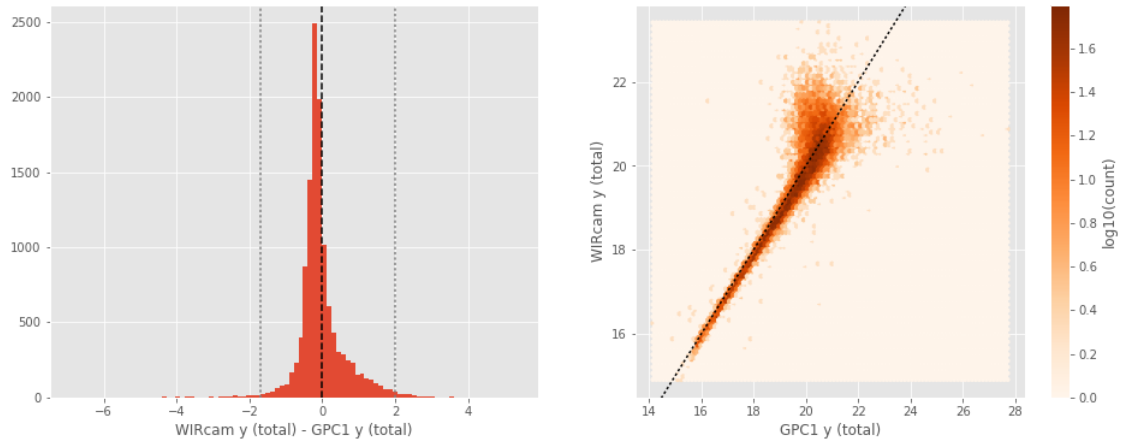
- Median: 0.24
- Median Absolute Deviation: 0.10
- 1% percentile: -0.8746409225463867
- 99% percentile: 1.1230560874938962



WIRcam y (total) - GPC1 y (total):

- Median: -0.14
- Median Absolute Deviation: 0.20
- 1% percentile: -1.7099049377441407

- 99% percentile: 1.996623935699459



1.5 III - Comparing magnitudes to reference bands

Cross-match the master list to SDSS and 2MASS to compare its magnitudes to SDSS and 2MASS ones.

1.5.1 III.a - Comparing u, g, r, i, and z bands to SDSS

The catalogue is cross-matched to SDSS-DR13 withing 0.2 arcsecond.

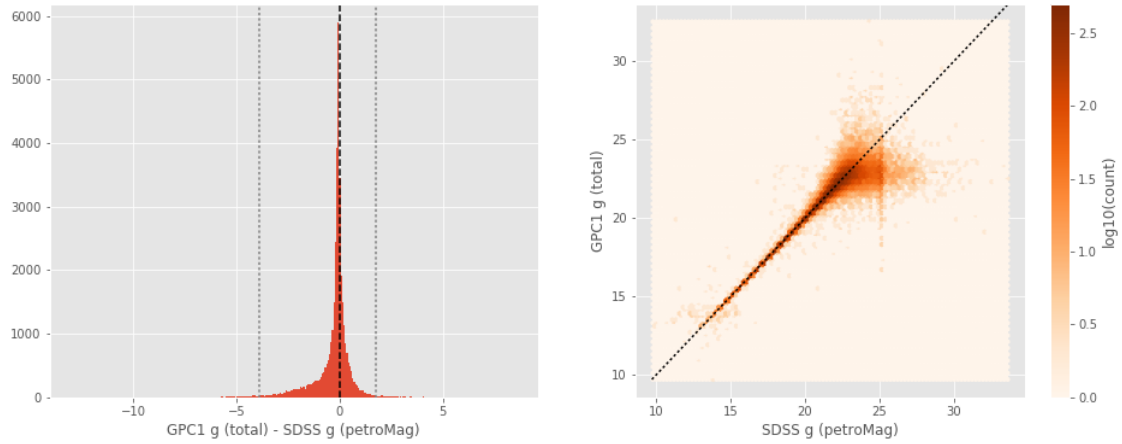
We compare the u, g, r, i, and z magnitudes to those from SDSS using `fiberMag` for the aperture magnitude and `petroMag` for the total magnitude.

No sources have both SDSS u (`petroMag`) and Megacam u (`total`) values.

No sources have both SDSS g (`petroMag`) and Megacam g (`total`) values.

GPC1 g (`total`) - SDSS g (`petroMag`):

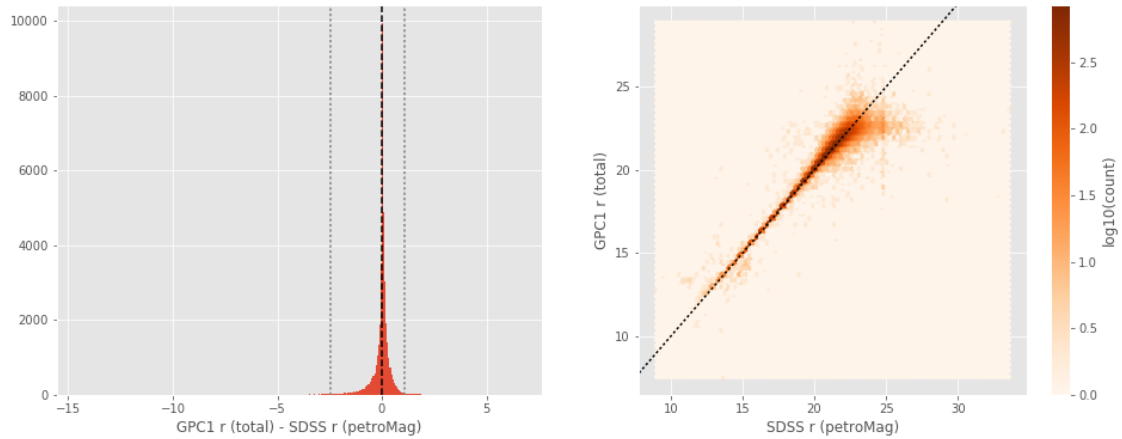
- Median: -0.09
- Median Absolute Deviation: 0.20
- 1% percentile: -3.9173999786376954
- 99% percentile: 1.7383343505859197



No sources have both SDSS r (petroMag) and Megacam r (total) values.

GPC1 r (total) - SDSS r (petroMag):

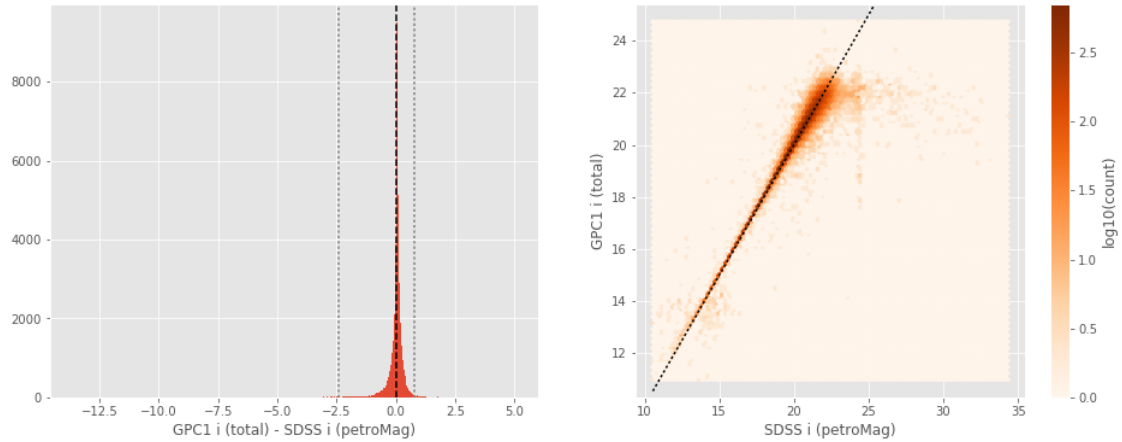
- Median: 0.03
- Median Absolute Deviation: 0.10
- 1% percentile: -2.456576461791992
- 99% percentile: 1.0839012145996092



No sources have both SDSS i (petroMag) and Megacam i (total) values.

GPC1 i (total) - SDSS i (petroMag):

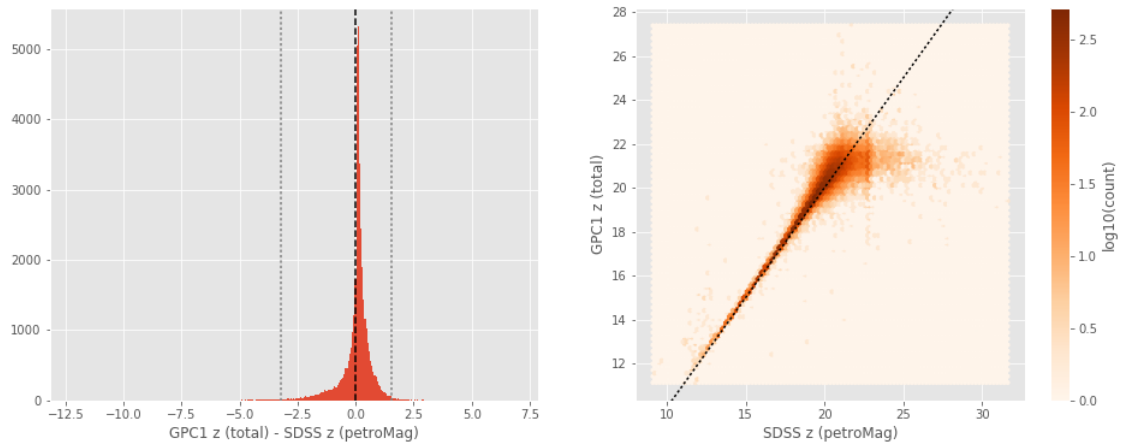
- Median: 0.04
- Median Absolute Deviation: 0.08
- 1% percentile: -2.4044787216186525
- 99% percentile: 0.7786418914794913



No sources have both SDSS z (petroMag) and Megacam z (total) values.

GPC1 z (total) - SDSS z (petroMag):

- Median: 0.11
- Median Absolute Deviation: 0.18
- 1% percentile: -3.216100063323975
- 99% percentile: 1.5221108055114743



1.5.2 III.b - Comparing J and K bands to 2MASS

The catalogue is cross-matched to 2MASS-PSC withing 0.2 arcsecond. We compare the UKIDSS total J and K magnitudes to those from 2MASS.

The 2MASS magnitudes are “Vega-like” and we have to convert them to AB magnitudes using the zero points provided on [this page](#):

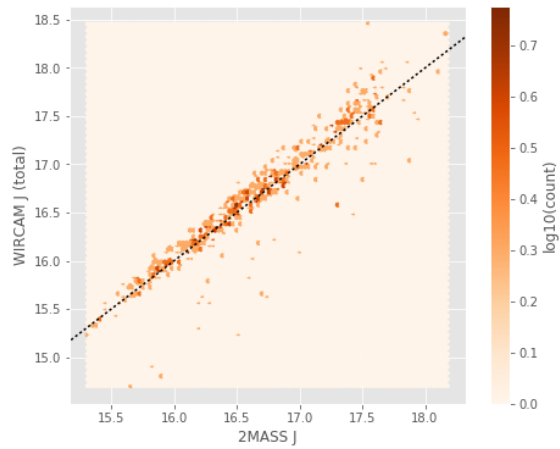
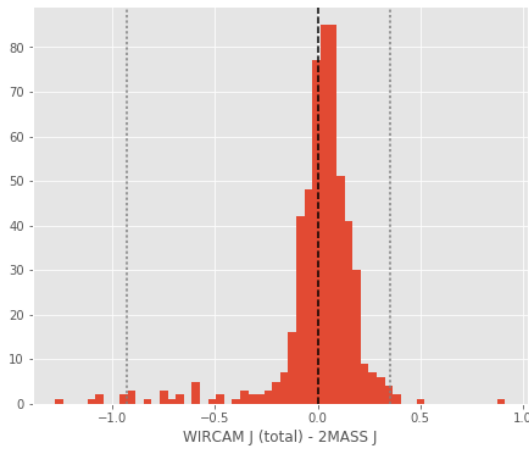
Band	F - 0 mag (Jy)
J	1594
H	1024
Ks	666.7

In addition, UKIDSS uses a K band whereas 2MASS uses a Ks (“short”) band, [this page](#) give a correction to convert the K band in a Ks band with the formula:

$$K_{s(2MASS)} = K_{UKIRT} + 0.003 + 0.004 * (JK)_{UKIRT}$$

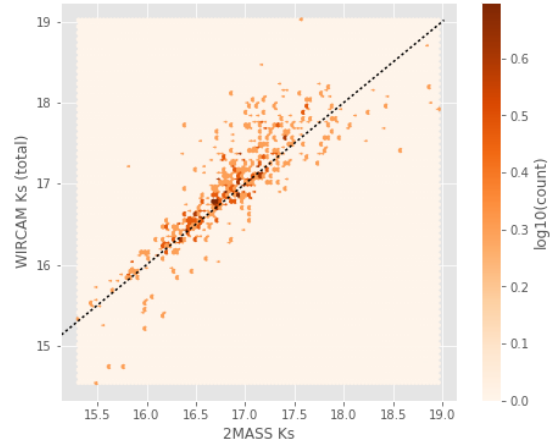
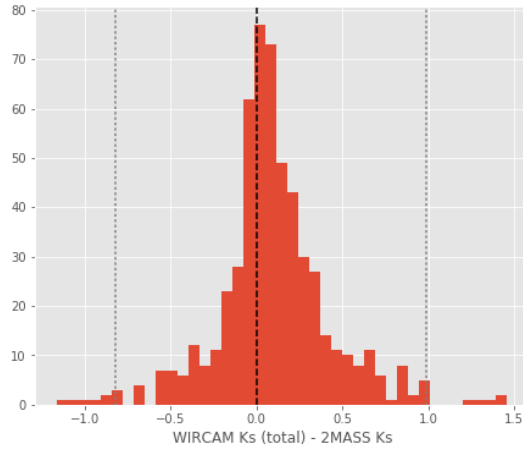
WIRCAM J (total) - 2MASS J:

- Median: 0.03
- Median Absolute Deviation: 0.07
- 1% percentile: -0.928474068441564
- 99% percentile: 0.3543059425447635



WIRCAM Ks (total) - 2MASS Ks:

- Median: 0.08
- Median Absolute Deviation: 0.14
- 1% percentile: -0.8225997304400008
- 99% percentile: 0.9870411423627337



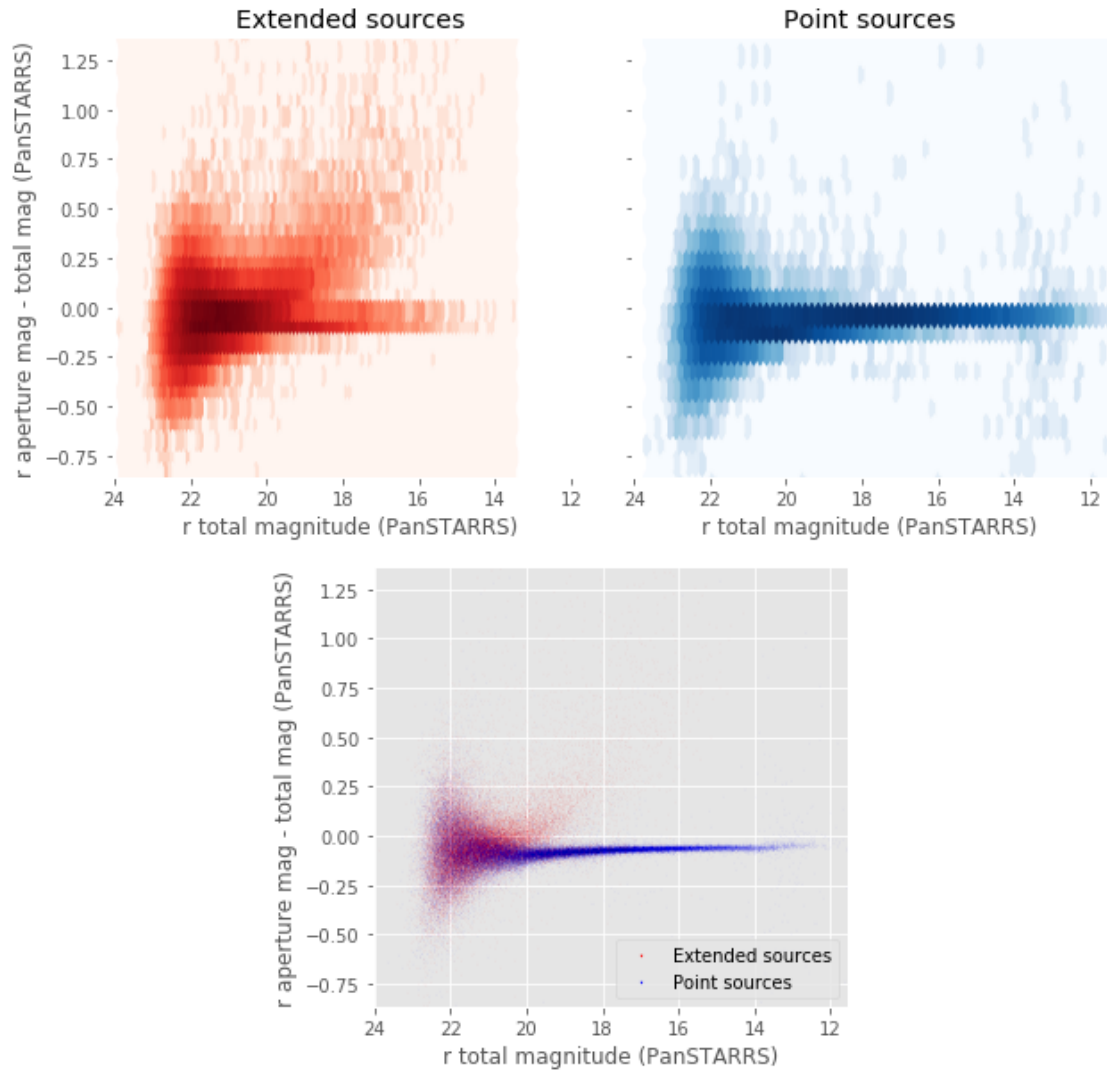
1.6 Keeping only sources with good signal to noise ratio

From here, we are only comparing sources with a signal to noise ratio above 3, i.e. roughly we have a magnitude error below 0.3.

To make it easier, we are setting to NaN in the catalogue the magnitudes associated with an error above 0.3 so we can't use these magnitudes after the next cell.

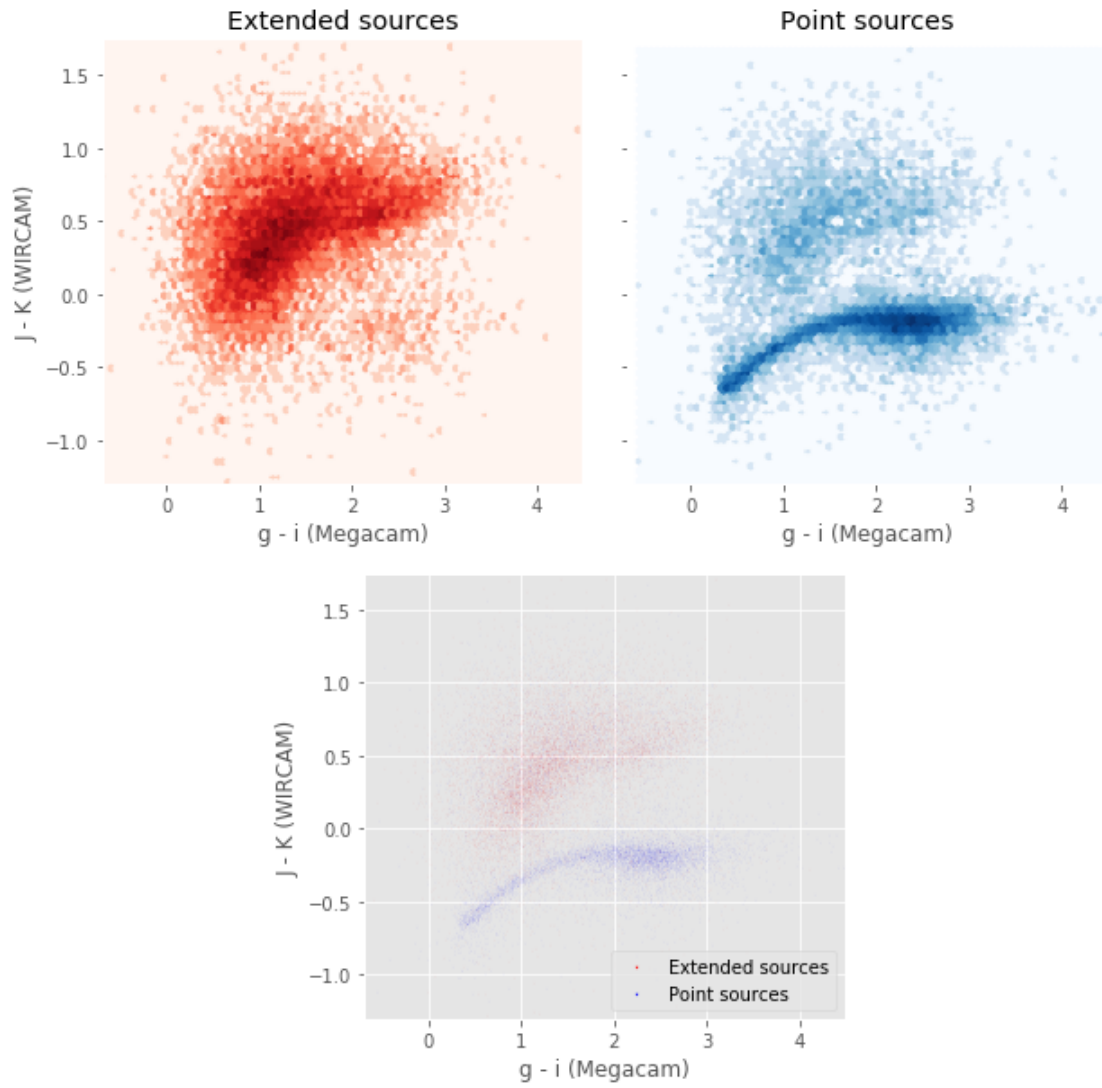
1.7 IV - Comparing aperture magnitudes to total ones.

Number of source used: 103040 / 531746 (19.38%)

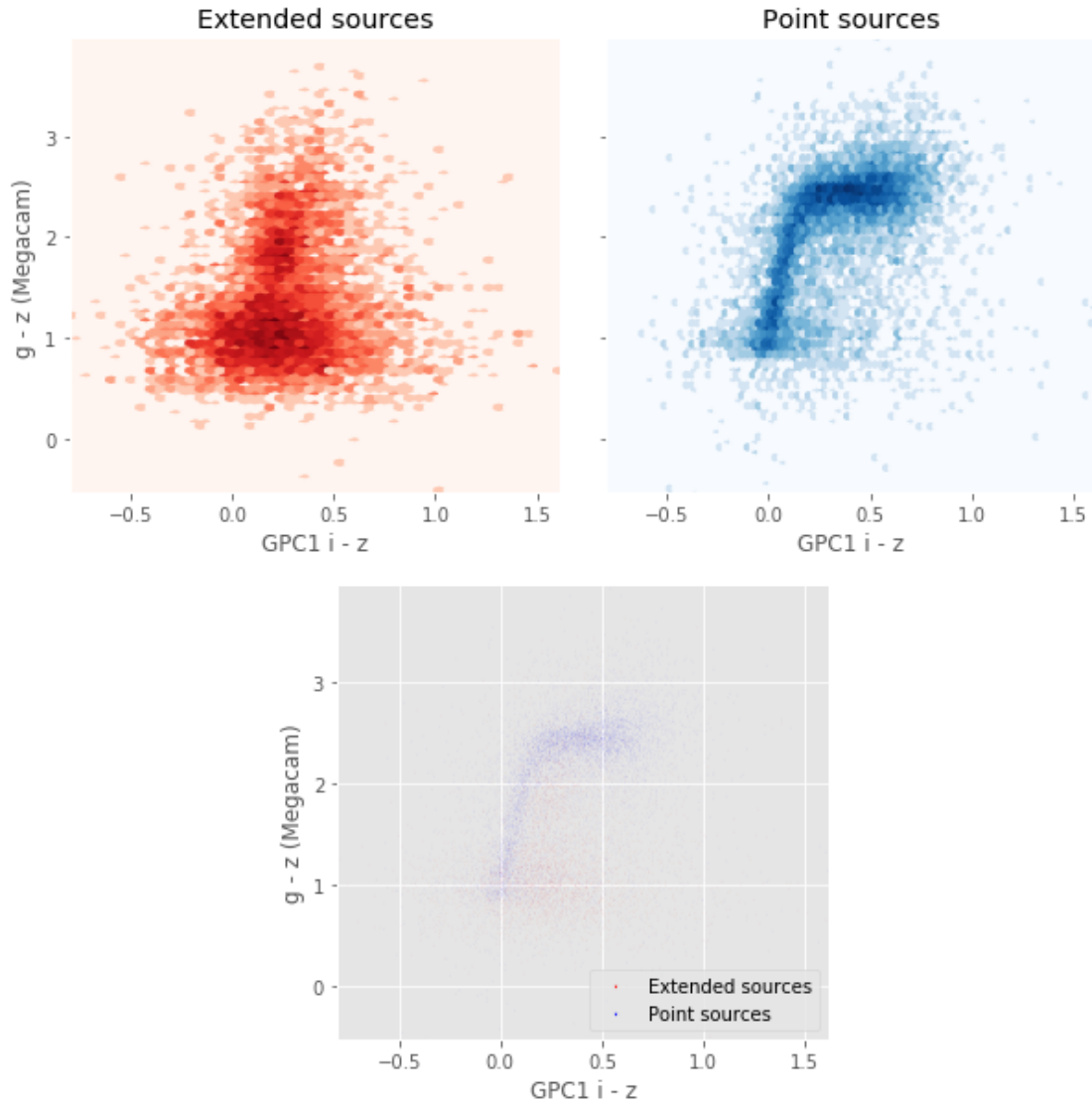


1.8 V - Color-color and magnitude-color plots

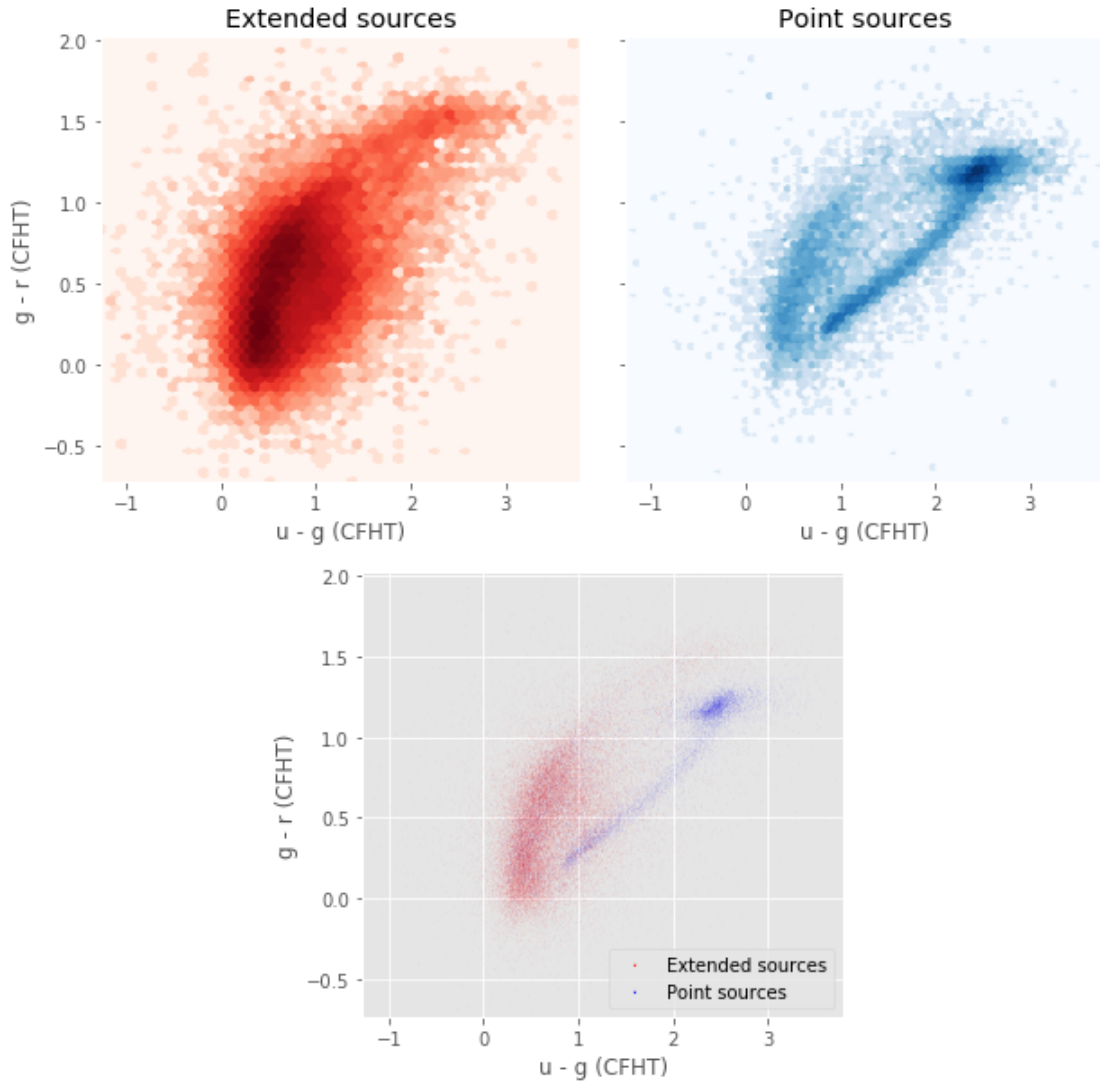
Number of source used: 25036 / 531746 (4.71%)



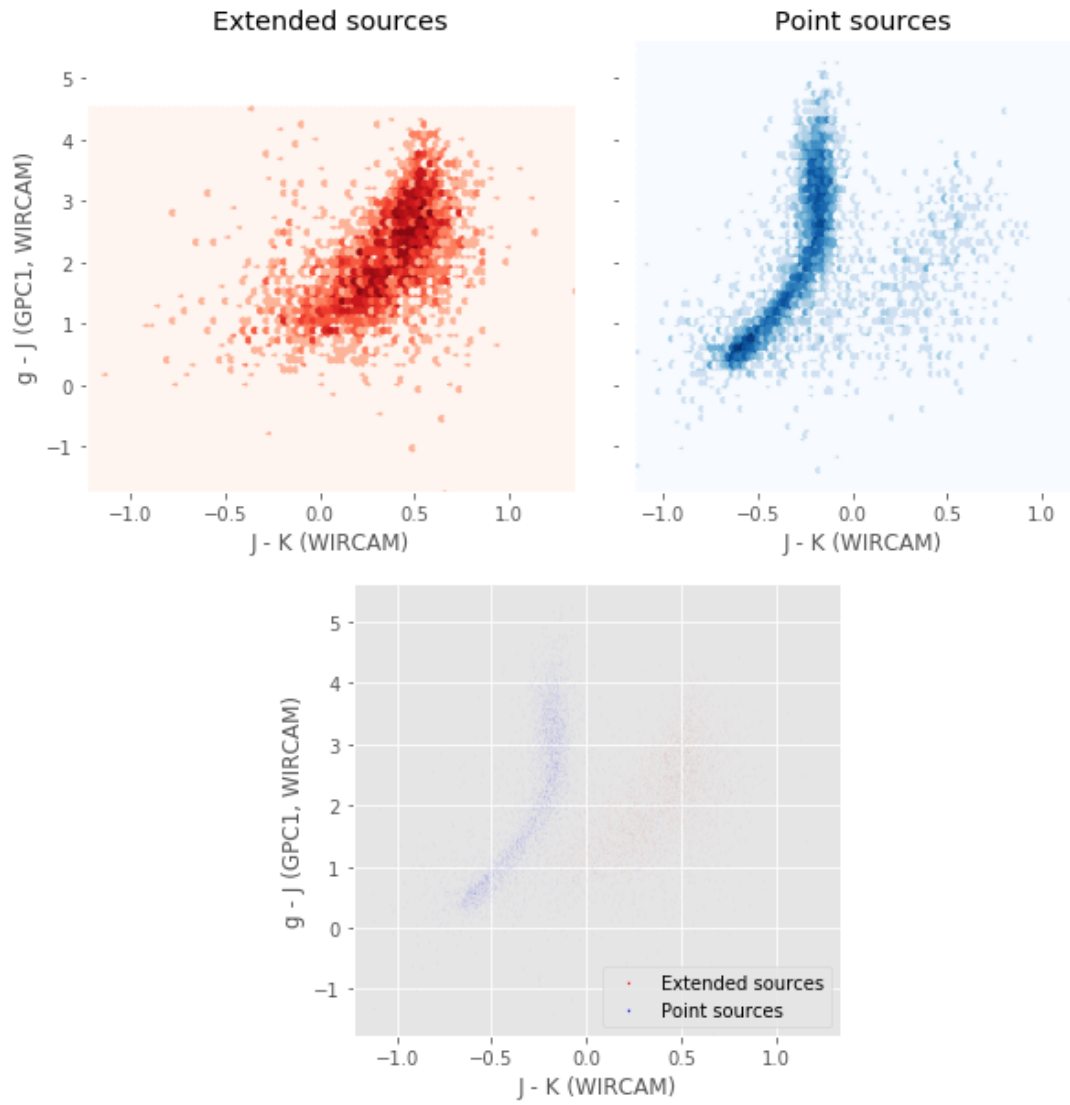
Number of source used: 13862 / 531746 (2.61%)



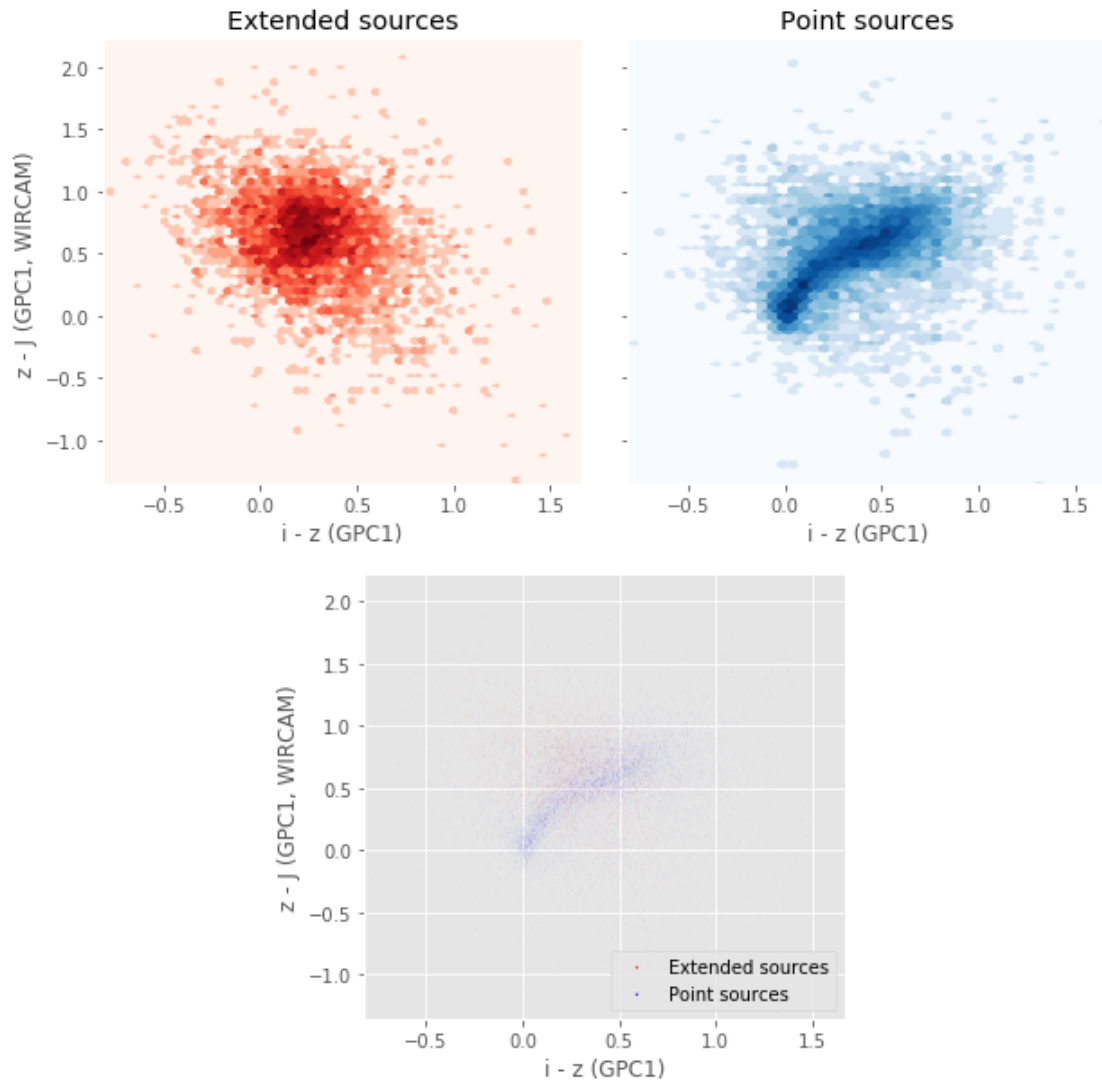
Number of source used: 52325 / 531746 (9.84%)



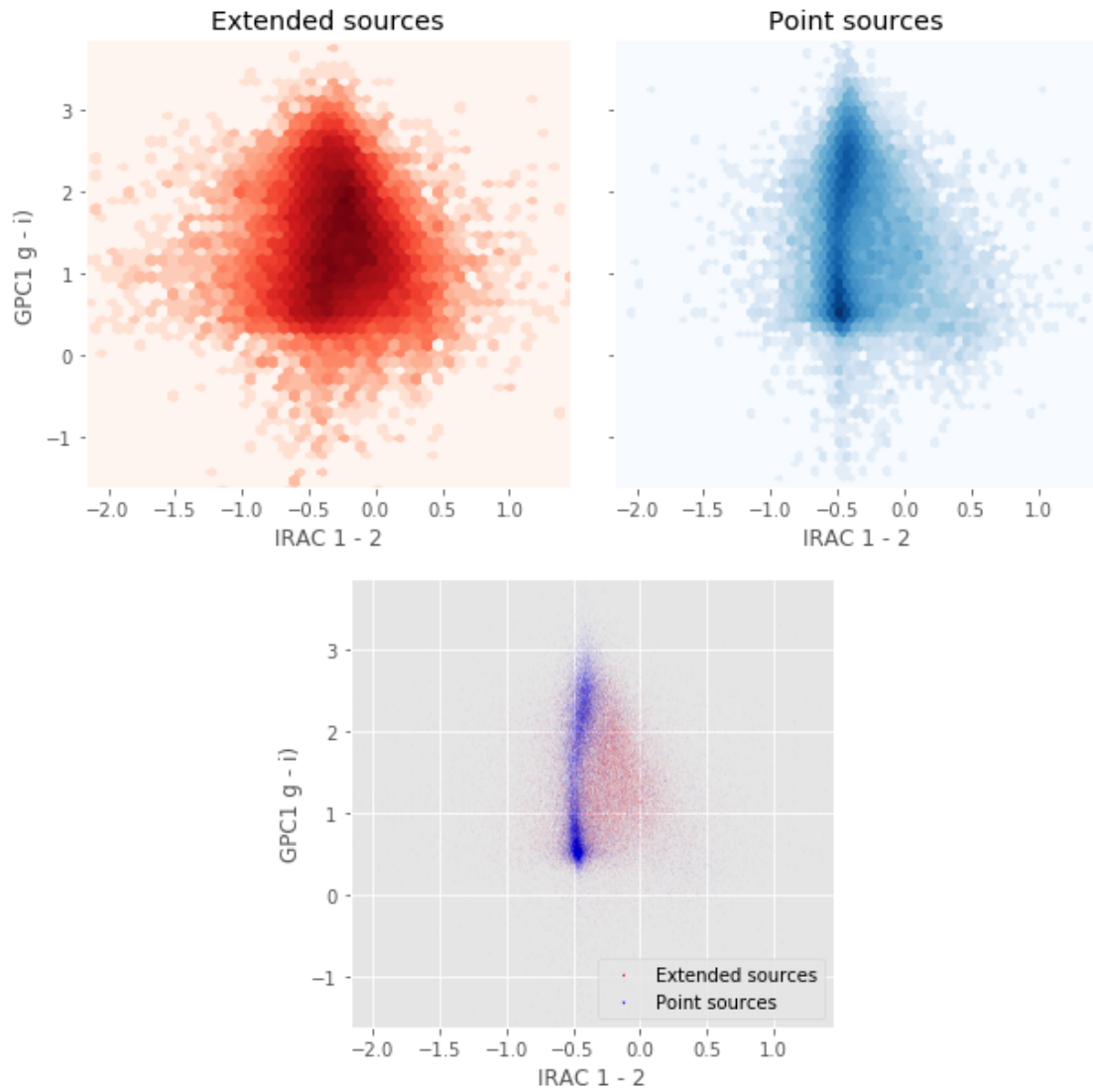
Number of source used: 7775 / 531746 (1.46%)



Number of source used: 11335 / 531746 (2.13%)



Number of source used: 68687 / 531746 (12.92%)



4_Selection_function

March 8, 2018

1 AKARI-NEP Selection Functions

1.1 Depth maps and selection functions

The simplest selection function available is the field MOC which specifies the area for which there is Herschel data. Each pristine catalogue also has a MOC defining the area for which that data is available.

The next stage is to provide mean flux standard deviations which act as a proxy for the catalogue's 5σ depth

```
This notebook was run with herschelhelp_internal version:  
0246c5d (Thu Jan 25 17:01:47 2018 +0000) [with local modifications]  
This notebook was executed on:  
2018-02-27 17:30:59.886785
```

Depth maps produced using: master_catalogue_akari-nep_20180215.fits

1.2 I - Group masterlist objects by healpix cell and calculate depths

We add a column to the masterlist catalogue for the target order healpix cell per object.

1.3 II Create a table of all Order=13 healpix cells in the field and populate it

We create a table with every order=13 healpix cell in the field MOC. We then calculate the healpix cell at lower order that the order=13 cell is in. We then fill in the depth at every order=13 cell as calculated for the lower order cell that that the order=13 cell is inside.

```
Out[9]: <IPython.core.display.HTML object>
```

```
Out[11]: <IPython.core.display.HTML object>
```

```
Out[12]: <IPython.core.display.HTML object>
```

1.4 III - Save the depth map table

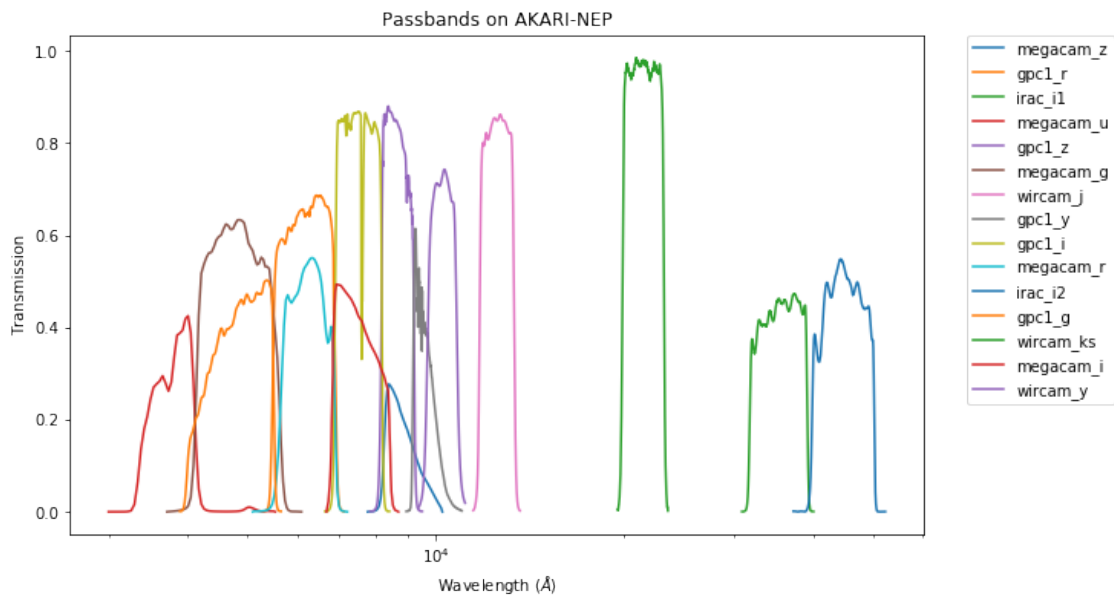
1.5 IV - Overview plots

1.5.1 IV.a - Filters

First we simply plot all the filters available on this field to give an overview of coverage.

```
Out[14]: {'gpc1_g',  
          'gpc1_i',  
          'gpc1_r',  
          'gpc1_y',  
          'gpc1_z',  
          'irac_i1',  
          'irac_i2',  
          'megacam_g',  
          'megacam_i',  
          'megacam_r',  
          'megacam_u',  
          'megacam_z',  
          'wircam_j',  
          'wircam_ks',  
          'wircam_y'}
```

```
Out[15]: <matplotlib.text.Text at 0x7f94d0b51198>
```



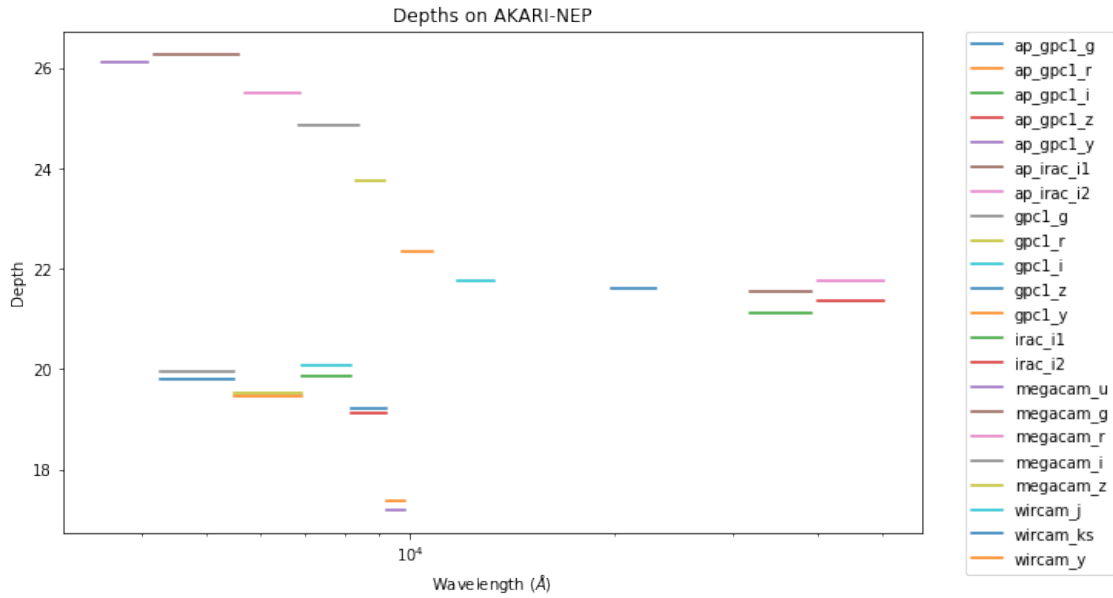
1.5.2 IV.a - Depth overview

Then we plot the mean depths available across the area a given band is available

gpc1_g: mean flux error: 8.806699357169498, 3sigma in AB mag (Aperture): 20.34516393653916
gpc1_r: mean flux error: 12.067129104246586, 3sigma in AB mag (Aperture): 20.00318696519175
gpc1_i: mean flux error: 8.251892546976062, 3sigma in AB mag (Aperture): 20.41581295290225
gpc1_z: mean flux error: 16.296125057512807, 3sigma in AB mag (Aperture): 19.676985991179073
gpc1_y: mean flux error: 97.42729605212693, 3sigma in AB mag (Aperture): 17.735495239381372
irac_i1: mean flux error: 1.7294846773147583, 3sigma in AB mag (Aperture): 22.112405066560406
irac_i2: mean flux error: 1.4147037267684937, 3sigma in AB mag (Aperture): 22.330533119207182
gpc1_g: mean flux error: 7.616552343703014, 3sigma in AB mag (Total): 20.50280078555148
gpc1_r: mean flux error: 11.285212165987328, 3sigma in AB mag (Total): 20.075922542249153
gpc1_i: mean flux error: 6.779790275782894, 3sigma in AB mag (Total): 20.62915621388678
gpc1_z: mean flux error: 15.101349154690517, 3sigma in AB mag (Total): 19.759657490950865
gpc1_y: mean flux error: 81.92646632009088, 3sigma in AB mag (Total): 17.9236363053817
irac_i1: mean flux error: 2.5585498809814453, 3sigma in AB mag (Total): 21.687212142426283
irac_i2: mean flux error: 2.087070941925049, 3sigma in AB mag (Total): 21.908353834496303
megacam_u: mean flux error: 0.025724060833454132, 3sigma in AB mag (Total): 26.681348043321556
megacam_g: mean flux error: 0.02217845991253853, 3sigma in AB mag (Total): 26.842368400316808
megacam_r: mean flux error: 0.04555840045213699, 3sigma in AB mag (Total): 26.060775694031584
megacam_i: mean flux error: 0.08159784972667694, 3sigma in AB mag (Total): 25.428000077349985
megacam_z: mean flux error: 0.2246941775083542, 3sigma in AB mag (Total): 24.32821731642934
wircam_j: mean flux error: 1.4508192539215088, 3sigma in AB mag (Total): 22.30316358695483
wircam_ks: mean flux error: 1.6619582176208496, 3sigma in AB mag (Total): 22.15564661013537
wircam_y: mean flux error: 0.8292735815048218, 3sigma in AB mag (Total): 22.9104522879201

ap_gpc1_g (4260.0, 5500.0, 1240.0)
ap_gpc1_r (5500.0, 6900.0, 1400.0)
ap_gpc1_i (6910.0, 8190.0, 1280.0)
ap_gpc1_z (8190.0, 9210.0, 1020.0)
ap_gpc1_y (9200.0, 9820.0, 620.0)
ap_irac_i1 (31754.0, 39164.801, 7410.8008)
ap_irac_i2 (39980.102, 50052.301, 10072.199)
gpc1_g (4260.0, 5500.0, 1240.0)
gpc1_r (5500.0, 6900.0, 1400.0)
gpc1_i (6910.0, 8190.0, 1280.0)
gpc1_z (8190.0, 9210.0, 1020.0)
gpc1_y (9200.0, 9820.0, 620.0)
irac_i1 (31754.0, 39164.801, 7410.8008)
irac_i2 (39980.102, 50052.301, 10072.199)
megacam_u (3500.0, 4100.0, 600.0)
megacam_g (4180.0, 5580.0, 1400.0)
megacam_r (5680.0, 6880.0, 1200.0)
megacam_i (6831.7305, 8388.5557, 1556.8252)
megacam_z (8280.0, 9160.0, 880.0)
wircam_j (11748.0, 13334.0, 1586.0)
wircam_ks (19870.0, 23135.0, 3265.0)
wircam_y (9702.0, 10806.0, 1104.0)

Out[20]: <matplotlib.text.Text at 0x7f94d0780c88>



1.5.3 IV.c - Depth vs coverage comparison

How best to do this? Colour/intensity plot over area? Percentage coverage vs mean depth?

Out [21]: <matplotlib.text.Text at 0x7f94ccc32cf8>

