1.1_DECaLS

March 8, 2018

1 NGP master catalogue

1.1 Preparation of DECam Legacy Survey data

This catalogue comes from `dmu0_DECaLS`.

In the catalogue, we keep:

- The `object_id` as unique object identifier;
- The position;
- The u, g, r, i, z, Y aperture magnitude (2");
- The u, g, r, i, z, Y kron fluxes and magnitudes.

We check for all ugrizY then only take bands for which there are measurements

```
This notebook was run with herschelhelp_internal version:
44f1ae0 (Thu Nov 30 18:27:54 2017 +0000)
```

```
WARNING: UnitsWarning: '1/deg^2' did not parse as fits unit: Numeric factor not supported by FIT
WARNING: UnitsWarning: 'nanomaggy' did not parse as fits unit: At col 0, Unit 'nanomaggy' not su
WARNING: UnitsWarning: '1/nanomaggy^2' did not parse as fits unit: Numeric factor not supported
WARNING: UnitsWarning: '1/arcsec^2' did not parse as fits unit: Numeric factor not supported by
```

1.2 I - Aperture correction

To compute aperture correction we need to dertermine two parametres: the target aperture and the range of magnitudes for the stars that will be used to compute the correction.

Target aperture: To determine the target aperture, we simulate a curve of growth using the provided apertures and draw two figures:

- The evolution of the magnitudes of the objects by plotting on the same plot aperture number vs the mean magnitude.
- The mean gain (loss when negative) of magnitude is each aperture compared to the previous (except for the first of course).

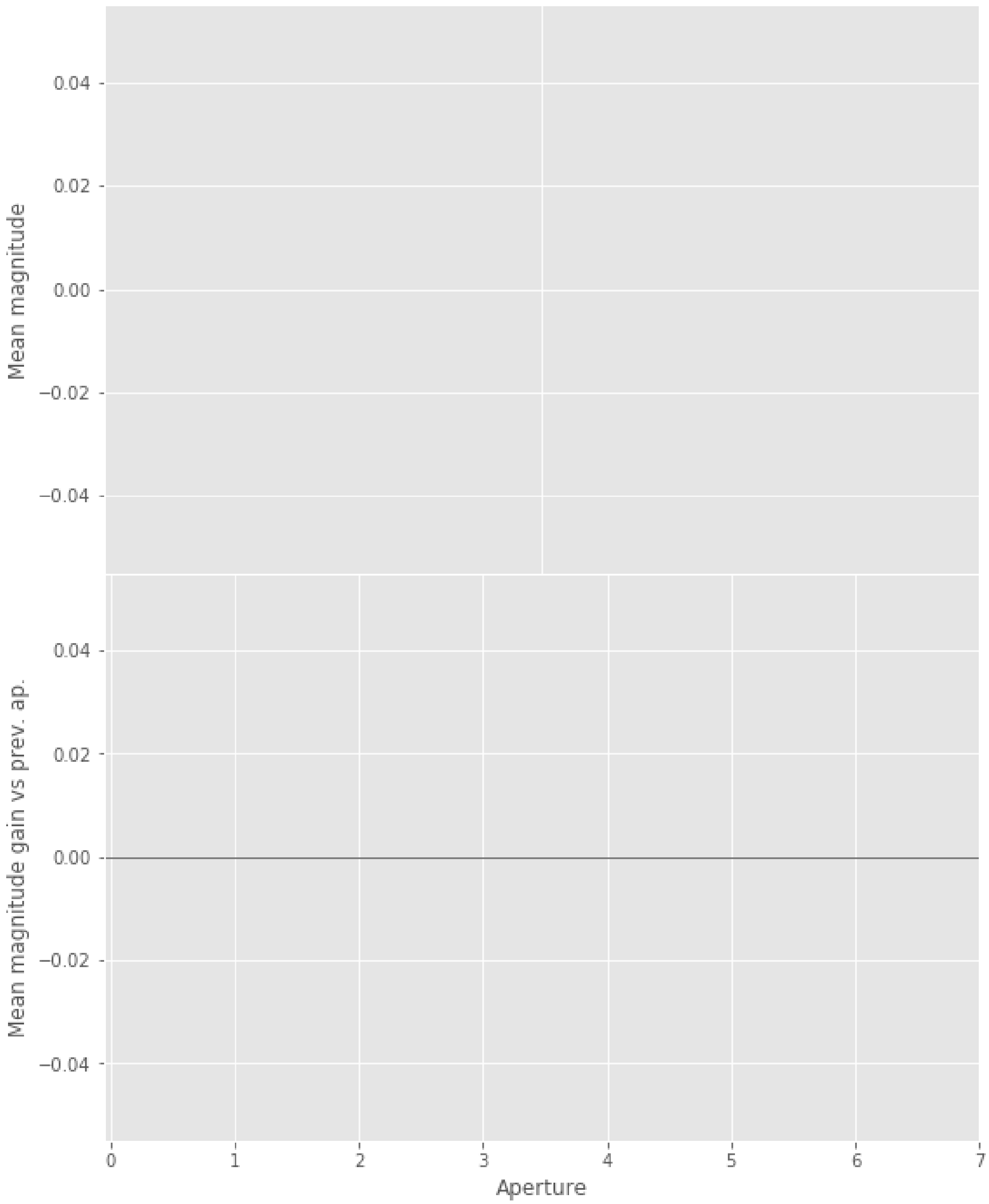
As target aperture, we should use the smallest (i.e. less noisy) aperture for which most of the flux is captures.

Magnitude range: To know what limits in aperture to use when doing the aperture correction, we plot for each magnitude bin the correction that is computed and its RMS. We should then use the wide limits (to use more stars) where the correction is stable and with few dispersion.

```
/opt/herschelhelp_internal/herschelhelp_internal/utils.py:76: RuntimeWarning: divide by zero enc
  magnitudes = 2.5 * (23 - np.log10(fluxes)) - 48.6
/opt/herschelhelp_internal/herschelhelp_internal/utils.py:80: RuntimeWarning: invalid value enco
  errors = 2.5 / np.log(10) * errors_on_fluxes / fluxes
/opt/herschelhelp_internal/herschelhelp_internal/utils.py:76: RuntimeWarning: invalid value enco
  magnitudes = 2.5 * (23 - np.log10(fluxes)) - 48.6
```

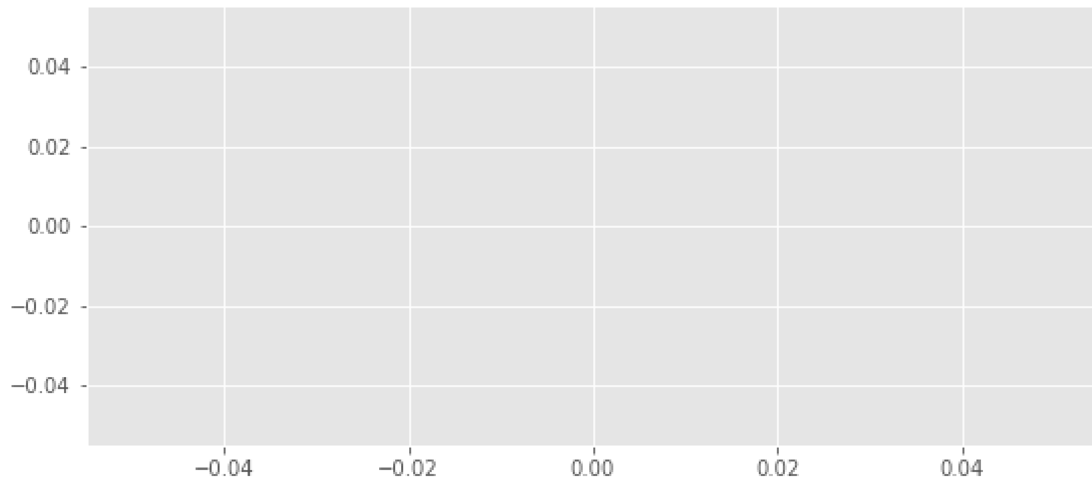
1.2.1 1.a u band

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
```



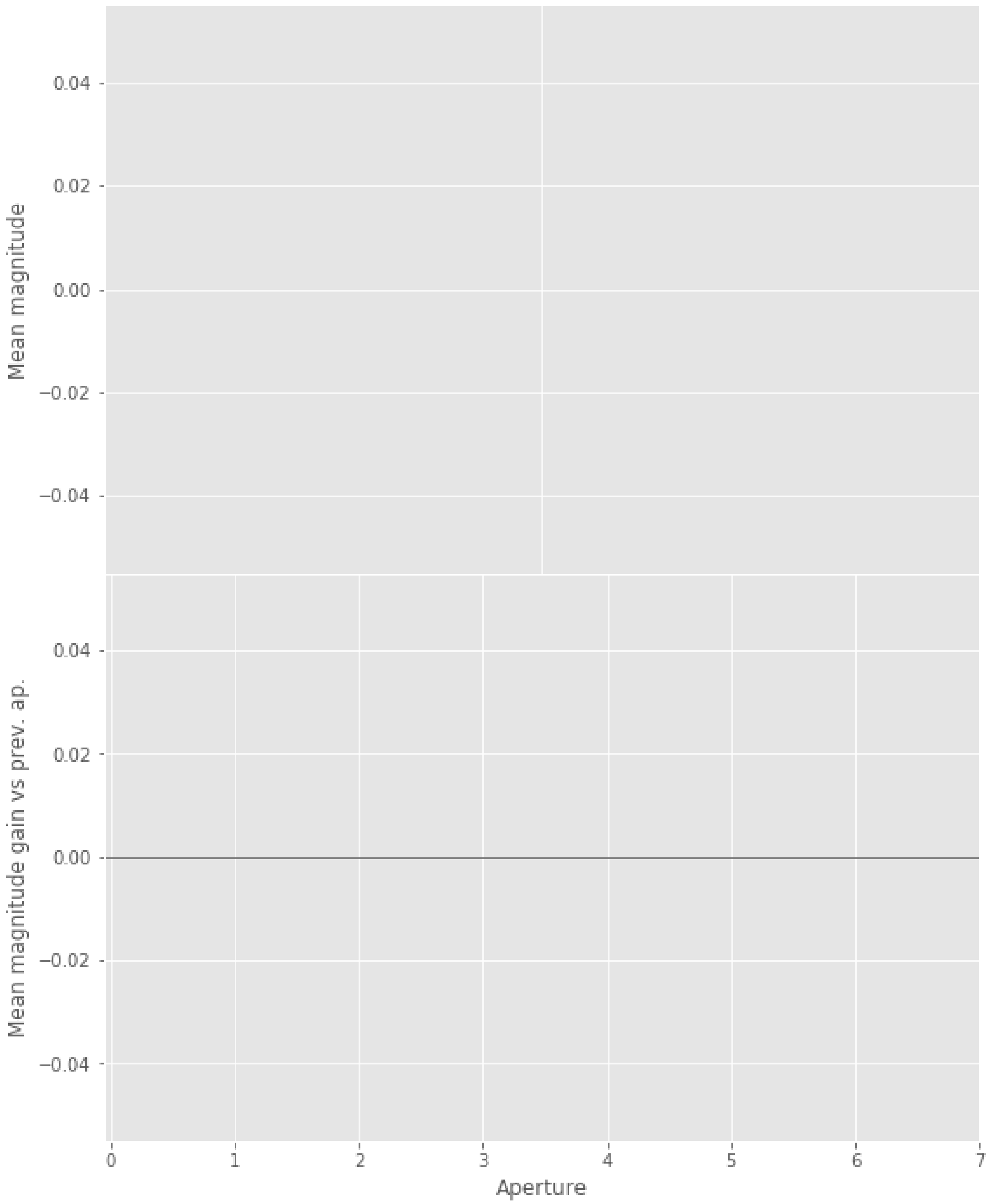
u band is all nan

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("All-NaN slice encountered", RuntimeWarning)
```



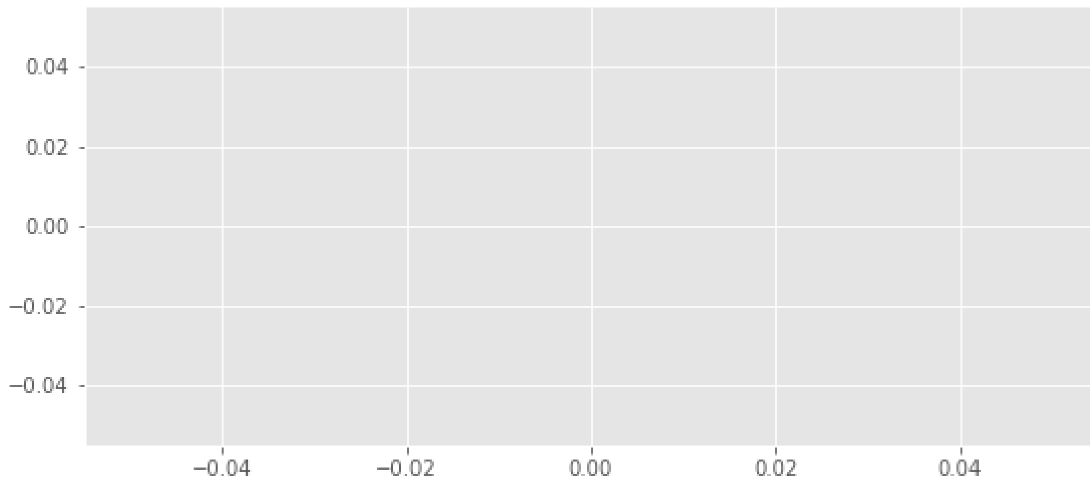
1.2.2 I.a - g band

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
```



We will use aperture 5 as target.

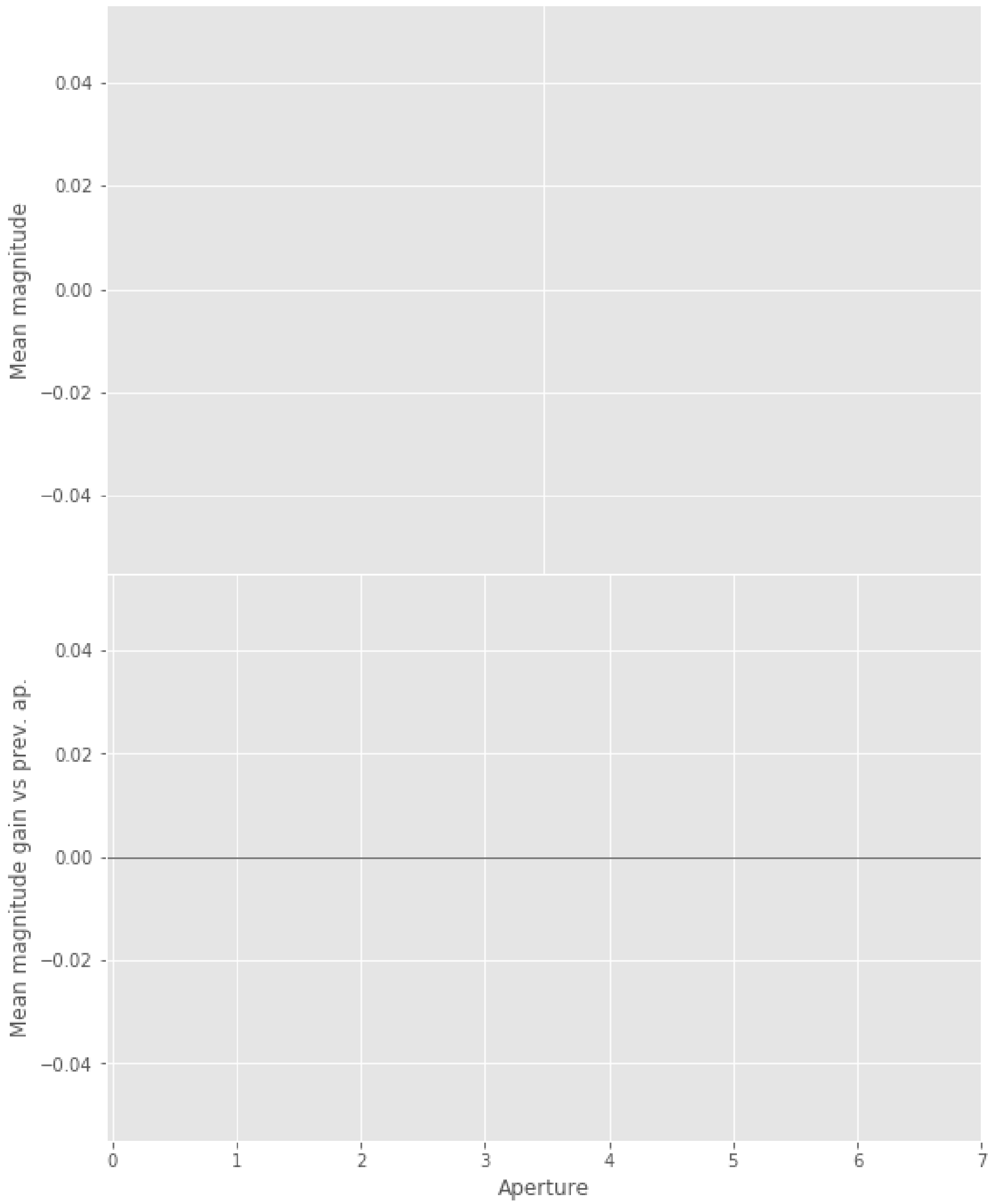
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("All-NaN slice encountered", RuntimeWarning)
```



We will use magnitudes between 16.0 and 19.0

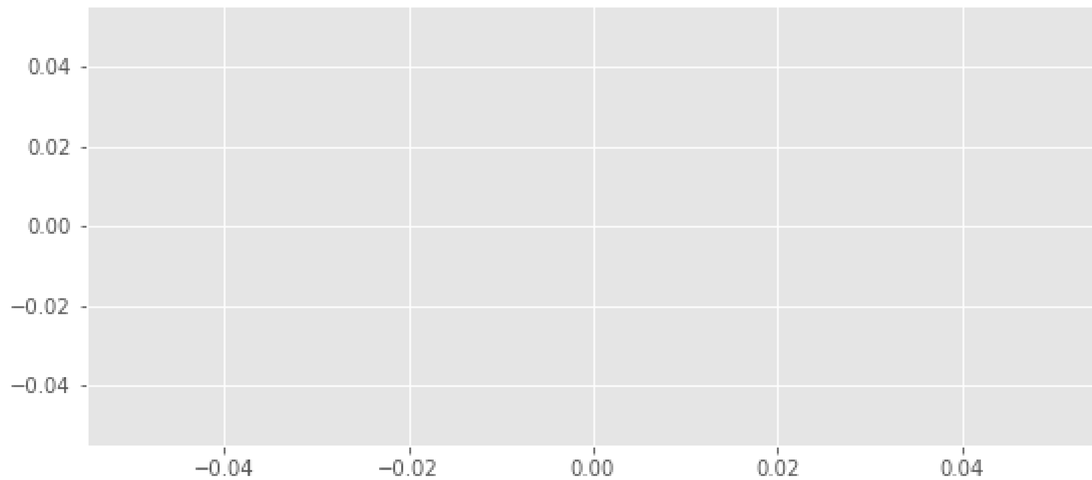
1.2.3 I.b - r band

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
```



We will use aperture 5 as target.

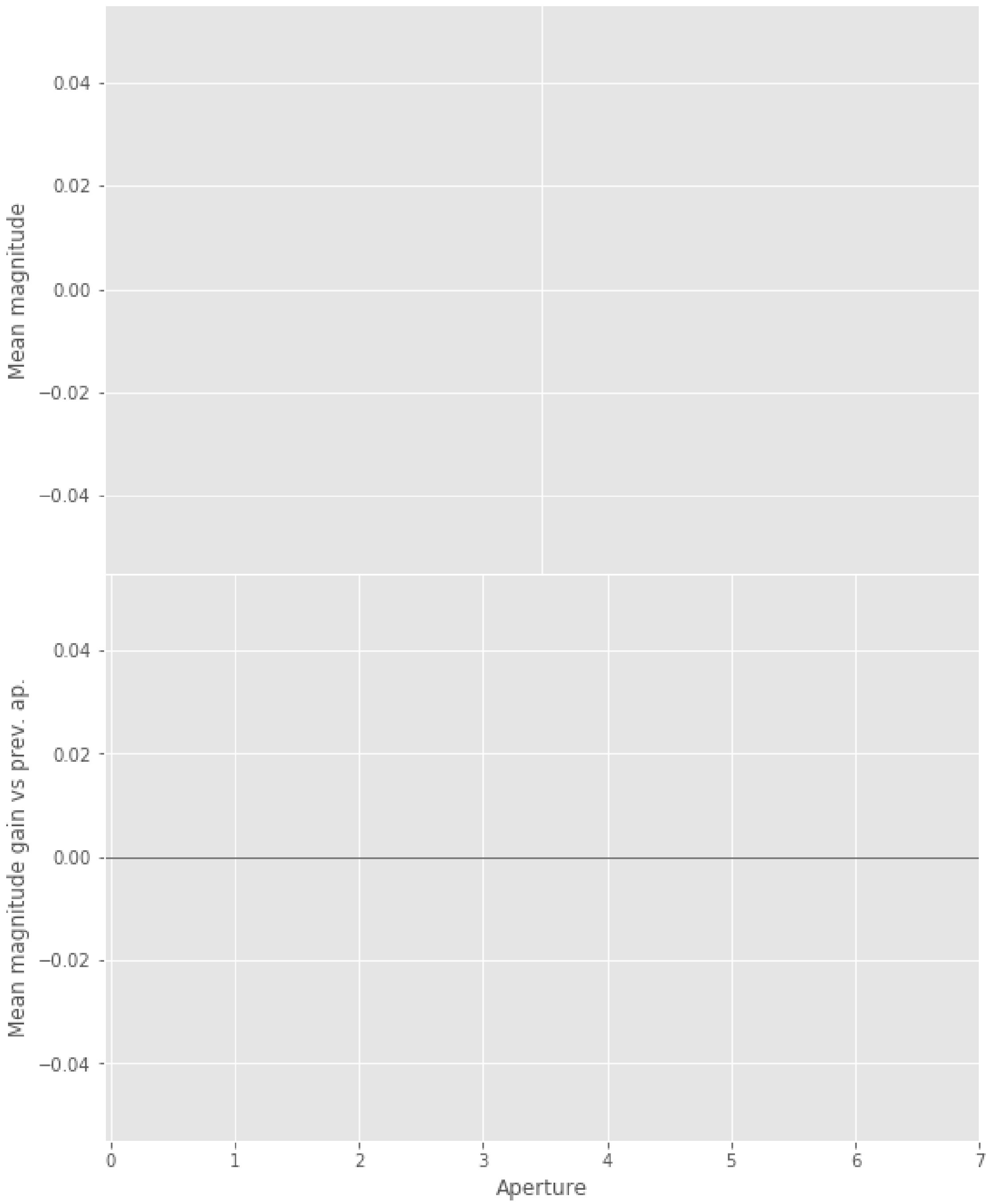
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("All-NaN slice encountered", RuntimeWarning)
```



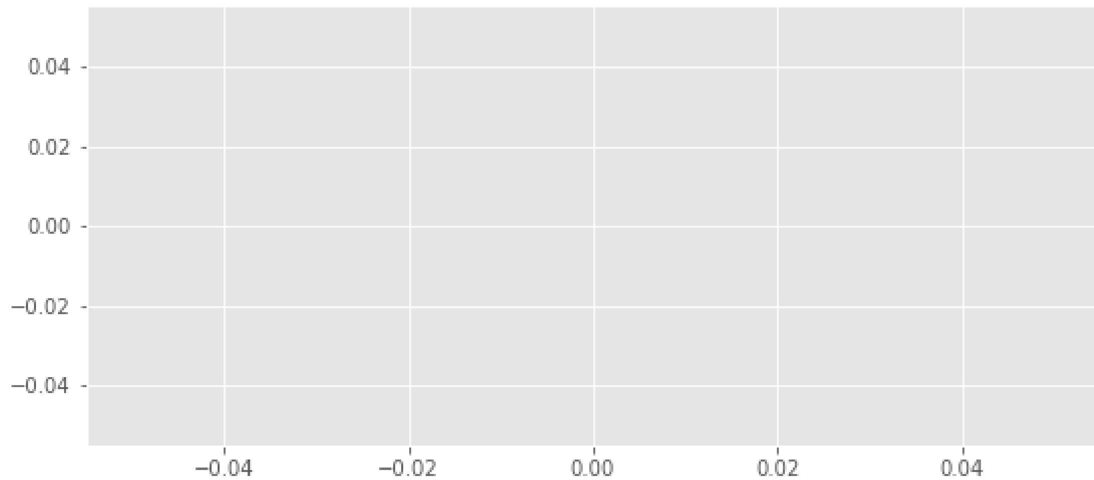
We use magnitudes between 16.0 and 18.0.

1.2.4 I.d - i band

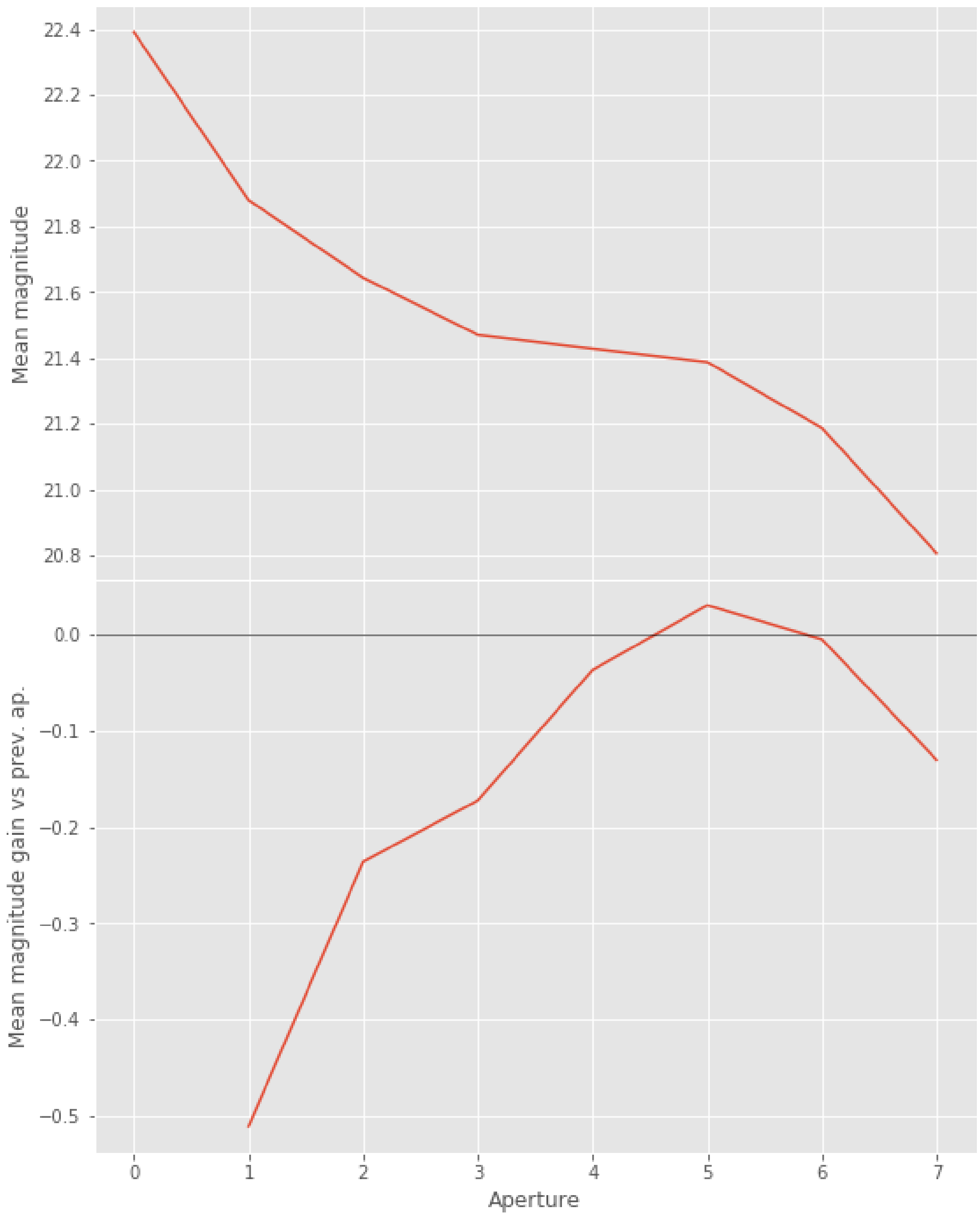
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
```



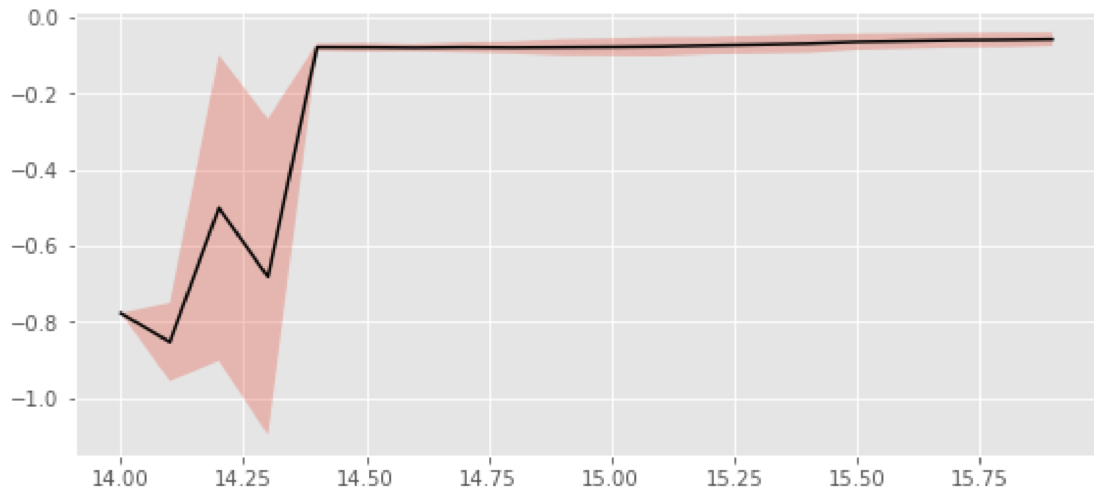

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("All-NaN slice encountered", RuntimeWarning)
```


0.04
0.02
0.00
−0.02
−0.04
−0.04
−0.02
0.00
0.02
0.04

1.2.5 I.e - z band



We will use aperture 5 as target.

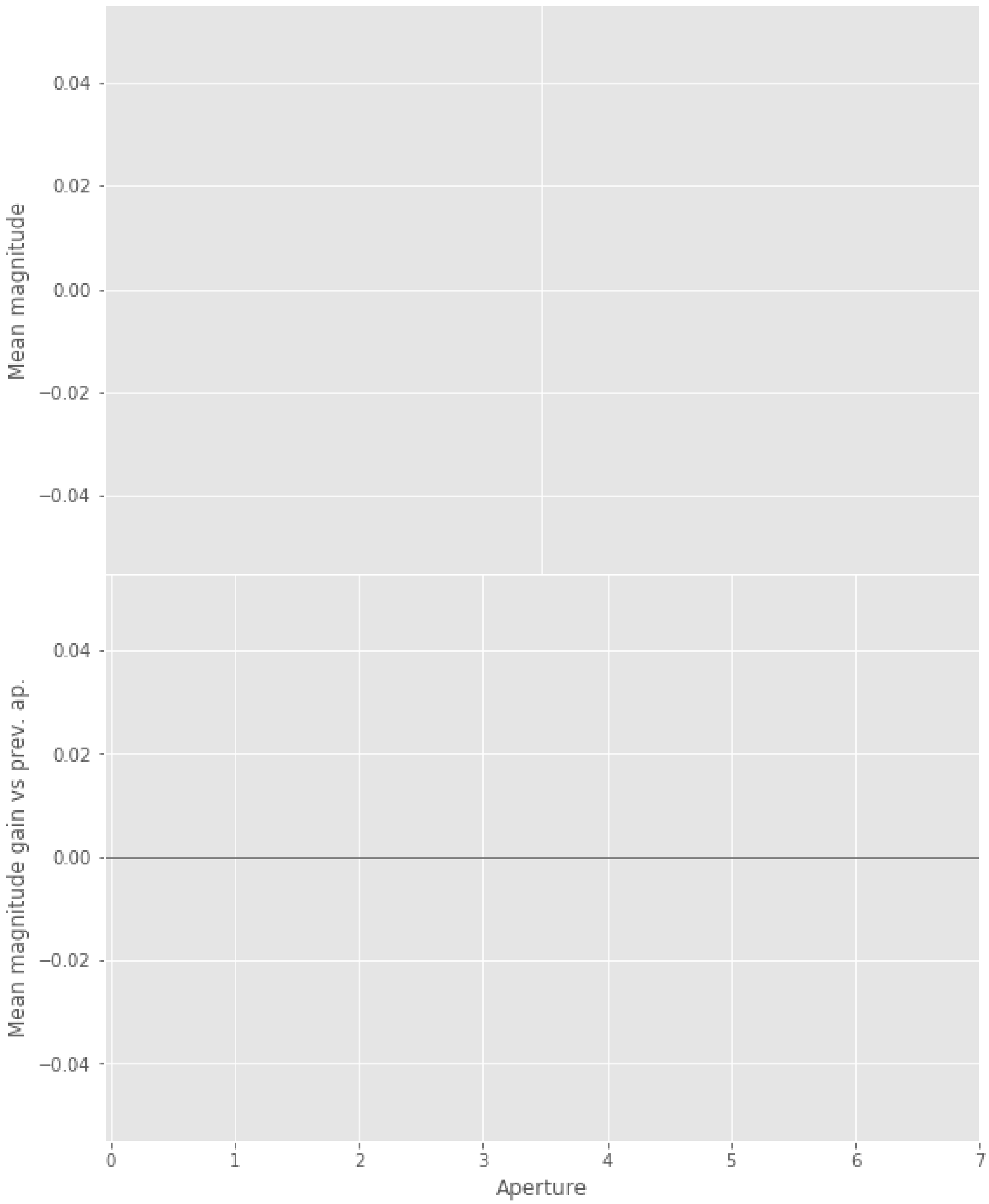


We use magnitudes between 15.0 and 16.0.

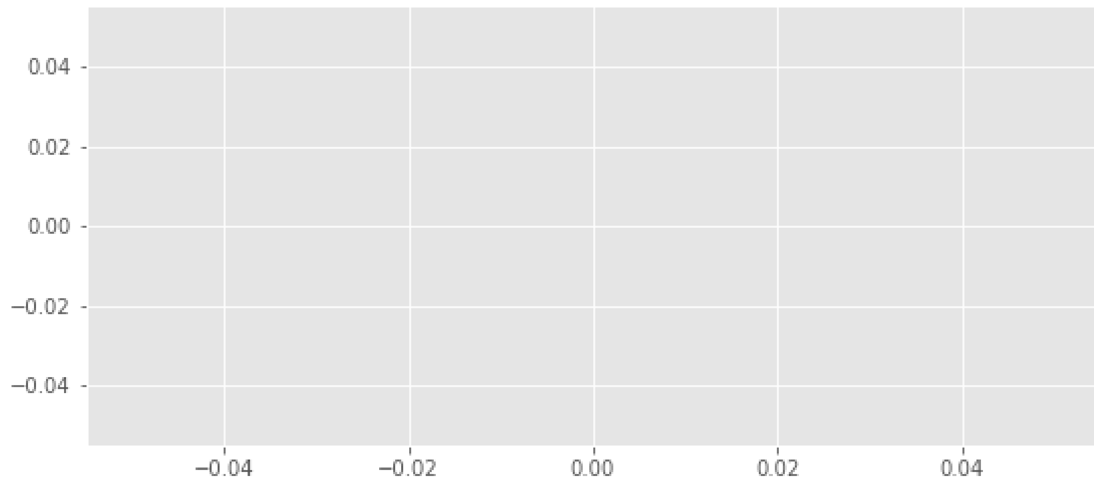
```
Aperture correction for z band:
Correction: -0.0636219666916844
Number of source used: 23160
RMS: 0.022062343753839846
```

1.2.6 I.f - Y band

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("Mean of empty slice", RuntimeWarning)
```




```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/lib/nanfunctions.py:
  warnings.warn("All-NaN slice encountered", RuntimeWarning)
```



1.3 II - Stellarity

Legacy Survey does not provide a 0 to 1 stellarity so we replace items flagged as PSF accpording to the following table:

$$P(star) = \frac{\prod_i P(star)_i}{\prod_i P(star)_i + \prod_i P(galaxy)_i}$$

where i is the band, and with using the same probabilities as UKDISS:

HSC flag	UKIDSS flag	Meaning	P(star)	P(galaxy)	P(noise)	P(saturated)
	-9	Saturated	0.0	0.0	5.0	95.0
	-3	Probable galaxy	25.0	70.0	5.0	0.0
	-2	Probable star	70.0	25.0	5.0	0.0
0	-1	Star	90.0	5.0	5.0	0.0
	0	Noise	5.0	5.0	90.0	0.0
1	+1	Galaxy	5.0	90.0	5.0	0.0

1.4 II - Column selection

```
WARNING: UnitsWarning: '1/deg^2' did not parse as fits unit: Numeric factor not supported by FIT
WARNING: UnitsWarning: 'nanomaggy' did not parse as fits unit: At col 0, Unit 'nanomaggy' not su
WARNING: UnitsWarning: '1/nanomaggy^2' did not parse as fits unit: Numeric factor not supported
WARNING: UnitsWarning: '1/arcsec^2' did not parse as fits unit: Numeric factor not supported by


/opt/herschelhelp_internal/herschelhelp_internal/utils.py:76: RuntimeWarning: invalid value enco
  magnitudes = 2.5 * (23 - np.log10(fluxes)) - 48.6
```

```
Out[27]: <IPython.core.display.HTML object>
```

1.5 III - Removal of duplicated sources

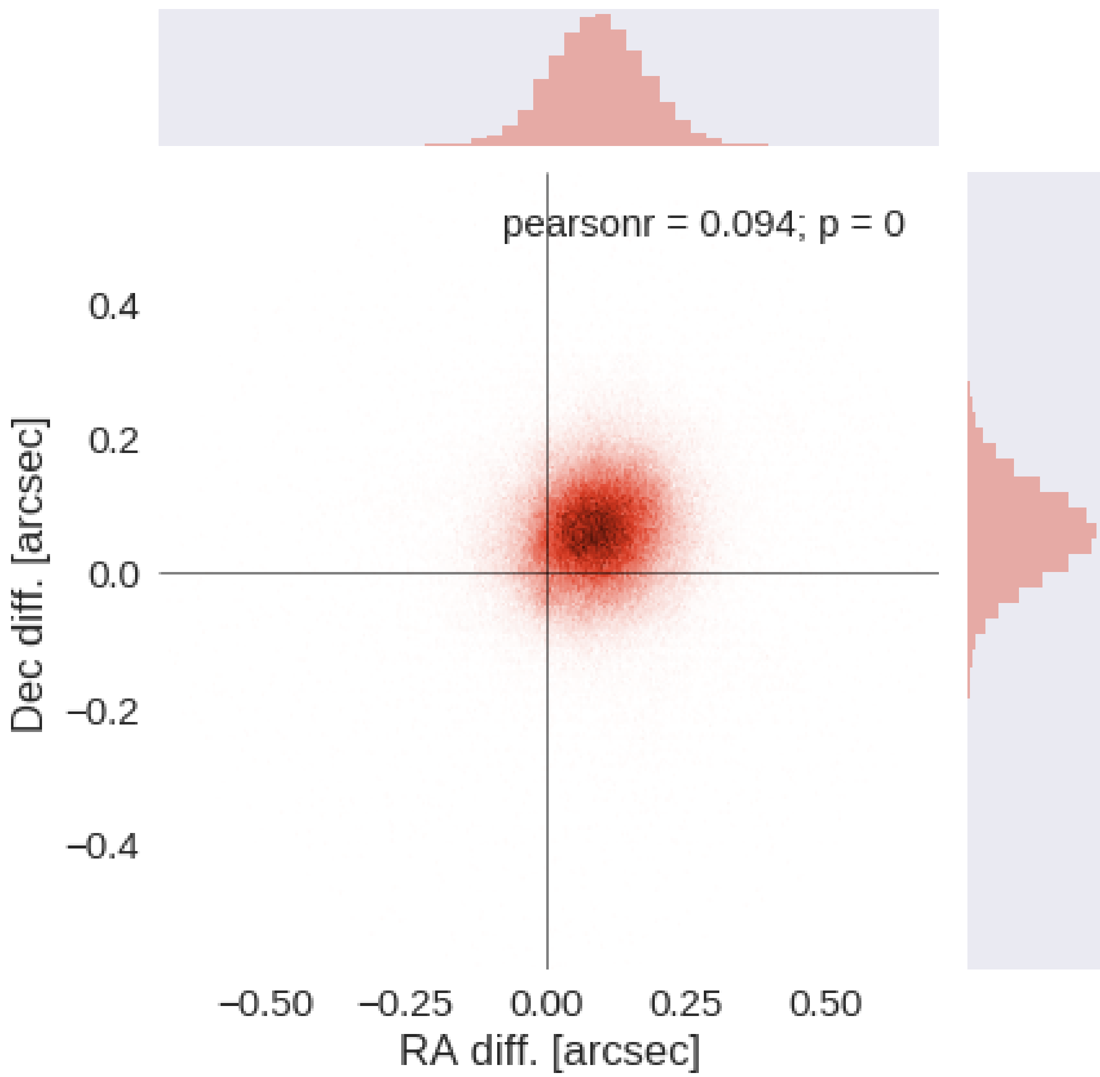
We remove duplicated objects from the input catalogues.

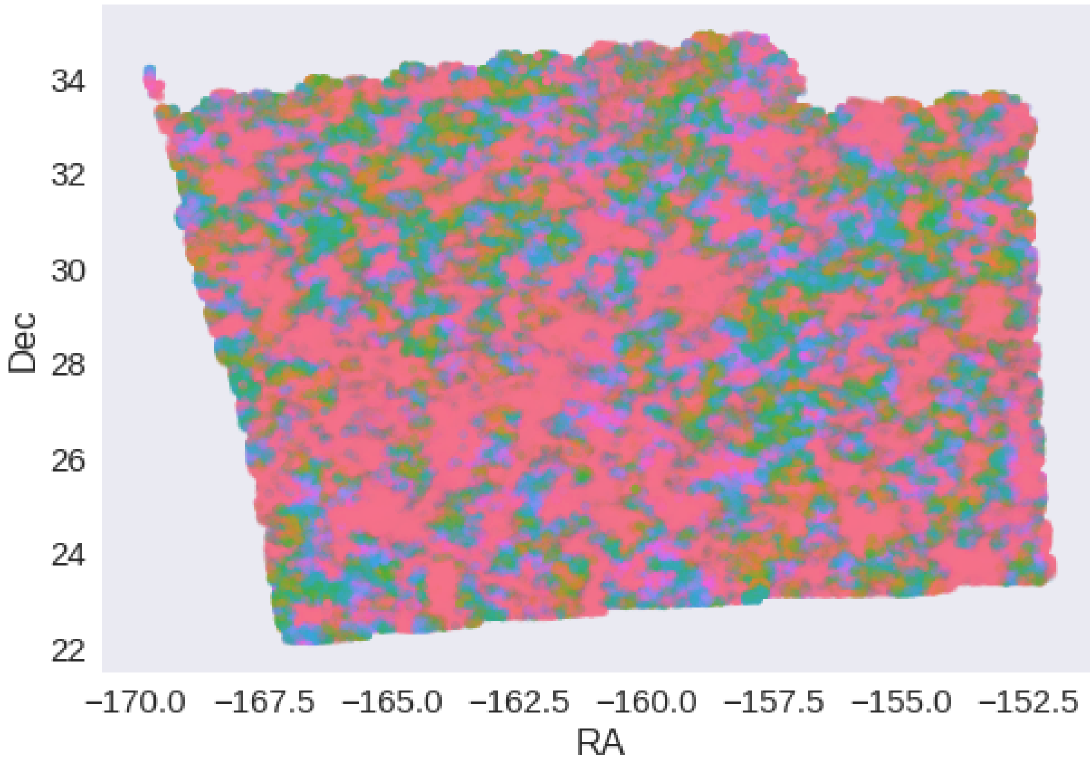
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10
Check the NumPy 1.11 release notes for more information.
  ma.MaskedArray.__setitem__(self, index, value)


The initial catalogue had 4942560 sources.
The cleaned catalogue has 4942322 sources (238 removed).
The cleaned catalogue has 238 sources flagged as having been cleaned
```

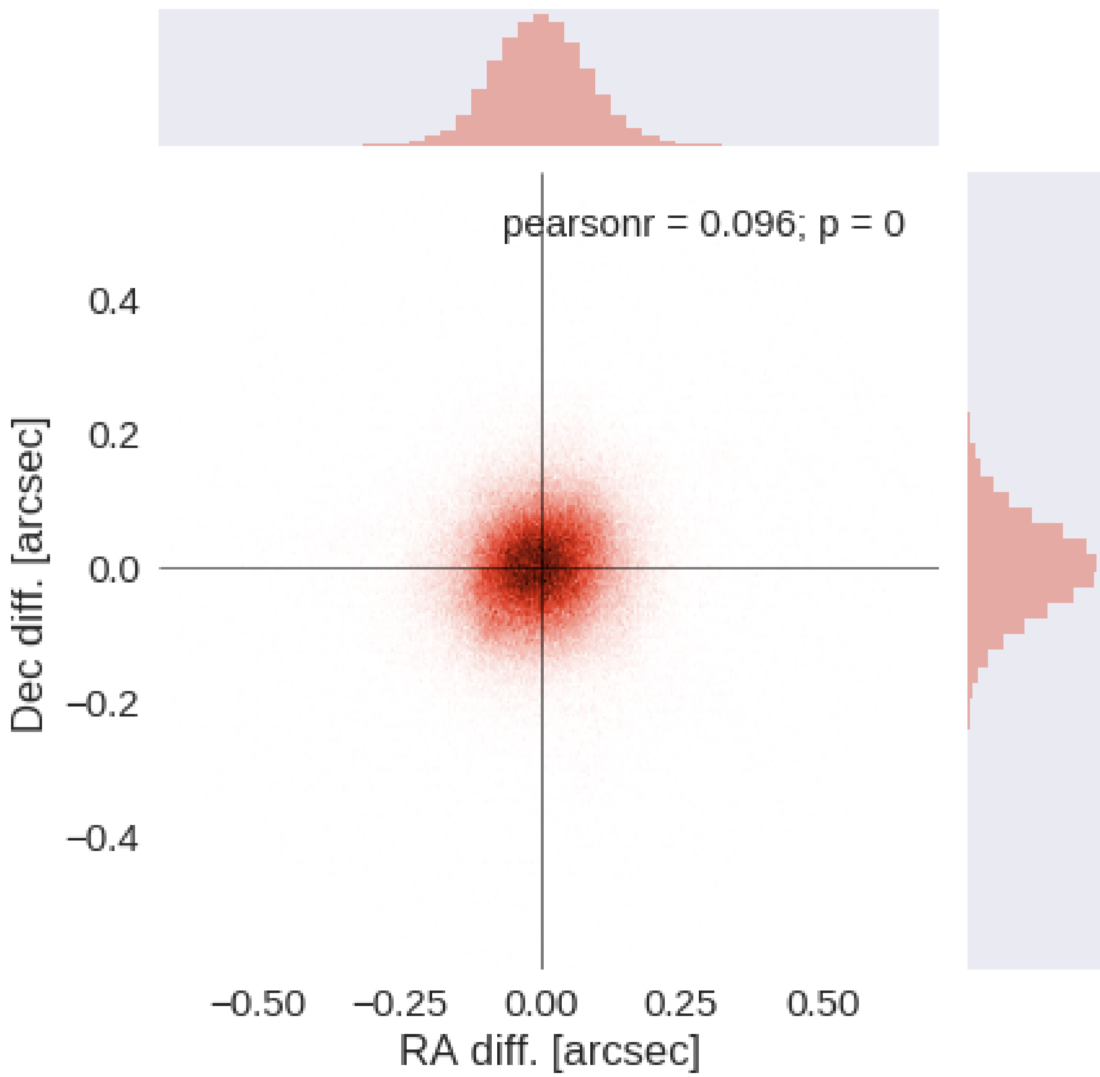
1.6 III - Astrometry correction

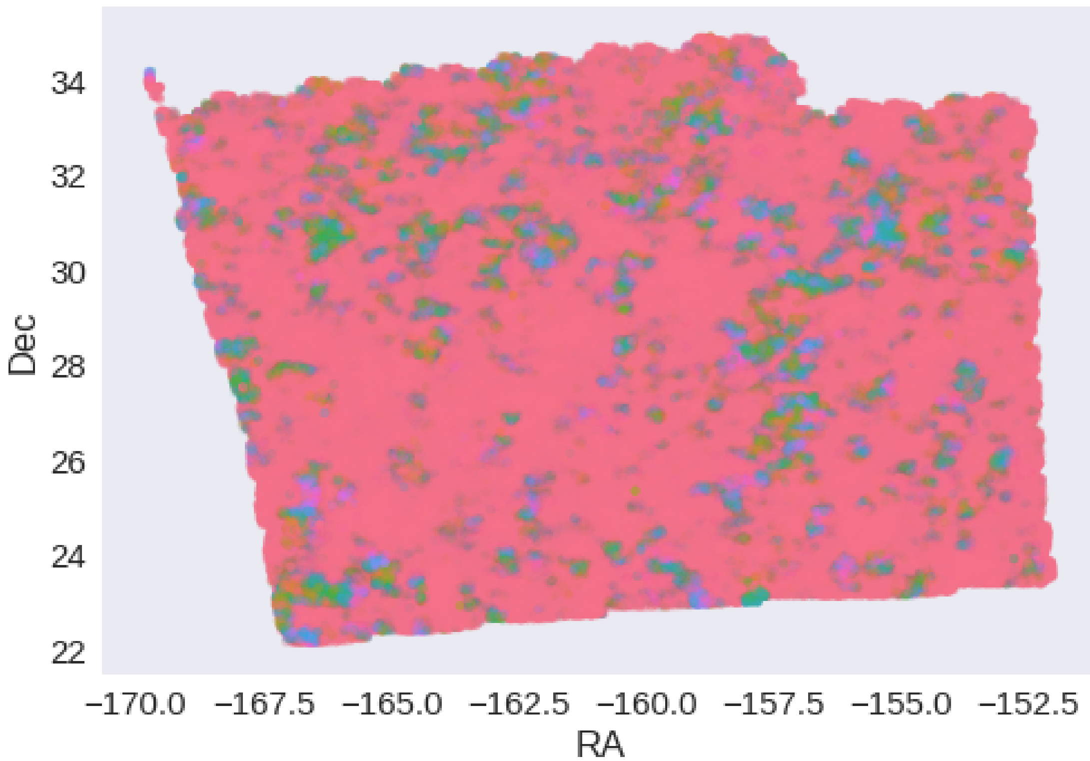
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.


pearsonr = 0.094; p = 0
0.4
0.2
0.0
−0.2
−0.4
Dec diff. [arcsec]
−0.50
−0.25
0.00
0.25
0.50
RA diff. [arcsec]




```
RA correction: -0.0928667625998969 arcsec
Dec correction: -0.059883860679832424 arcsec
```


pearsonr = 0.096; p = 0
0.4
0.2
0.0
−0.2
−0.4
Dec diff. [arcsec]
−0.50
−0.25
0.00
0.25
0.50
RA diff. [arcsec]



1.7 IV - Flagging Gaia objects

```
363563 sources flagged.
```

2 V - Saving to disk

1.2_PanSTARRS

March 8, 2018

1 HATLAS-NGP 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 `uniquePspsSTid` 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.

We take the 'F' photometry from the chi-squared image priors.

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 where done between 2010 and 2015 (ref).

```
This notebook was run with herschelhelp_internal version:
44f1ae0 (Thu Nov 30 18:27:54 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

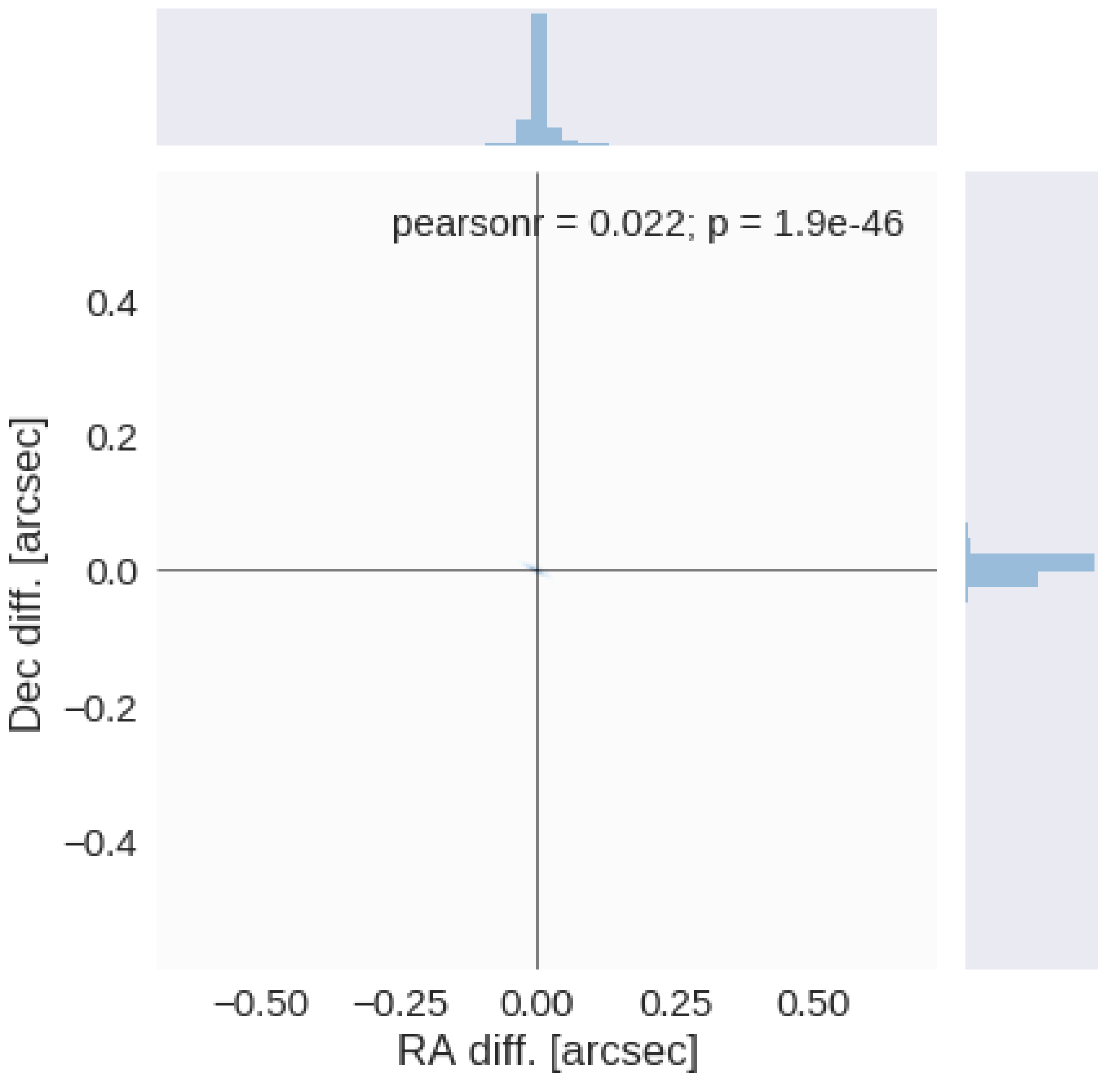
We remove duplicated objects from the input catalogues.

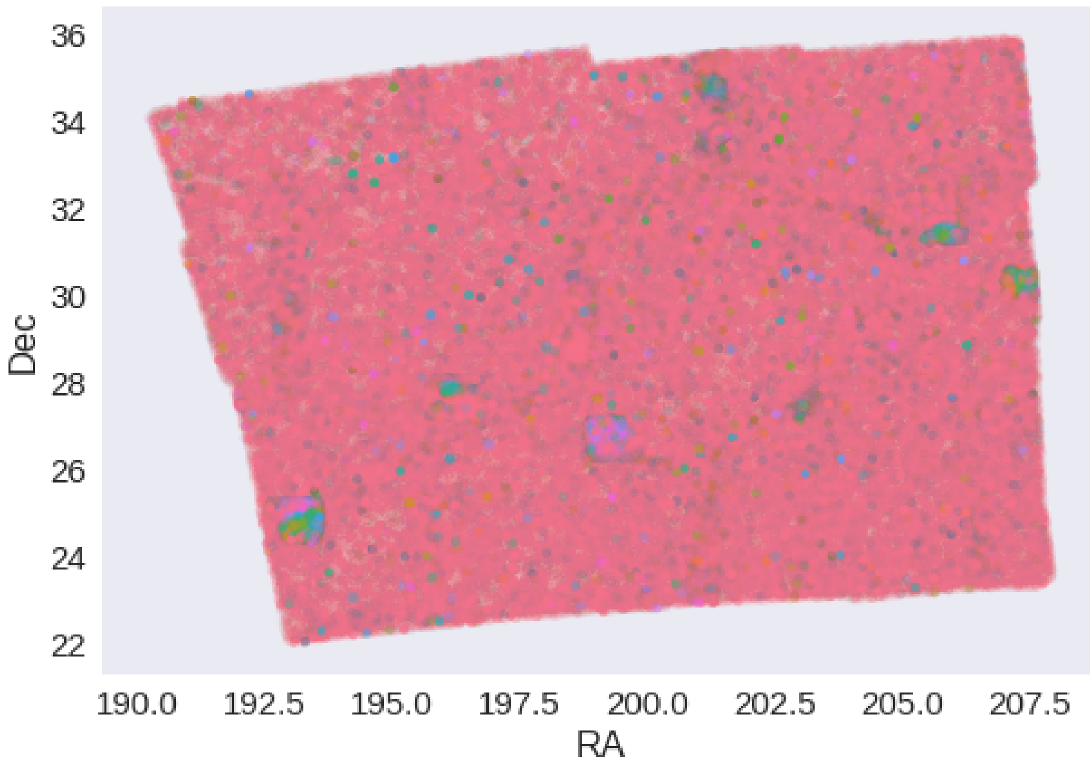
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10
Check the NumPy 1.11 release notes for more information.
  ma.MaskedArray.__setitem__(self, index, value)
```

```
The initial catalogue had 3268686 sources.
The cleaned catalogue has 3267763 sources (923 removed).
The cleaned catalogue has 923 sources flagged as having been cleaned
```

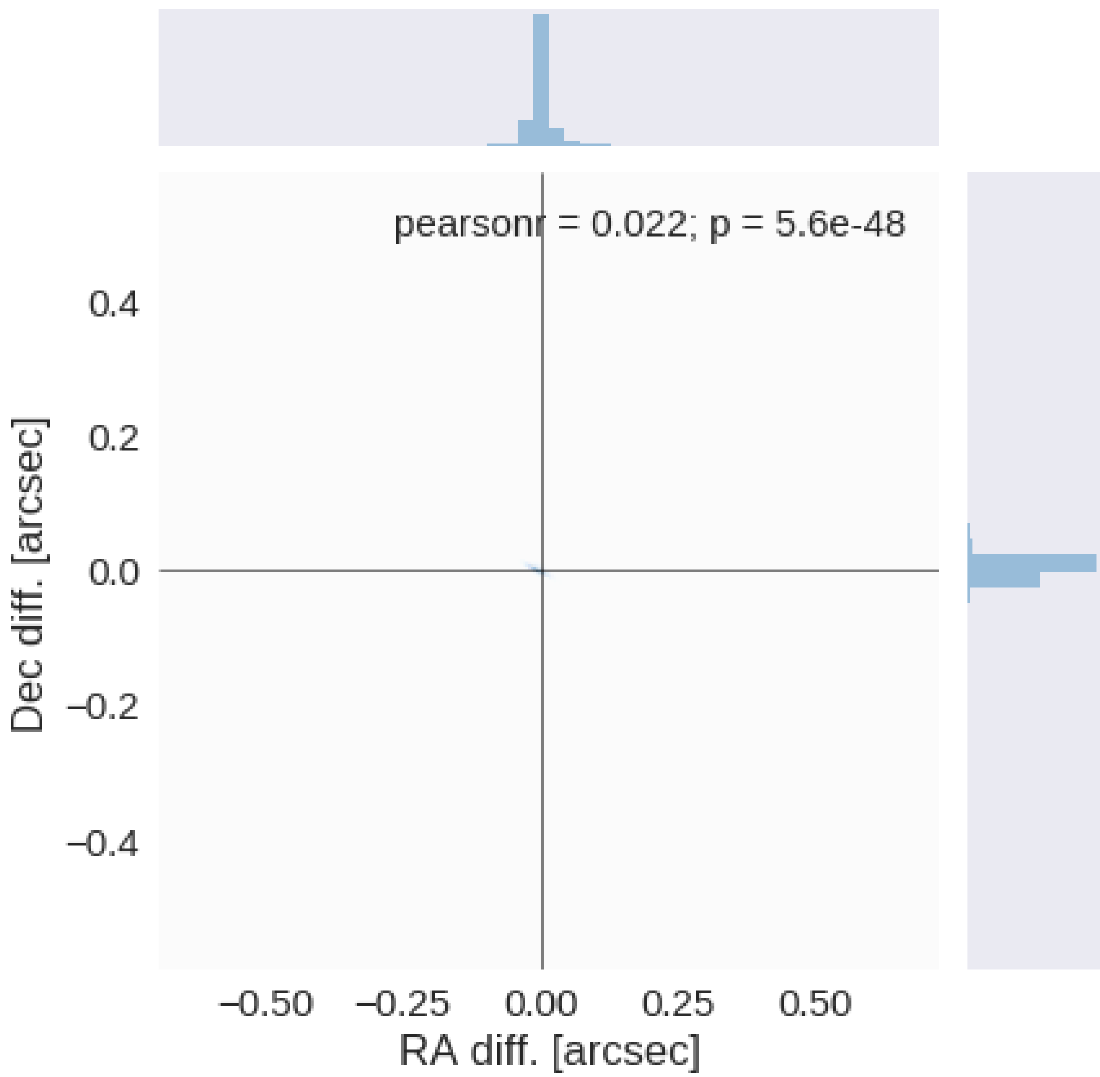
1.4 III - Astrometry correction

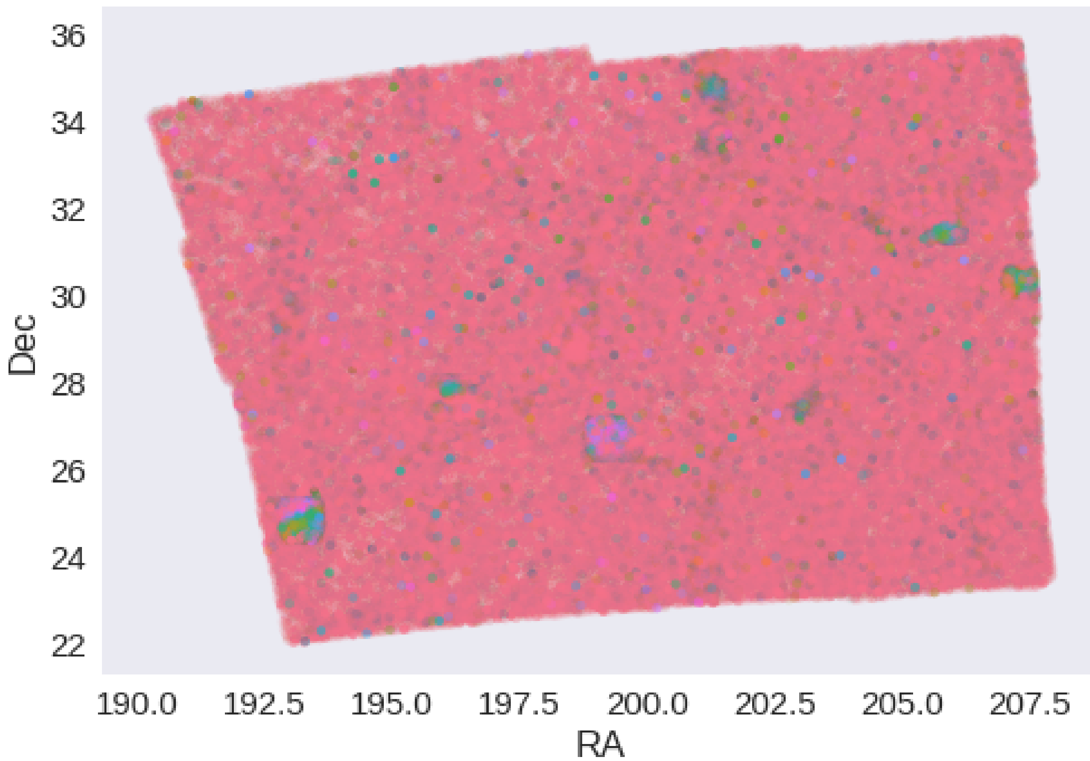
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.






```
RA correction: -0.0005492394507200515 arcsec
Dec correction: -0.00028873116377781116 arcsec
```


pearsonr = 0.022; p = 5.6e-48
0.4
0.2
0.0
−0.2
−0.4
Dec diff. [arcsec]
−0.50
−0.25
0.00
0.25
0.50
RA diff. [arcsec]



1.5 IV - Flagging Gaia objects

```
427003 sources flagged.
```

1.6 V - Flagging objects near bright stars

2 VI - Saving to disk

1.3_UKIDSS-LAS

March 8, 2018

1 HATLAS-NGP master catalogue

1.1 Preparation of UKIRT Infrared Deep Sky Survey / Large Area Survey (UKIDSS/LAS)

Information about UKIDSS can be found at http://www.ukidss.org/surveys/surveys.html

The catalogue comes from `dmu0_UKIDSS-LAS`.

In the catalogue, we keep:

- The identifier (it's unique in the catalogue);
- The position;
- The stellarity;
- The magnitude for each band in aperture 3 (2 arcsec).
- The hall magnitude is described as the total magnitude.

J band magnitudes are available in two eopchs. We take the first arbitrarily.

The magnitudes are "*Vega like*". The AB offsets are given by Hewett *et al.* (2016):

Band	AB offset
Y	0.634
J	0.938
H	1.379
K	1.900

Each source is associated with an epoch. These range between 2005 and 2007. We take 2006 for the epoch.

```
This notebook was run with herschelhelp_internal version:
44f1ae0 (Thu Nov 30 18:27:54 2017 +0000)
```

1.2 I - Column selection

```
WARNING: UnitsWarning: 'RADIANS' did not parse as fits unit: At col 0, Unit 'RADIANS' not suppor
```

```
/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]: &lt;IPython.core.display.HTML object&gt;
```

1.3 II - Removal of duplicated sources

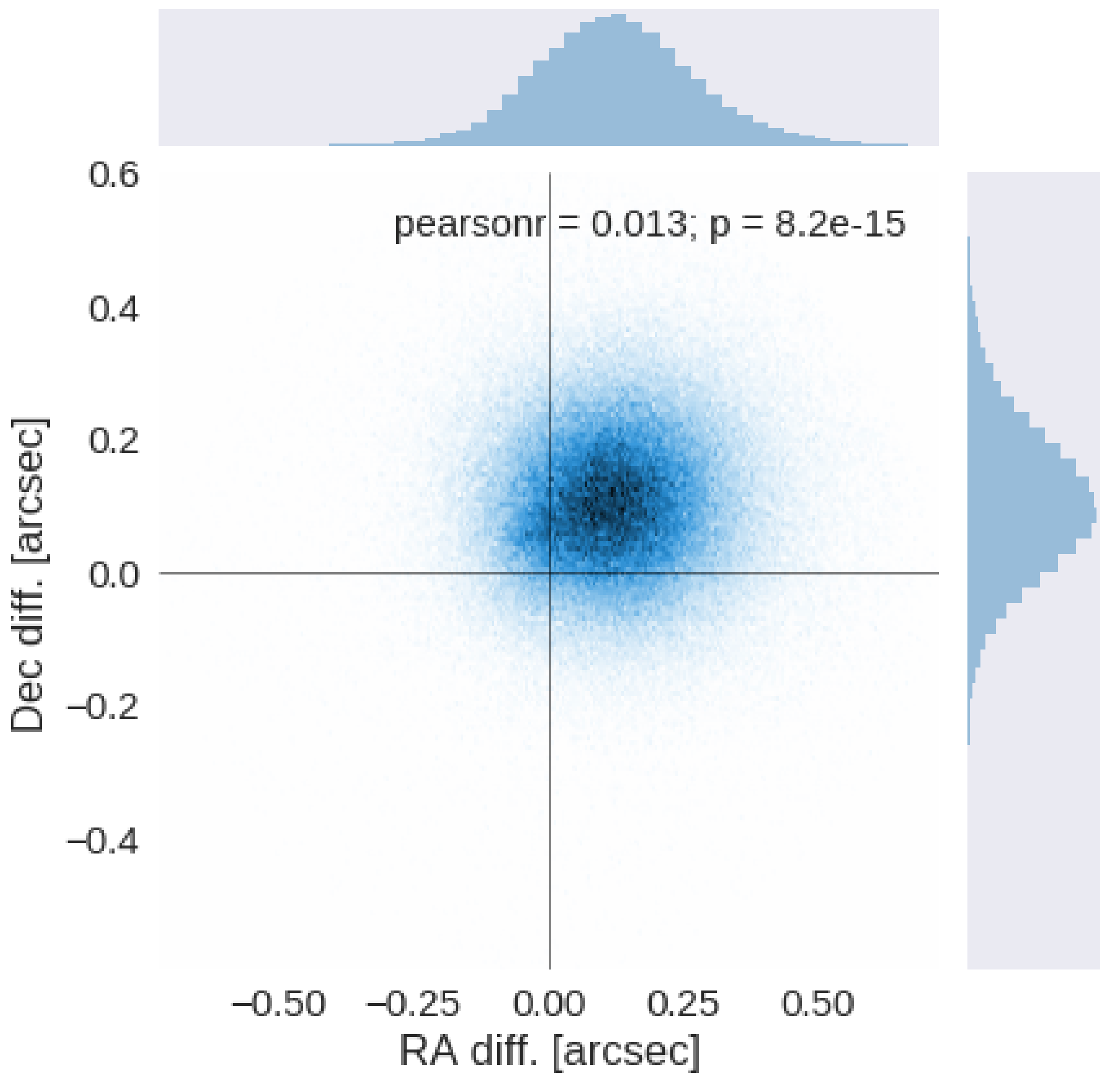
We remove duplicated objects from the input catalogues.

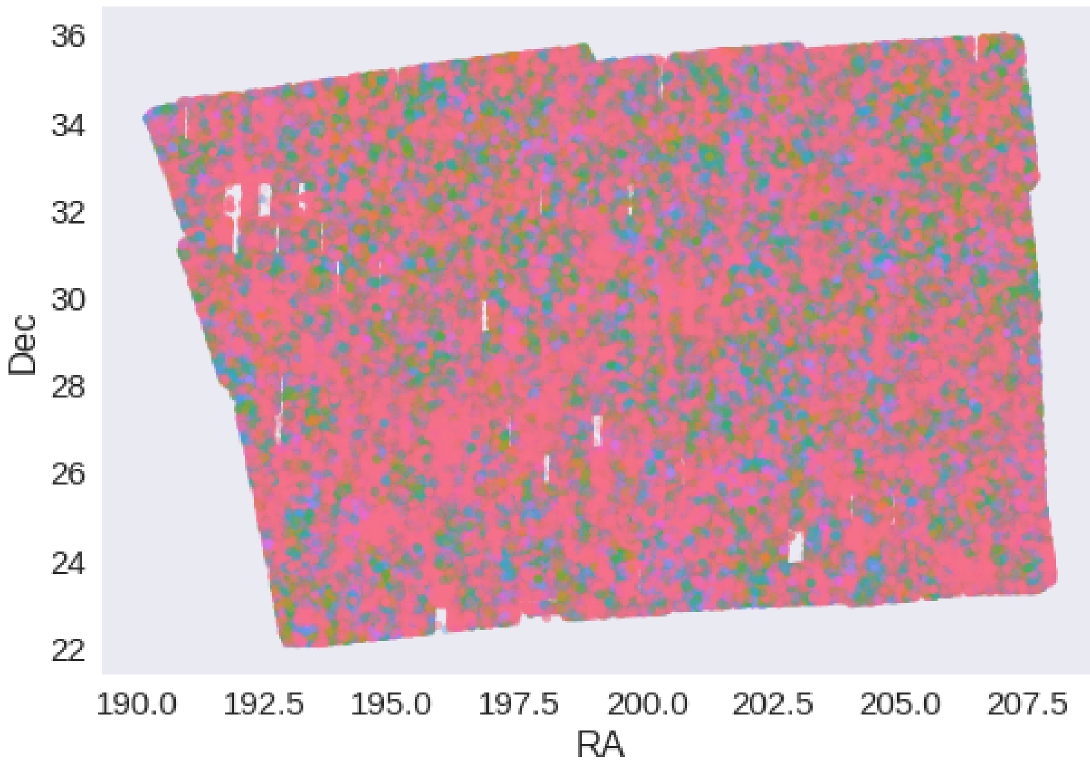
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10
Check the NumPy 1.11 release notes for more information.
  ma.MaskedArray.__setitem__(self, index, value)


The initial catalogue had 2517074 sources.
The cleaned catalogue has 2515473 sources (1601 removed).
The cleaned catalogue has 1584 sources flagged as having been cleaned
```

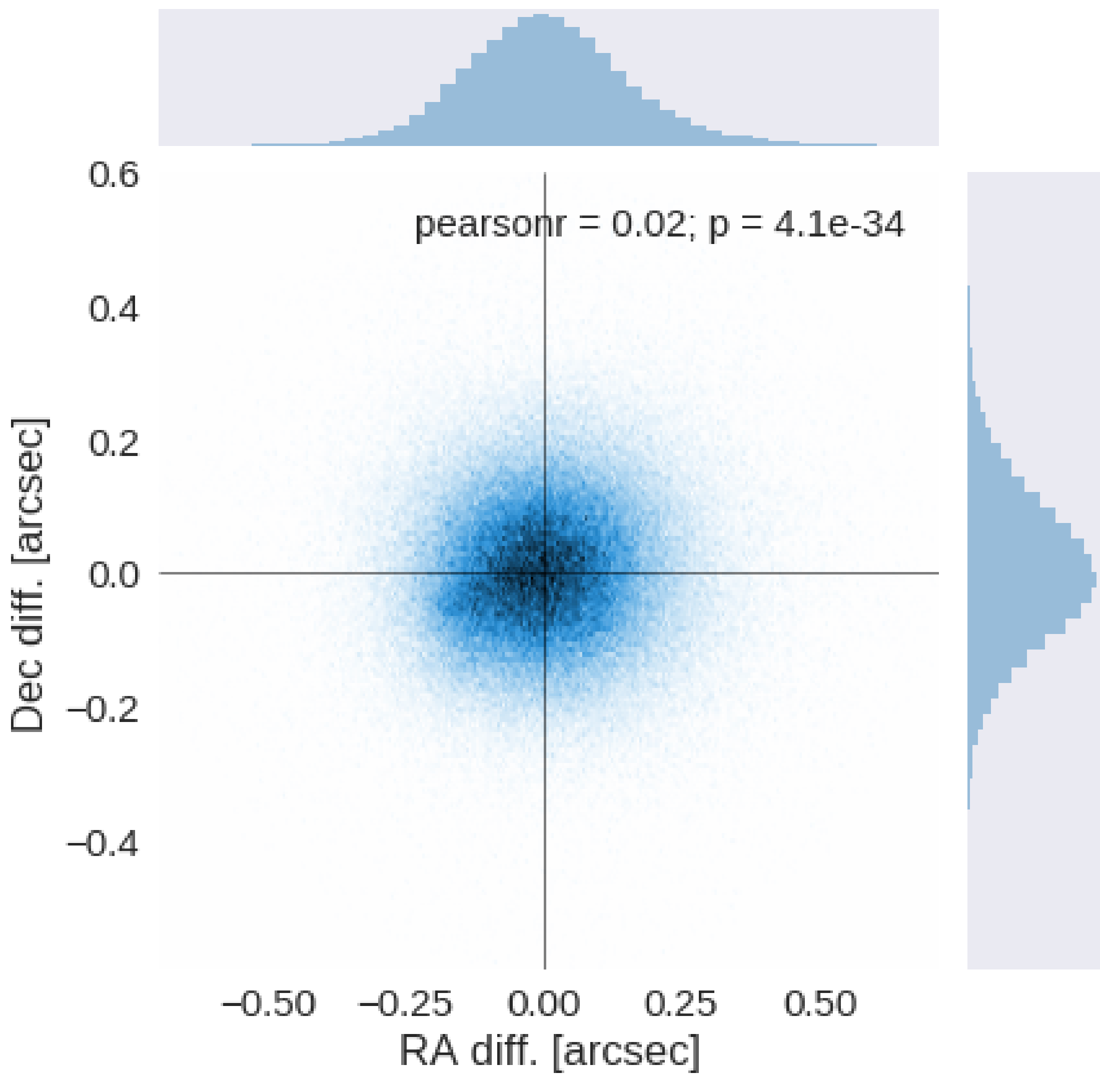
1.4 III - Astrometry correction

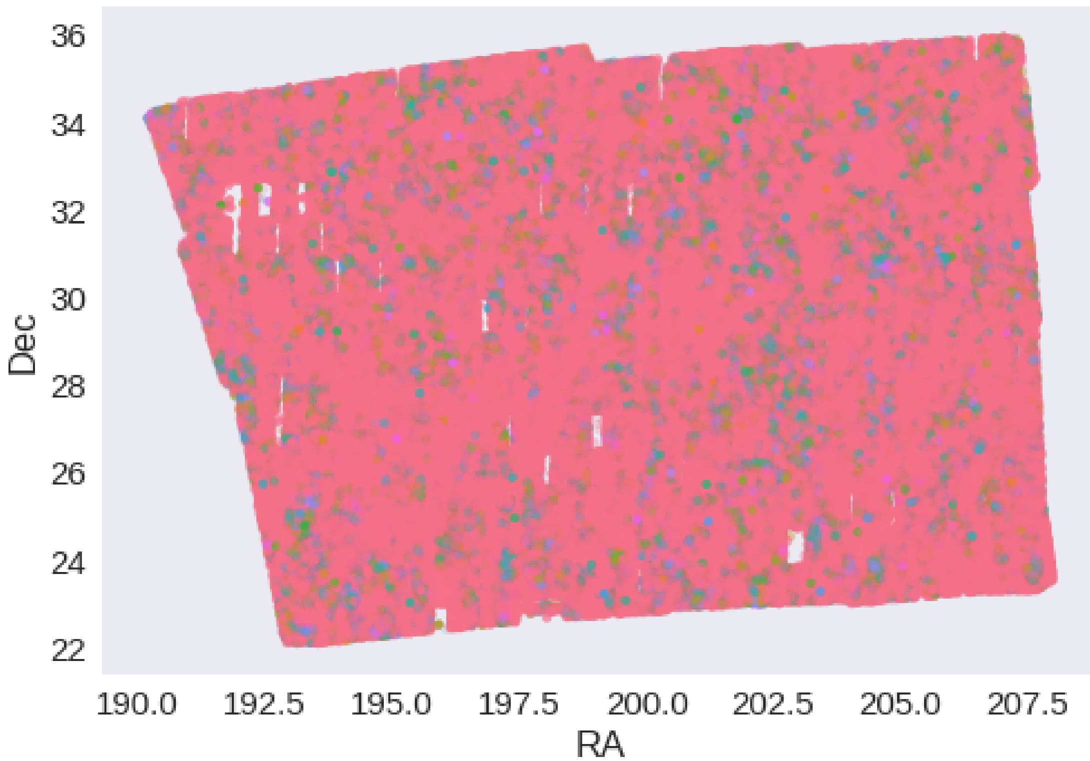
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.


pearsonr = 0.013; p = 8.2e-15
0.6
0.4
0.2
0.0
−0.2
−0.4
Dec diff. [arcsec]
−0.50
−0.25
0.00
0.25
0.50
RA diff. [arcsec]




```
RA correction: -0.11966463496264623 arcsec
Dec correction: -0.10092503875185344 arcsec
```


pearsonr = 0.02; p = 4.1e-34
0.6
0.4
0.2
0.0
−0.2
−0.4
Dec diff. [arcsec]
−0.50
−0.25
0.00
0.25
0.50
RA diff. [arcsec]



1.5 IV - Flagging Gaia objects

```
374866 sources flagged.
```

1.6 V - Flagging objects near bright stars

2 VI - Saving to disk

1.4_LegacySurvey

March 8, 2018

1 NGP master catalogue

1.1 Preparation of Legacy Survey data

The catalogue comes from `dmu0_LegacySurvey`.

In the catalogue, we keep:

- The identifier (it's unique in the catalogue);
- The position;
- The stellarity;
- The aperture fluxes. Are these aperture corrected?
- The kron magnitude to be used as total magnitude (no "auto" magnitude is provided).

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:
2018-02-19 18:39:32.352023
```

```
WARNING: UnitsWarning: '1/deg^2' did not parse as fits unit: Numeric factor not supported by FIT
WARNING: UnitsWarning: 'nanomaggy' did not parse as fits unit: At col 0, Unit 'nanomaggy' not su
WARNING: UnitsWarning: '1/nanomaggy^2' did not parse as fits unit: Numeric factor not supported
WARNING: UnitsWarning: '1/arcsec^2' did not parse as fits unit: Numeric factor not supported by
```

1.2 I - Aperture correction

To compute aperture correction we need to dertermine two parametres: the target aperture and the range of magnitudes for the stars that will be used to compute the correction.

Target aperture: To determine the target aperture, we simulate a curve of growth using the provided apertures and draw two figures:

- The evolution of the magnitudes of the objects by plotting on the same plot aperture number vs the mean magnitude.

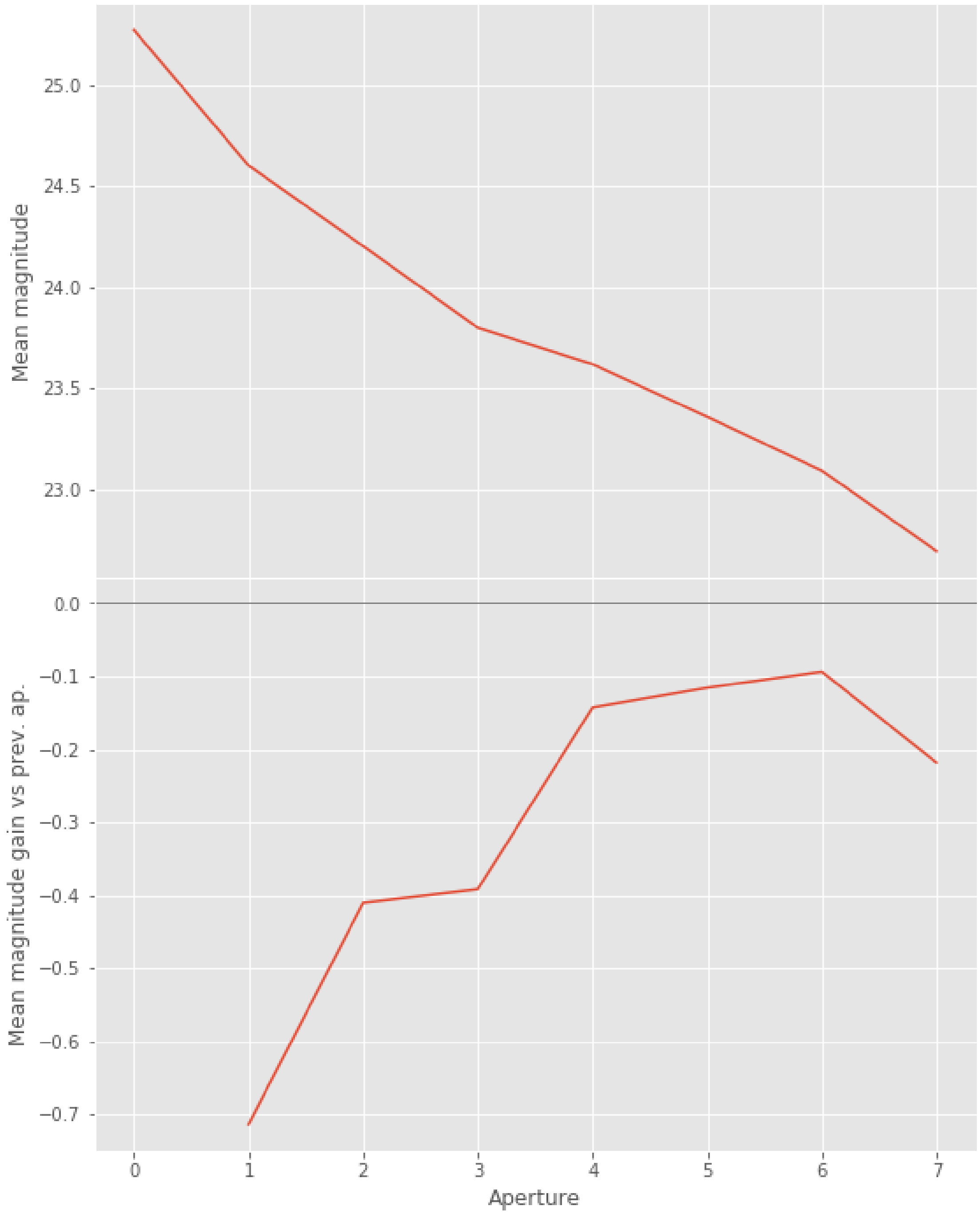
- The mean gain (loss when negative) of magnitude is each aperture compared to the previous (except for the first of course).

As target aperture, we should use the smallest (i.e. less noisy) aperture for which most of the flux is captures.

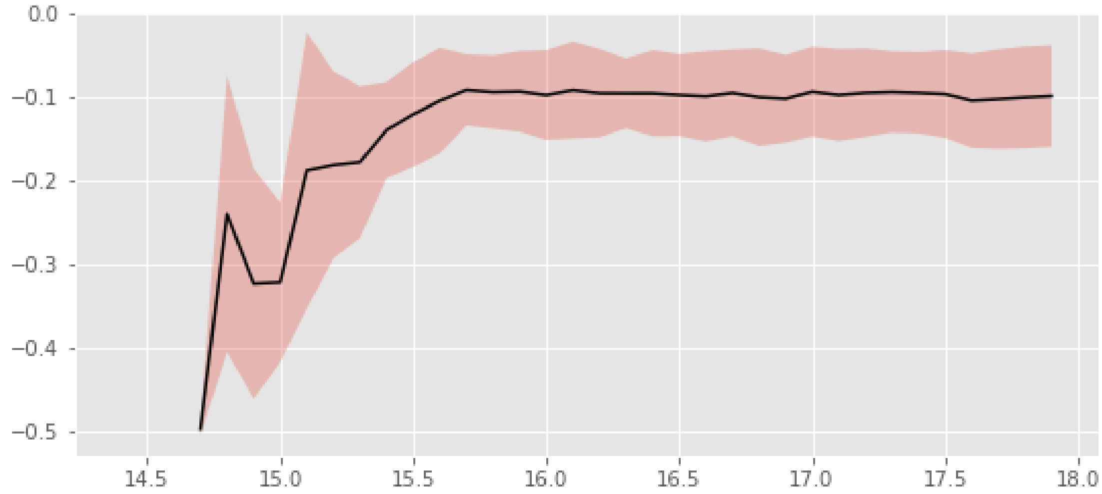
Magnitude range: To know what limits in aperture to use when doing the aperture correction, we plot for each magnitude bin the correction that is computed and its RMS. We should then use the wide limits (to use more stars) where the correction is stable and with few dispersion.

```
/opt/herschelhelp_internal/herschelhelp_internal/utils.py:76: RuntimeWarning: divide by zero enc
  magnitudes = 2.5 * (23 - np.log10(fluxes)) - 48.6
/opt/herschelhelp_internal/herschelhelp_internal/utils.py:80: RuntimeWarning: invalid value enco
  errors = 2.5 / np.log(10) * errors_on_fluxes / fluxes
/opt/herschelhelp_internal/herschelhelp_internal/utils.py:76: RuntimeWarning: invalid value enco
  magnitudes = 2.5 * (23 - np.log10(fluxes)) - 48.6
```

1.2.1 I.a - g band



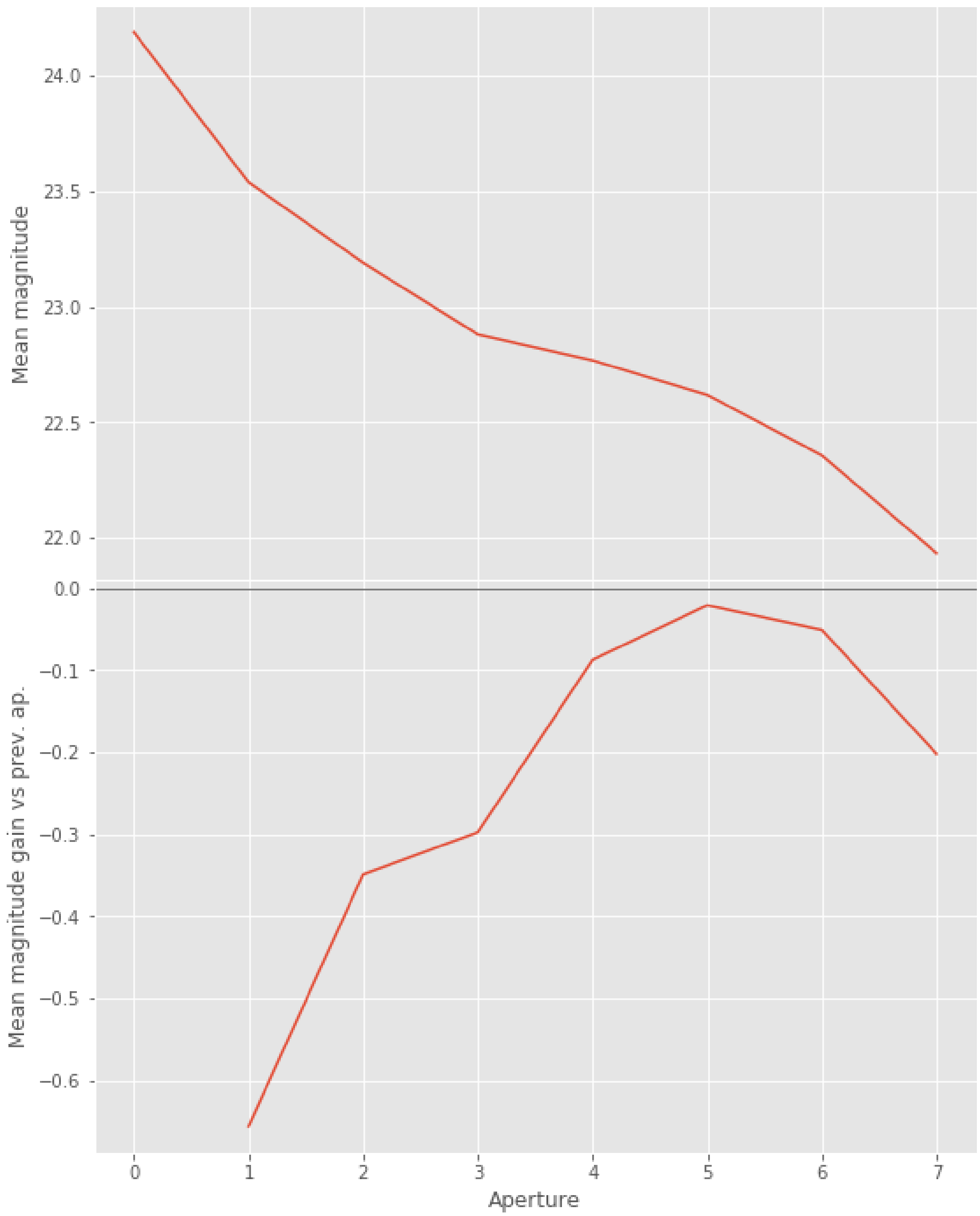
We will use aperture 5 as target.



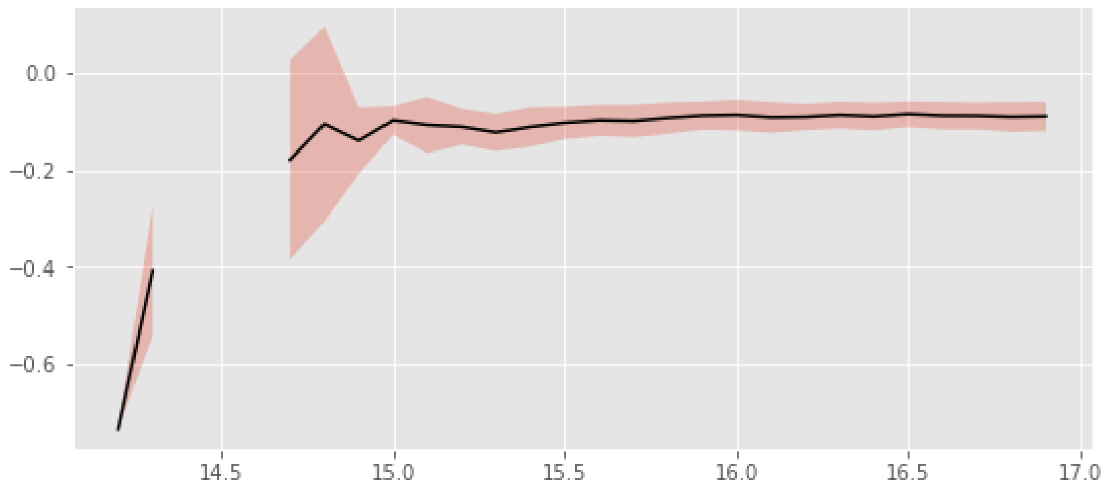
We will use magnitudes between 17.0 and 18.5

```
Aperture correction for g band:
Correction: -0.09936359051312138
Number of source used: 7175
RMS: 0.055409976464209754
```

1.2.2 I.b - r band



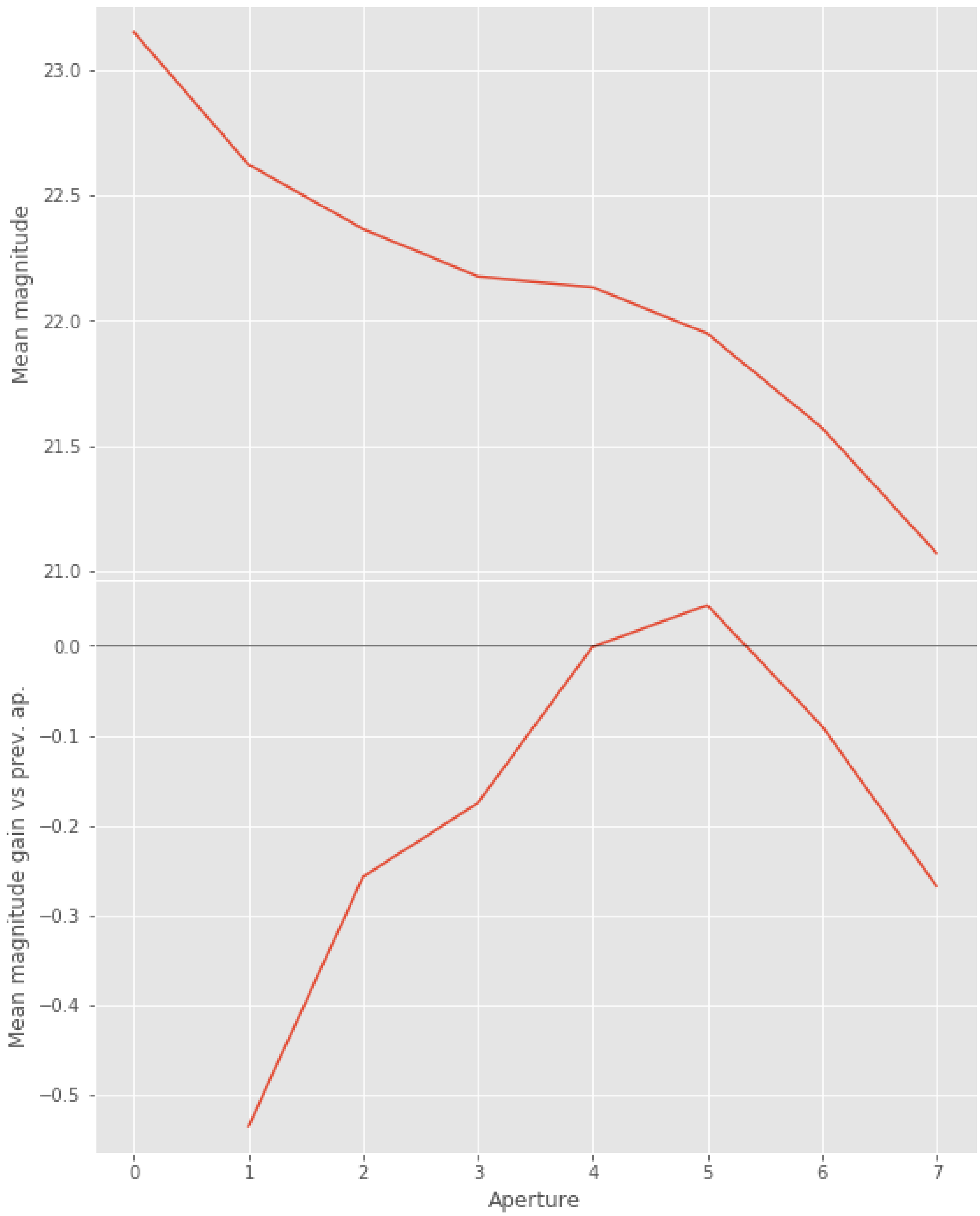
We will use aperture 5 as target.



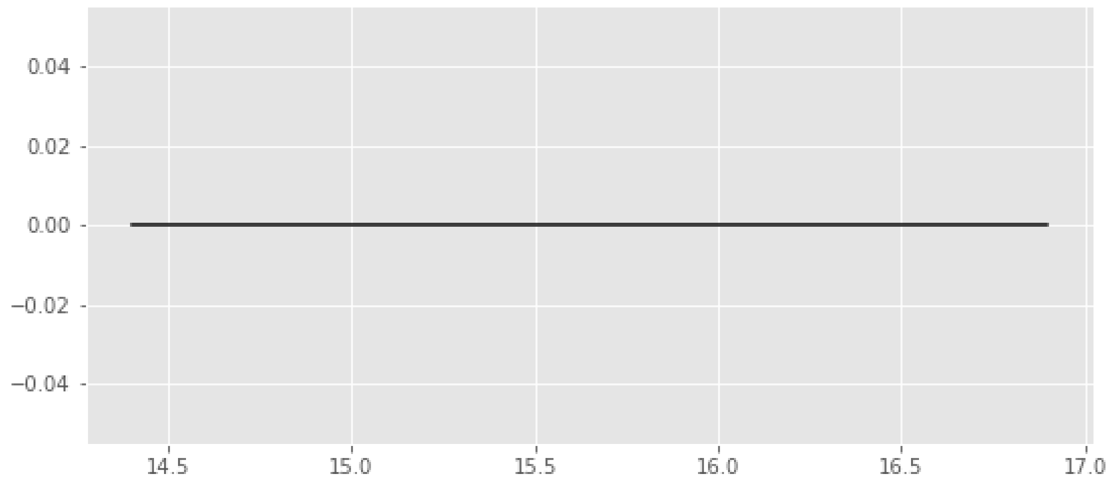
We use magnitudes between 17.0 and 18.5.

```
Aperture correction for r band:
Correction: -0.0879437624595738
Number of source used: 12780
RMS: 0.02870466417671864
```

1.2.3 I.c - z band



We will use aperture 4 as target.



We use magnitudes between 16.0 and 17.5.

```
Aperture correction for z band:
Correction: -0.05423560441600017
Number of source used: 9512
RMS: 0.03337385310736833
```

1.3 II - Stellarity

Legacy Survey does not provide a 0 to 1 stellarity so we replace items flagged as PSF accpording to the following table:

$$P(star) = \frac{\prod_i P(star)_i}{\prod_i P(star)_i + \prod_i P(galaxy)_i}$$

where i is the band, and with using the same probabilities as UKDISS:

HSC flag	UKIDSS flag	Meaning	P(star)	P(galaxy)	P(noise)	P(saturated)
	-9	Saturated	0.0	0.0	5.0	95.0
	-3	Probable galaxy	25.0	70.0	5.0	0.0
	-2	Probable star	70.0	25.0	5.0	0.0
0	-1	Star	90.0	5.0	5.0	0.0
	0	Noise	5.0	5.0	90.0	0.0
1	+1	Galaxy	5.0	90.0	5.0	0.0

1.4 II - Column selection

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/ipykernel/__main__.py:19:
/opt/herschelhelp_internal/herschelhelp_internal/utils.py:76: RuntimeWarning: divide by zero enc
  magnitudes = 2.5 * (23 - np.log10(fluxes)) - 48.6
```

```
/opt/herschelhelp_internal/herschelhelp_internal/utils.py:76: RuntimeWarning: invalid value enco
  magnitudes = 2.5 * (23 - np.log10(fluxes)) - 48.6
/opt/herschelhelp_internal/herschelhelp_internal/utils.py:80: RuntimeWarning: divide by zero enc
  errors = 2.5 / np.log(10) * errors_on_fluxes / fluxes
```

Out[19]: <IPython.core.display.HTML object>

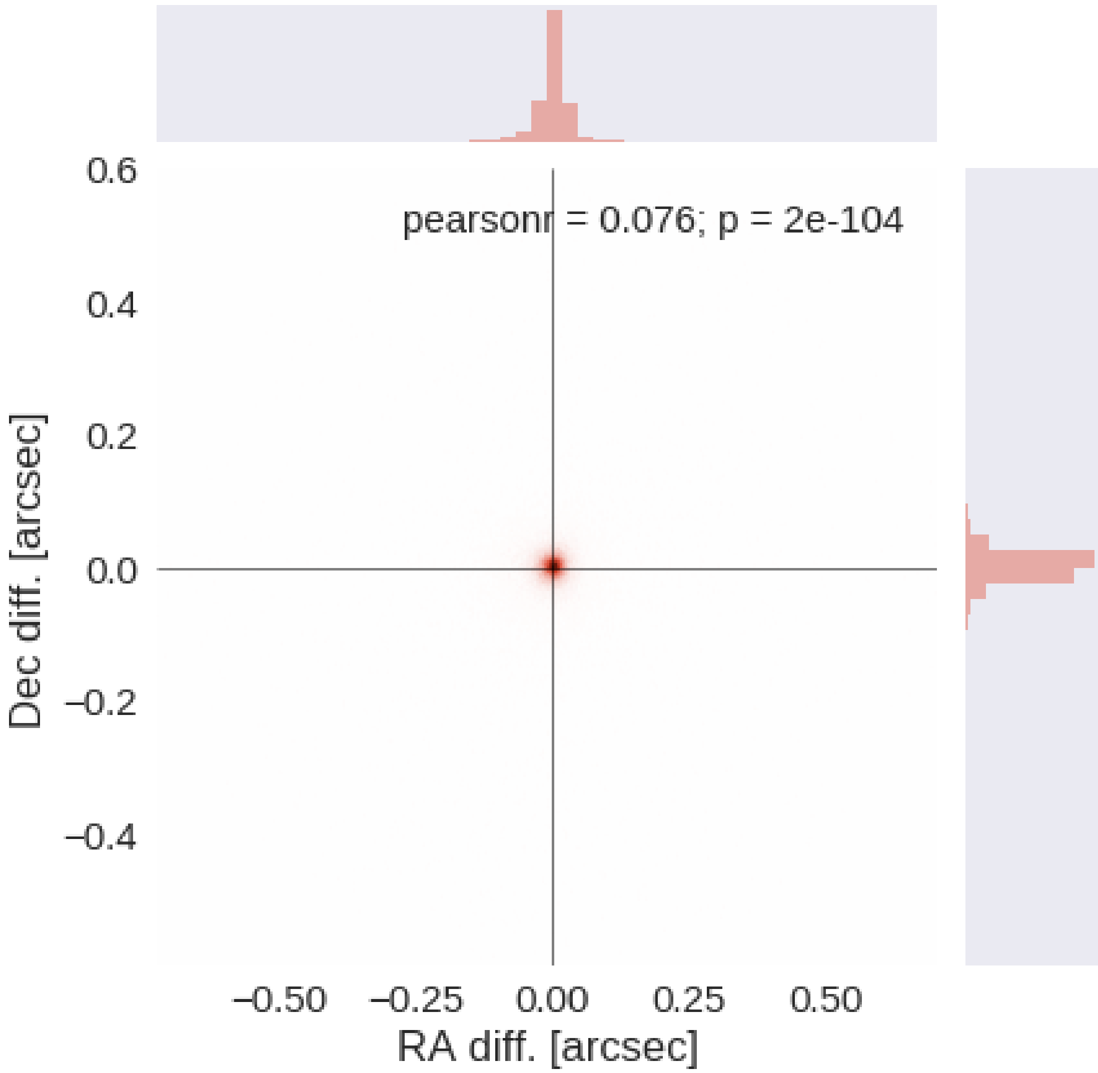
1.5 III - Removal of duplicated sources

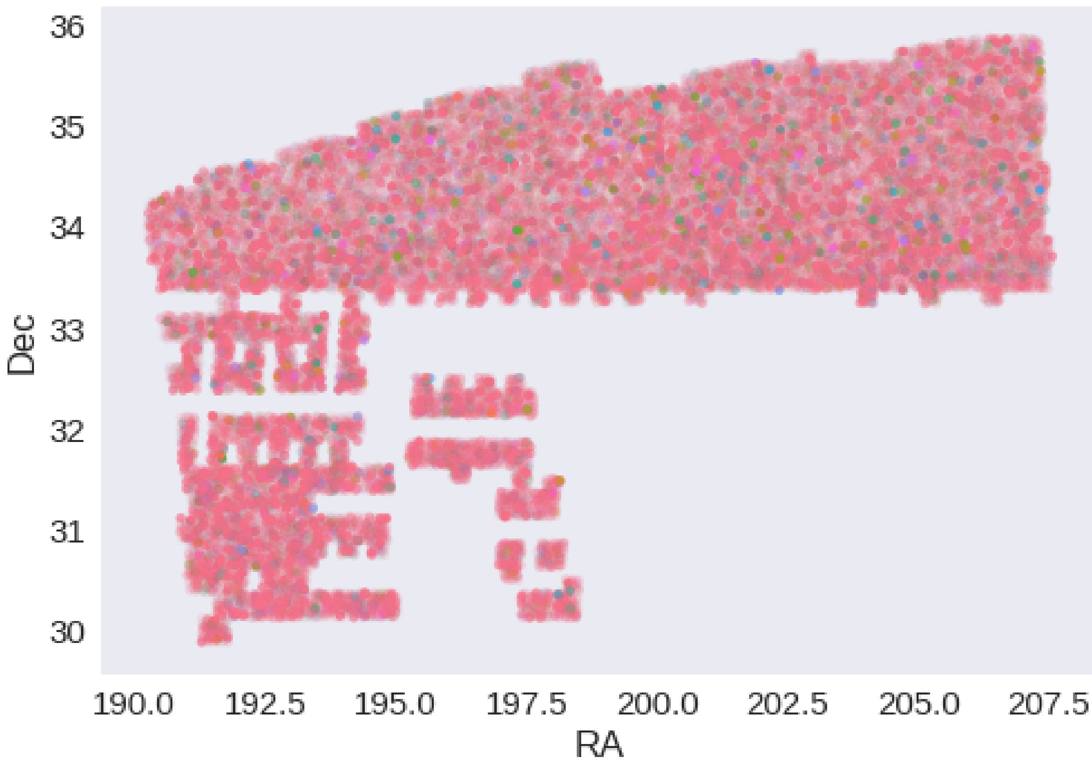
We remove duplicated objects from the input catalogues.

```
The initial catalogue had 1374183 sources.
The cleaned catalogue has 1348144 sources (26039 removed).
The cleaned catalogue has 25667 sources flagged as having been cleaned
```

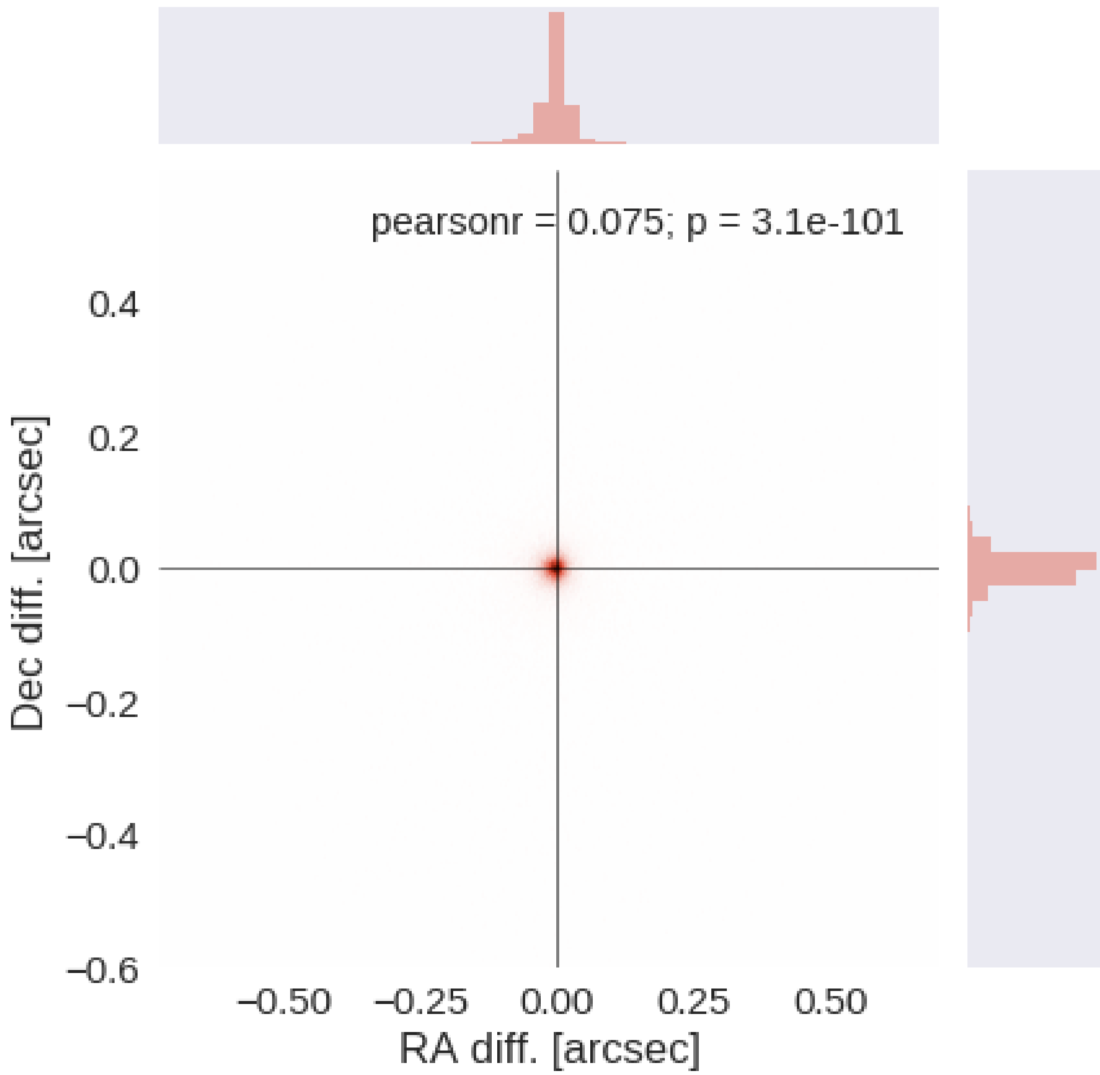
1.6 III - Astrometry correction

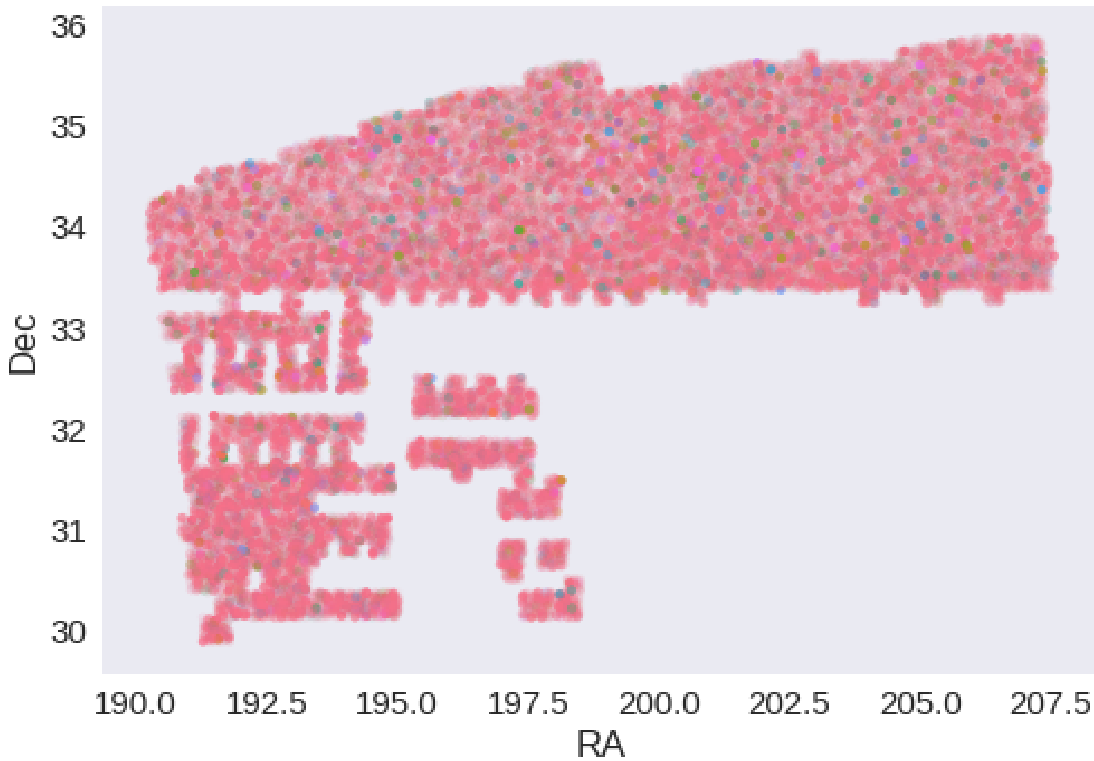
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.


pearsonr = 0.076; p = 2e-104
0.6
0.4
0.2
0.0
−0.2
−0.4
Dec diff. [arcsec]
−0.50
−0.25
0.00
0.25
0.50
RA diff. [arcsec]




```
RA correction: -0.003366076390420858 arcsec
Dec correction: -0.001721845239188724 arcsec
```


pearsonr = 0.075; p = 3.1e-101
0.4
0.2
0.0
−0.2
−0.4
−0.6
Dec diff. [arcsec]
−0.50
−0.25
0.00
0.25
0.50
RA diff. [arcsec]



1.7 IV - Flagging Gaia objects

```
83988 sources flagged.
```

2 V - Saving to disk

2_Merging

March 8, 2018

1 HATLAS-NGP master catalogue

This notebook presents the merge of the various pristine catalogues to produce HELP mater catalogue on GAMA-12.

```
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:
2018-02-19 18:48:57.801810
```

1.1 I - Reading the prepared pristine catalogues

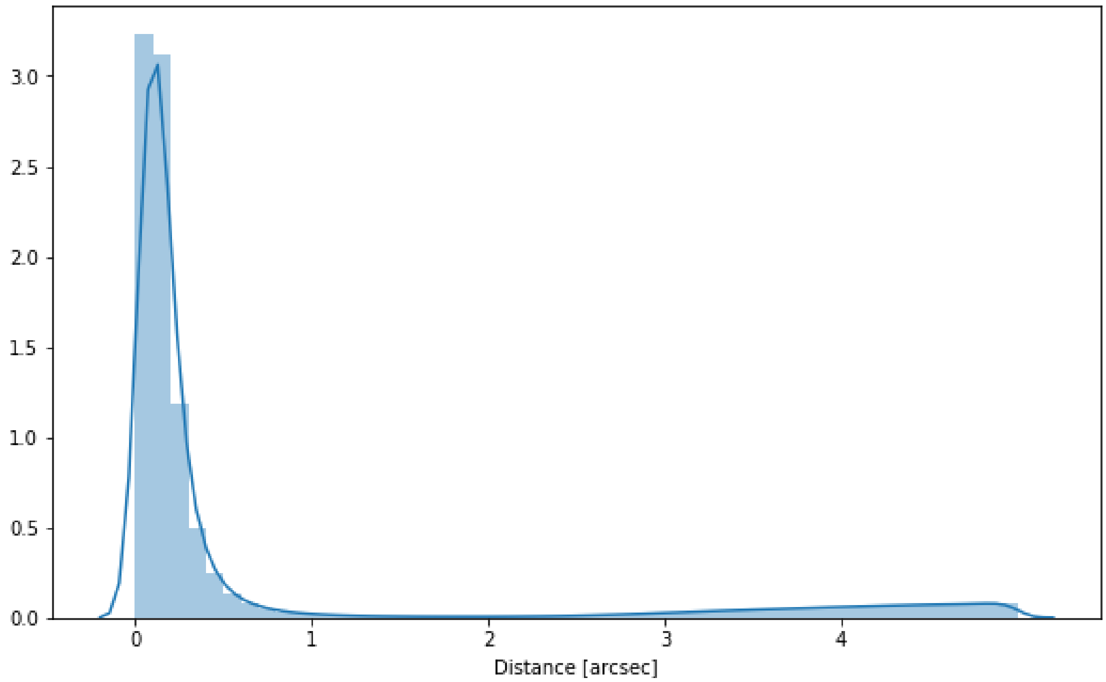
1.2 II - Merging tables

We first merge the optical catalogues and then add the infrared ones: DECaLS, HSC, KIDS, PanSTARRS, UKIDSS-LAS, and VISTA-VIKING.

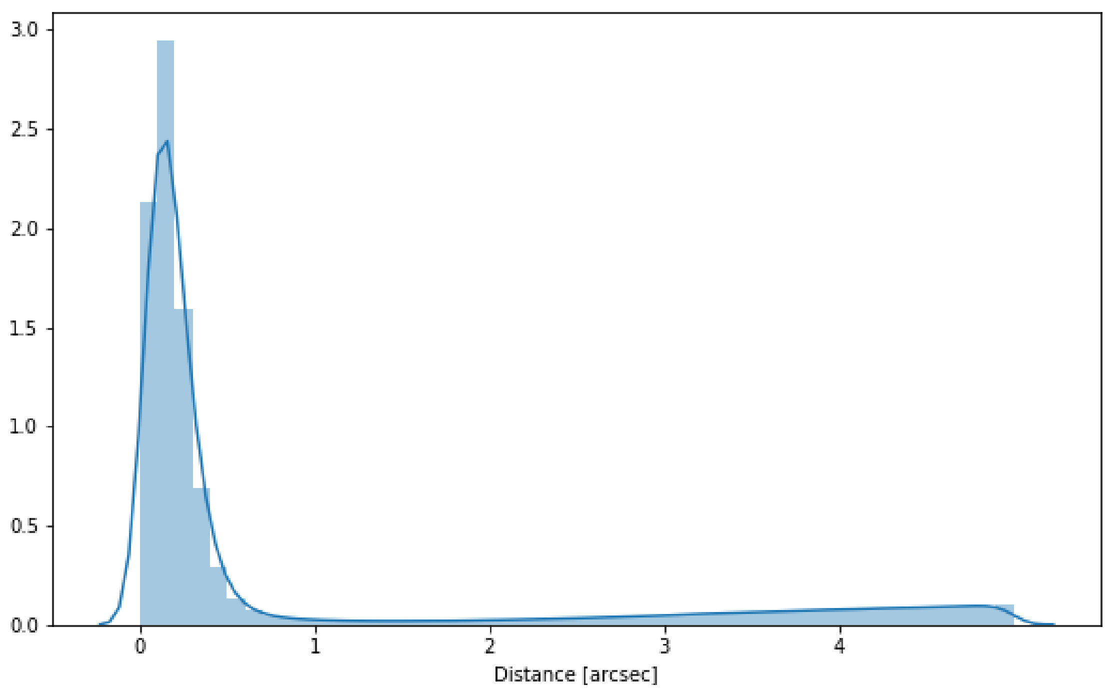
At every step, we look at the distribution of the distances to the nearest source in the merged catalogue to determine the best crossmatching radius.

1.2.1 DECaLS

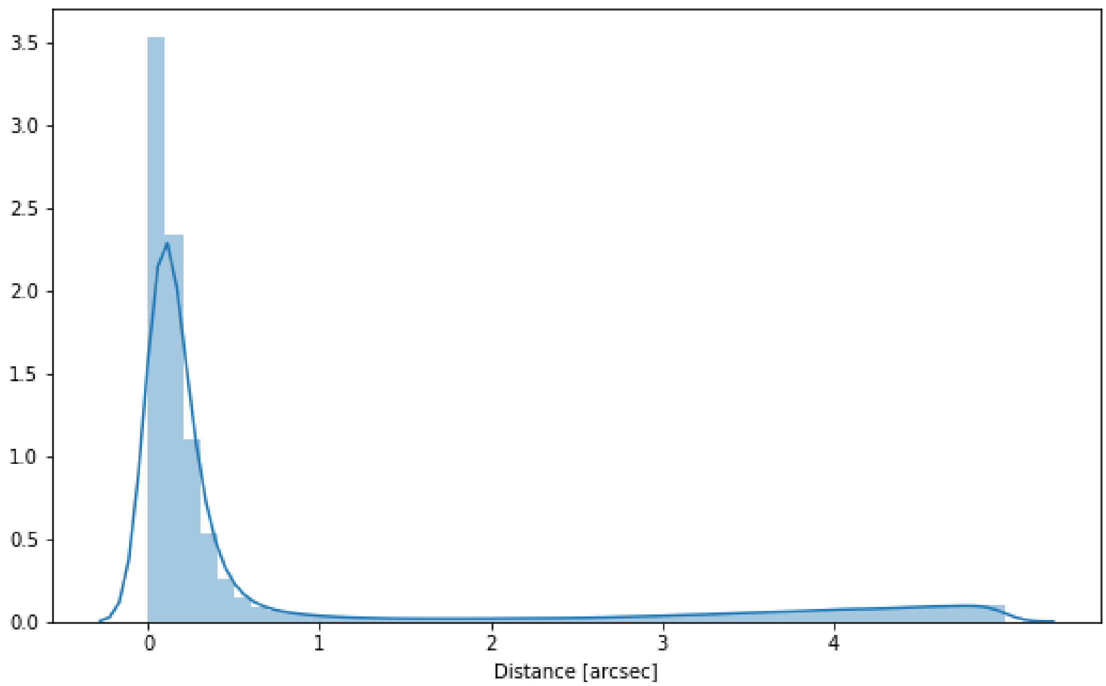
1.2.2 Add PanSTARRS



1.2.3 Add UKIDSS LAS



1.2.4 Add Legacy Survey



1.2.5 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[13]: <IPython.core.display.HTML object>
```

1.3 III - Merging flags and stellarity

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 stellarity columns indicating the probability (0 to 1) of each source being a star. We merge these columns taking the highest value.

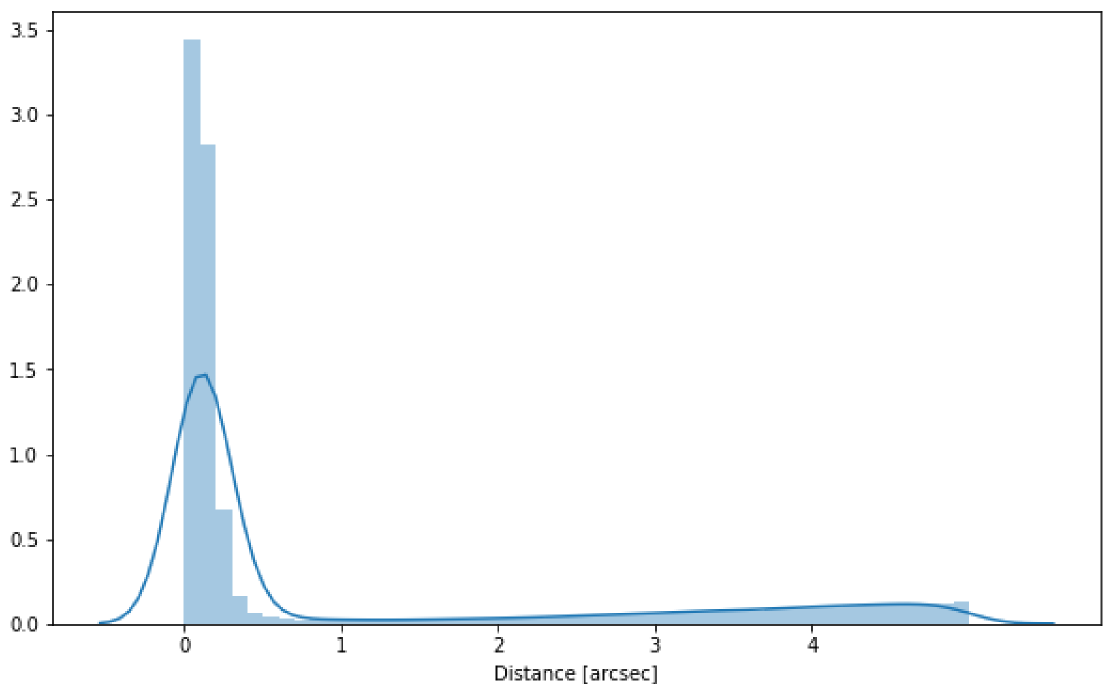
```
decals_stellarity, las_stellarity, legacy_stellarity
```

1.4 IV - Adding E(B-V) column

1.5 V - 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_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 by 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 de different depths in the catalogue we are using.

1.8 VIII.b Wavelength domain detection

We add a binary `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 catalogue. This can be used to easily get additional information from them.

```
['decals_id', 'ps1_id', 'las_id', 'legacy_id', 'help_id', 'specz_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: set()
```

3_Checks_and_diagnostics

March 8, 2018

1 HATLAS-NGP 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:
2018-02-19 22:02:36.996073
```

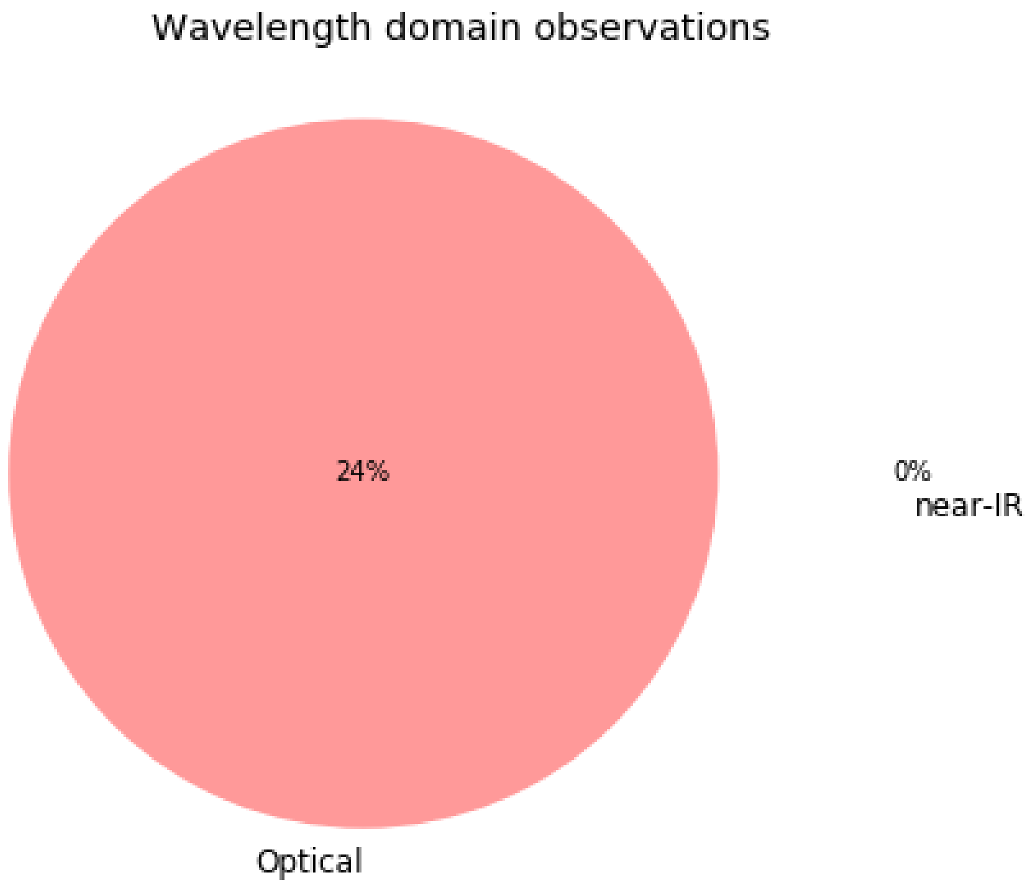
1.2 0 - Quick checks

```
/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)
```

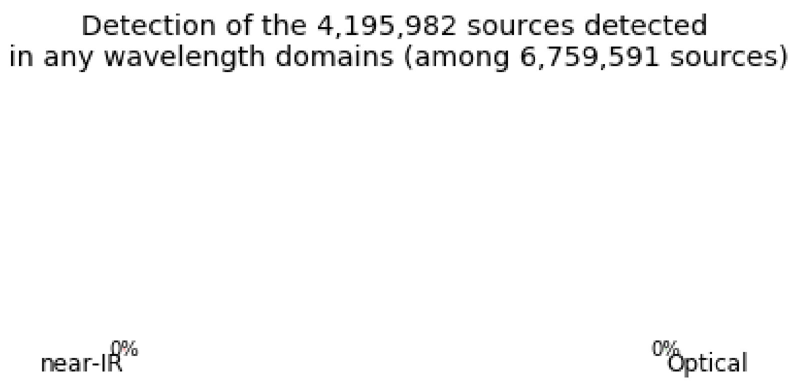
```
Table shows only problematic columns.
```

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

1.3 I - Summary of wavelength domains




```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/matplotlib_venn/_venn2.py:
  warnings.warn("Both circles have zero area")
```



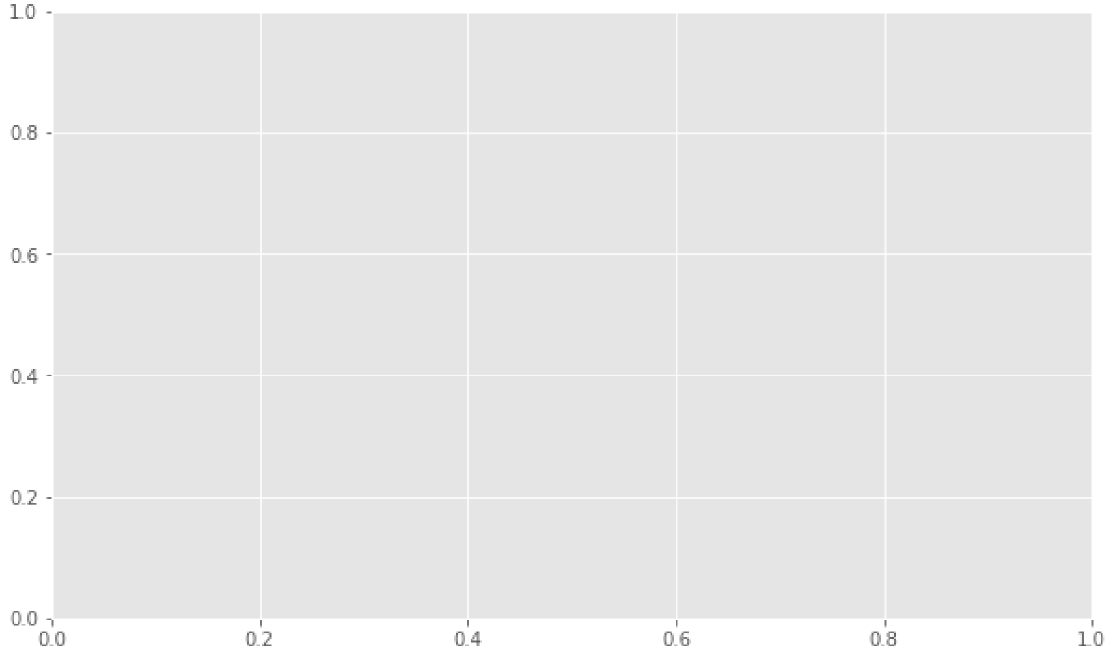
1.4 II - Comparing magnitudes in similar filters

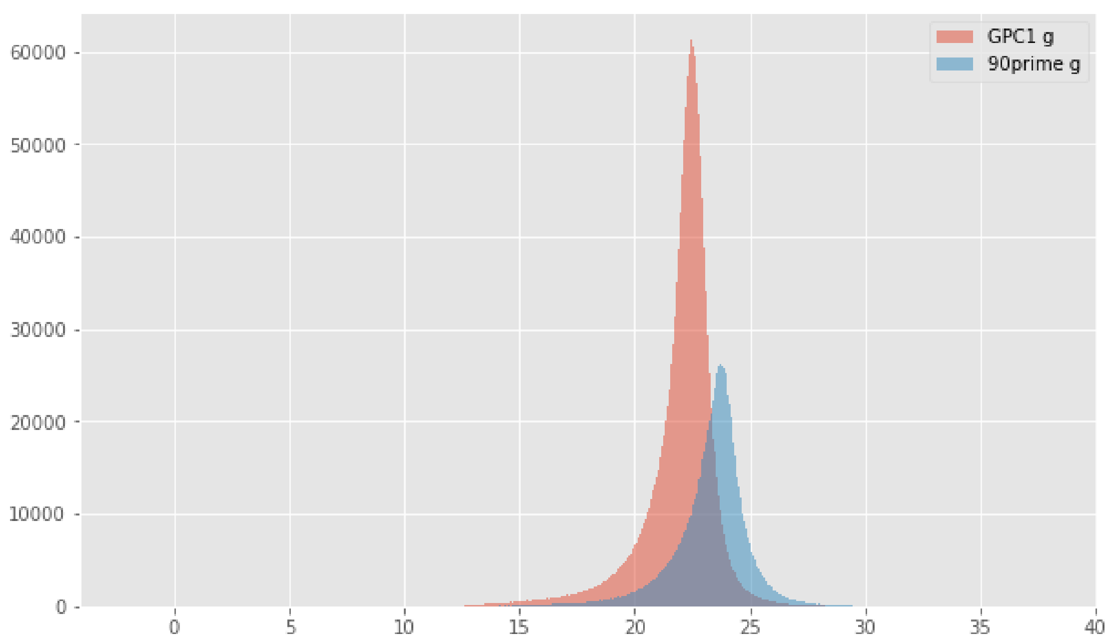
The master list if composed of several catalogues containing magnitudes in similar filters on different instruments. We are comparing the magnitudes in these corresponding filters.

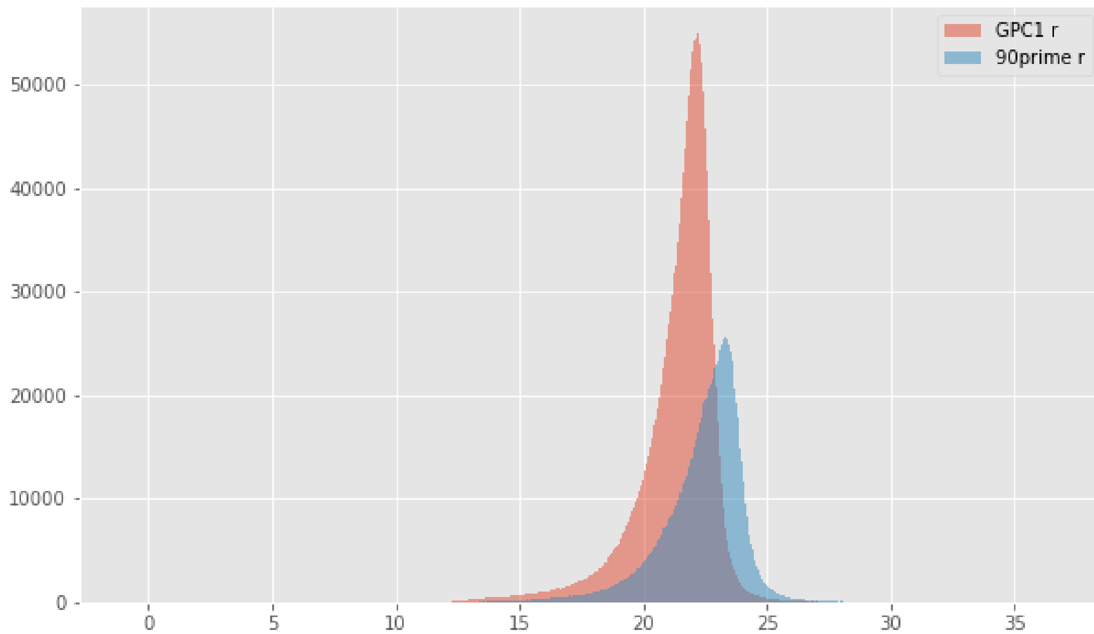
1.4.1 II.a - Comparing depths

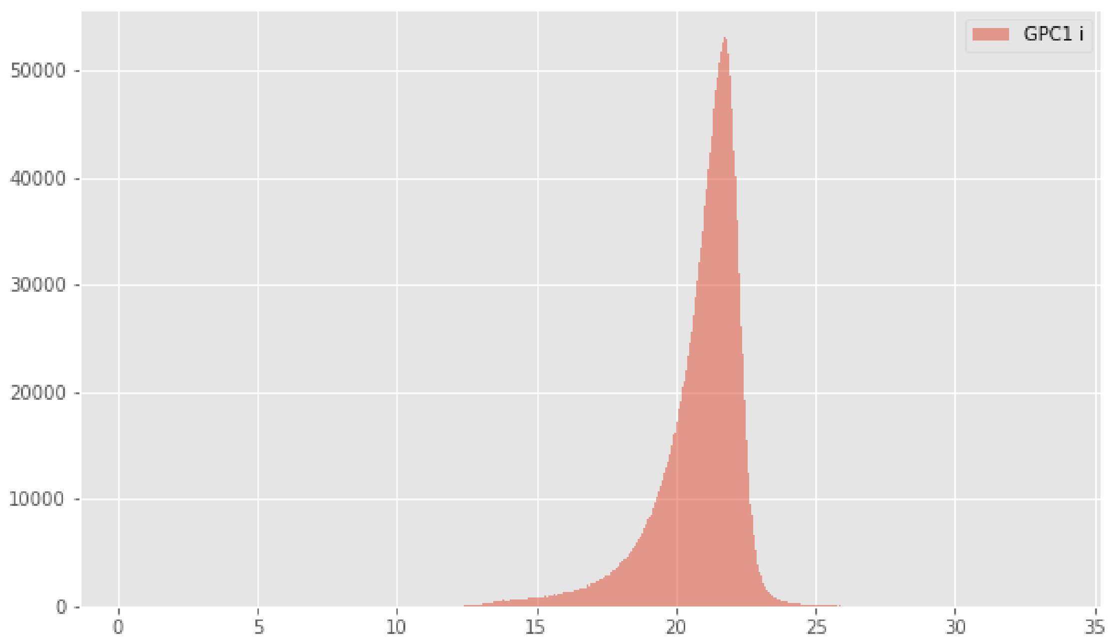
We compare the histograms of the total aperture magnitudes of similar bands.

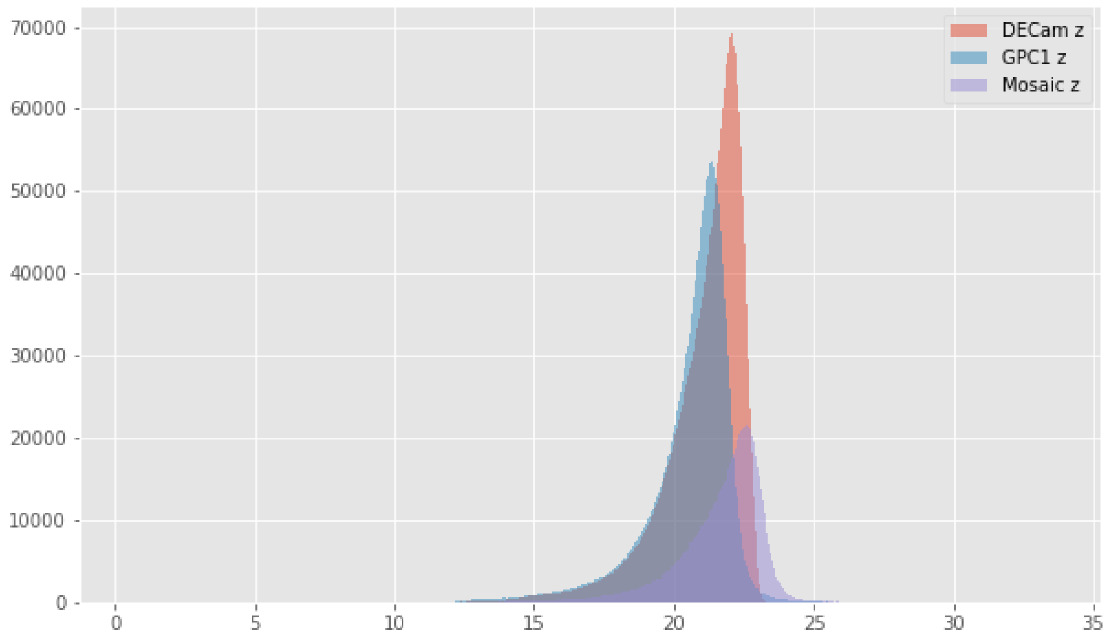
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/matplotlib/axes/_axes.py:5
  warnings.warn("No labelled objects found. "
```

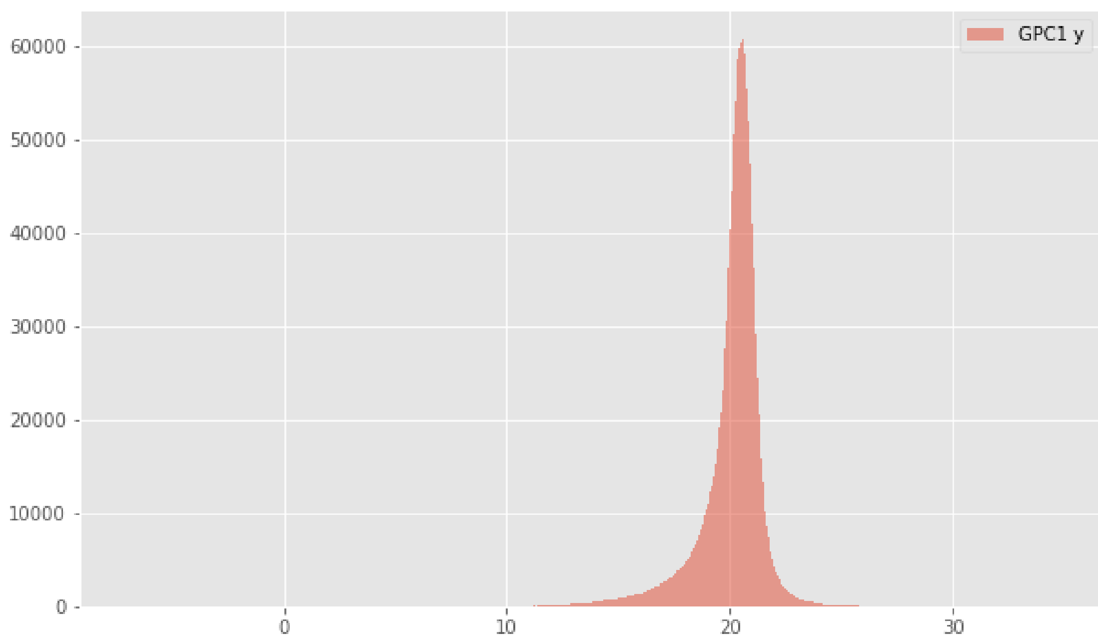

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0


GPC1 g
90prime g
60000
50000
40000
30000
20000
10000
0
0
5
10
15
20
25
30
35
40


GPC1 r
90prime r
50000
40000
30000
20000
10000
0
0
5
10
15
20
25
30
35


GPC1 i
50000
40000
30000
20000
10000
0
0
5
10
15
20
25
30
35

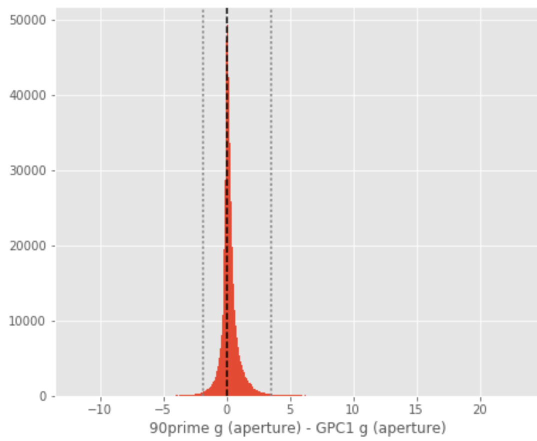


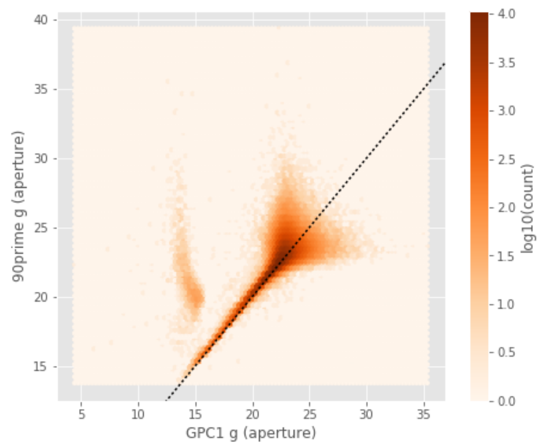


1.4.2 II.b - Comparing magnitudes

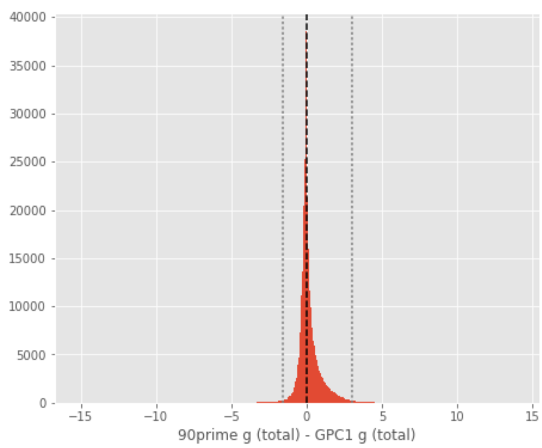
We compare one to one each magnitude in similar bands.

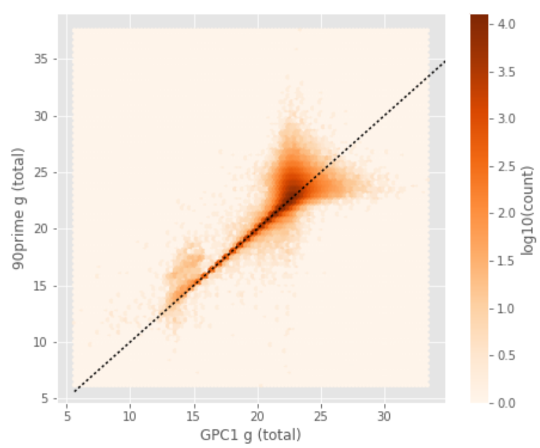
90prime g (aperture) - GPC1 g (aperture):
- Median: 0.16
- Median Absolute Deviation: 0.25
- 1% percentile: -1.8993983268737793
- 99% percentile: 3.54915212631226





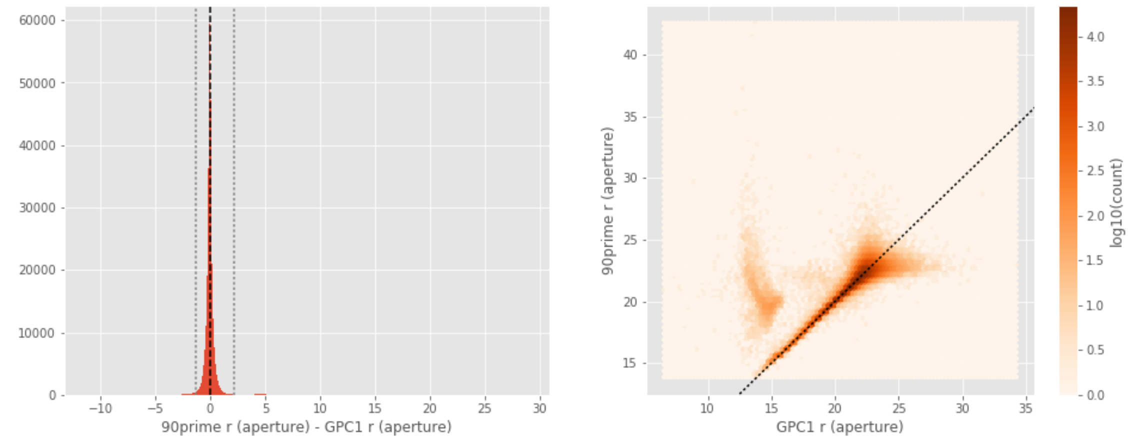
90prime g (total) - GPC1 g (total):
- Median: -0.00
- Median Absolute Deviation: 0.23
- 1% percentile: -1.5988301277160644
- 99% percentile: 2.9839715003967267



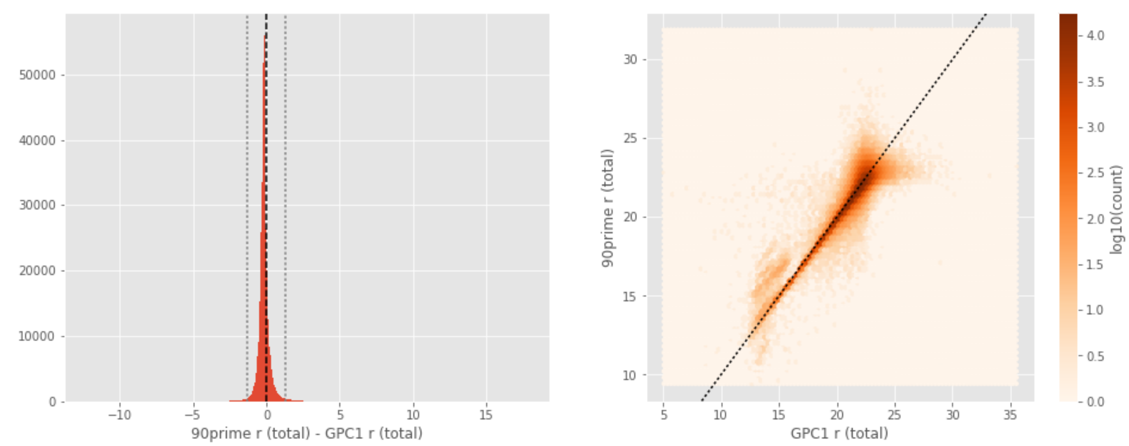


90prime r (aperture) - GPC1 r (aperture):
- Median: -0.05

- Median Absolute Deviation: 0.14
- 1% percentile: -1.3436206436157225
- 99% percentile: 2.164133319854683

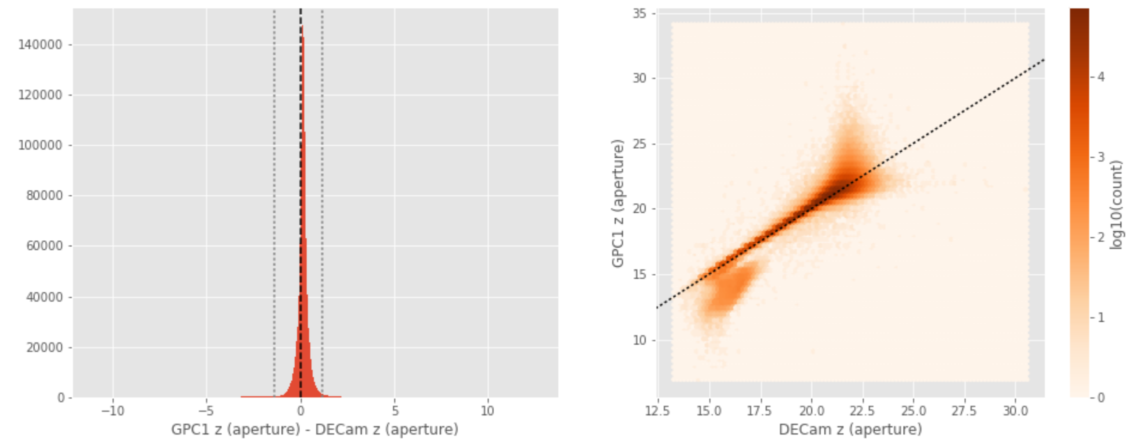


90prime r (total) - GPC1 r (total):
- Median: -0.18
- Median Absolute Deviation: 0.14
- 1% percentile: -1.291554069519043
- 99% percentile: 1.325196399688716

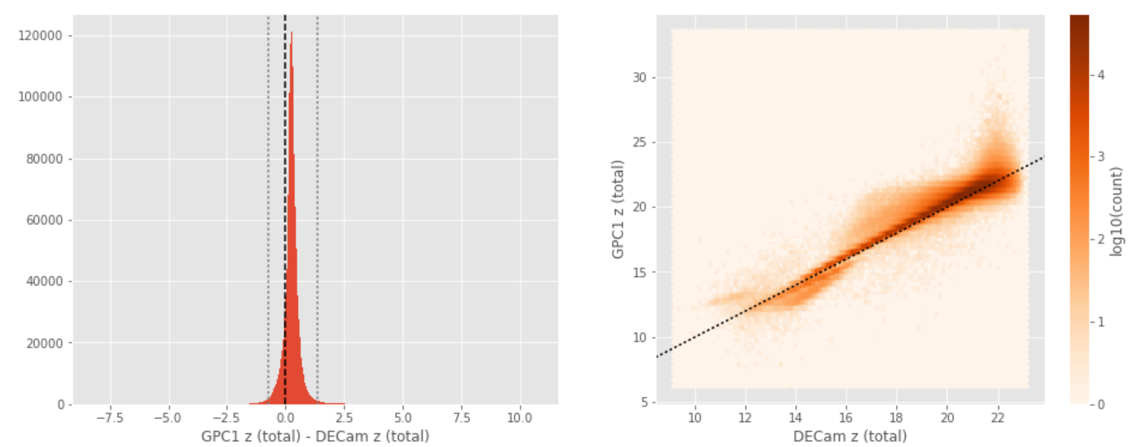


GPC1 z (aperture) - DECam z (aperture):
- Median: 0.15
- Median Absolute Deviation: 0.12
- 1% percentile: -1.3645053482055665

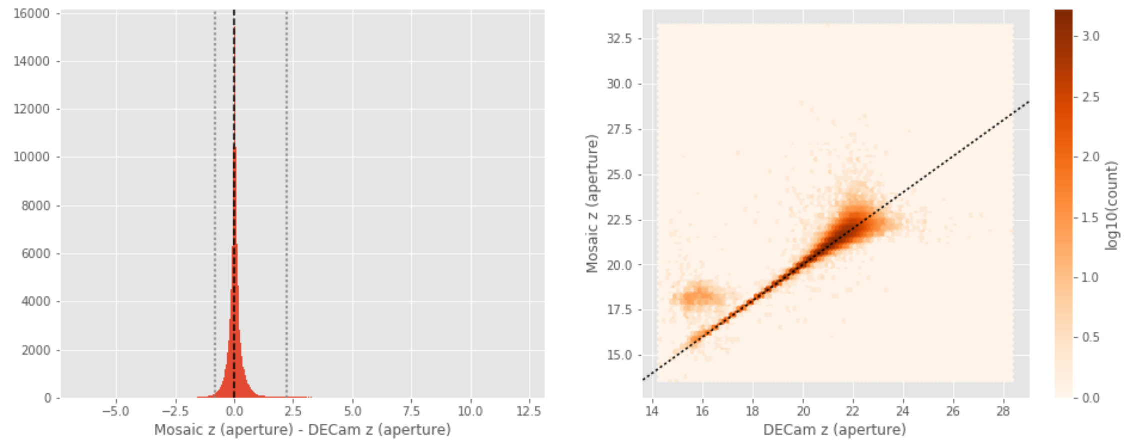
- 99% percentile: 1.171778297424316



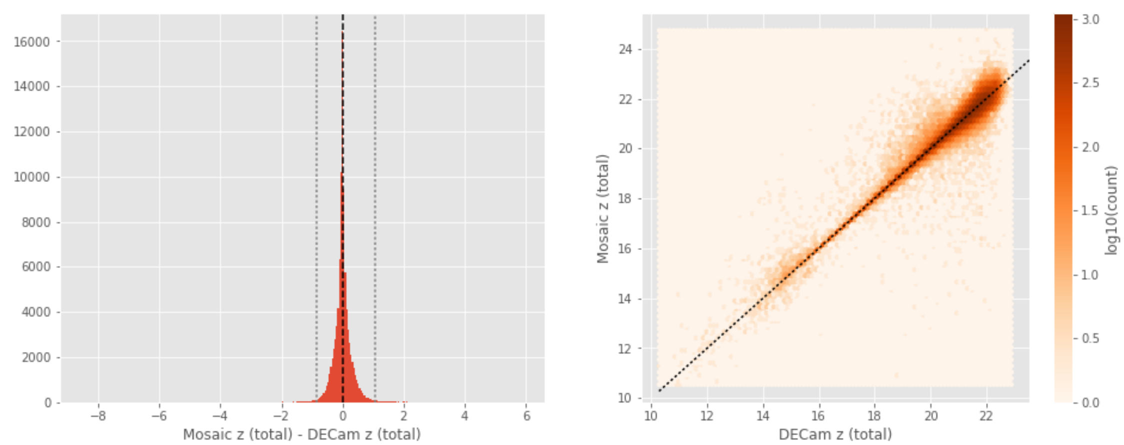
GPC1 z (total) - DECam z (total):
- Median: 0.26
- Median Absolute Deviation: 0.14
- 1% percentile: -0.7166573143005371
- 99% percentile: 1.356721229553223



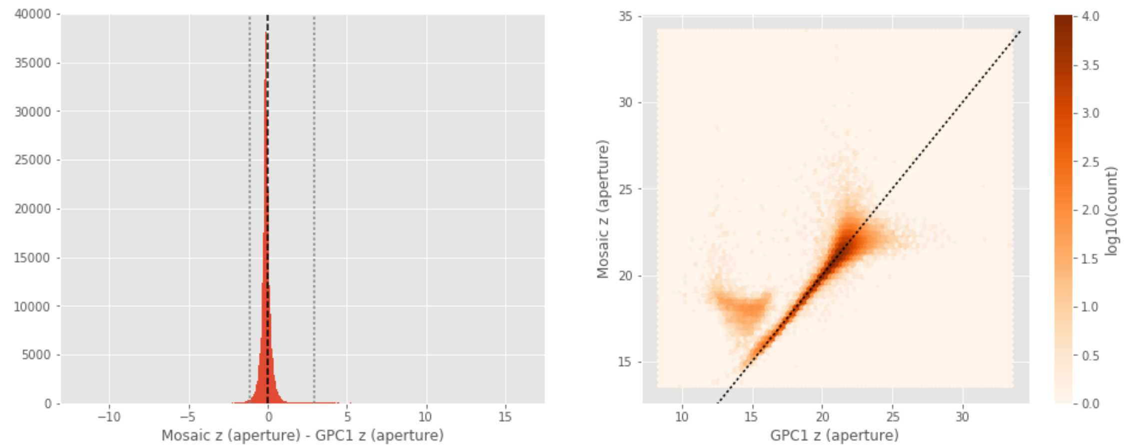
Mosaic z (aperture) - DECam z (aperture):
- Median: 0.01
- Median Absolute Deviation: 0.11
- 1% percentile: -0.8405391693115234
- 99% percentile: 2.199215316772468



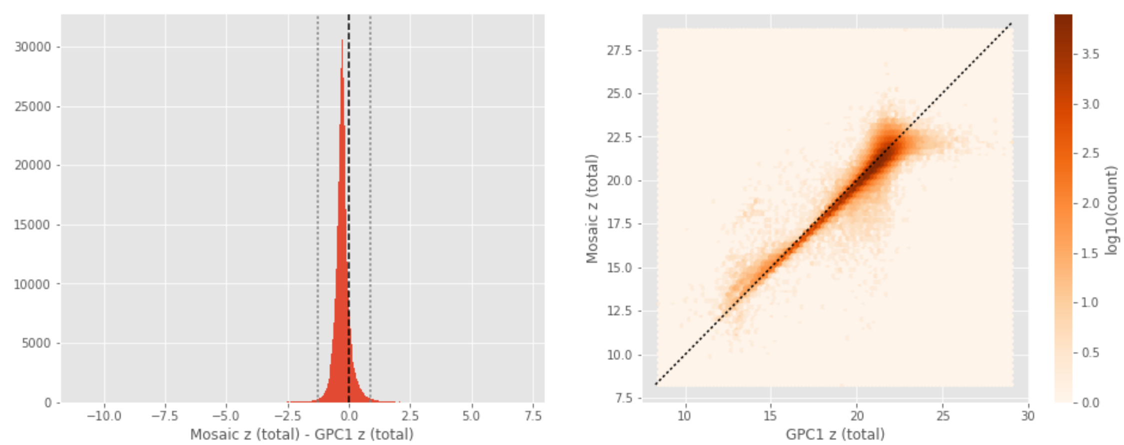
Mosaic z (total) - DECam z (total):
- Median: -0.01
- Median Absolute Deviation: 0.11
- 1% percentile: -0.8669711303710937
- 99% percentile: 1.0804067993164062



Mosaic z (aperture) - GPC1 z (aperture):
- Median: -0.11
- Median Absolute Deviation: 0.13
- 1% percentile: -1.1134843826293945
- 99% percentile: 2.9018537998199463




```
Mosaic z (total) - GPC1 z (total):
- Median: -0.28
- Median Absolute Deviation: 0.14
- 1% percentile: -1.2559052276611329
- 99% percentile: 0.8657968139648397
```



1.5 III - Comparing magnitudes to reference bands

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

1.5.1 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.

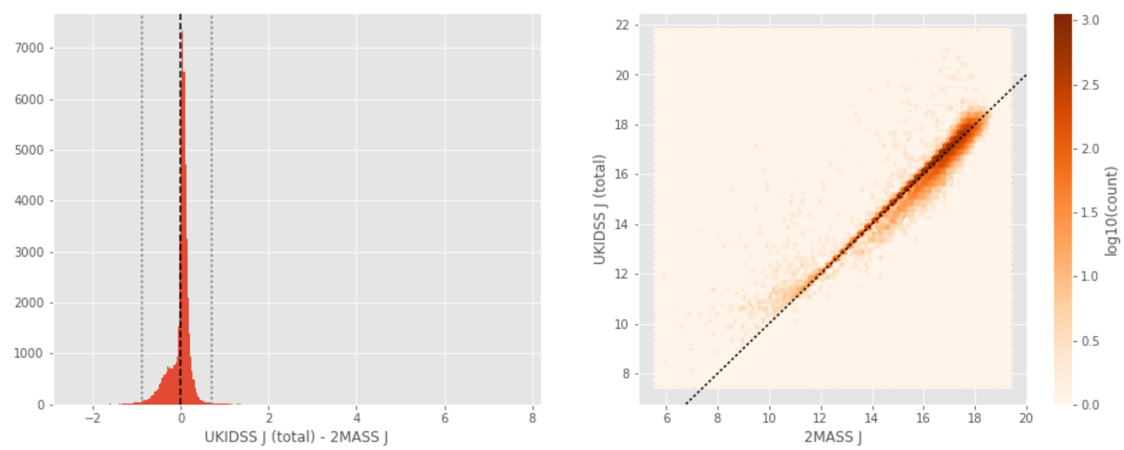
The 2MASS magnitudes are "*Vega-like*" and we have to convert them to AB magnitudes using the zero points provided on this page:

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

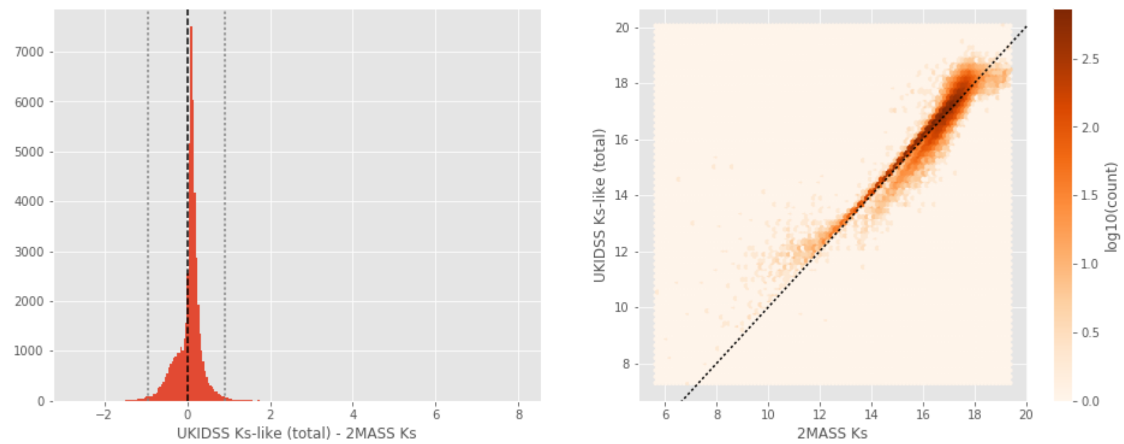
In addition, UKIDSS uses a K band whereas 2MASS uses a Ks ("short") band, this page give a correction to convert the K band in a Ks band with the formula:

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

```
UKIDSS J (total) - 2MASS J:
- Median: 0.05
- Median Absolute Deviation: 0.07
- 1% percentile: -0.8737602805234732
- 99% percentile: 0.7164782998750846
```

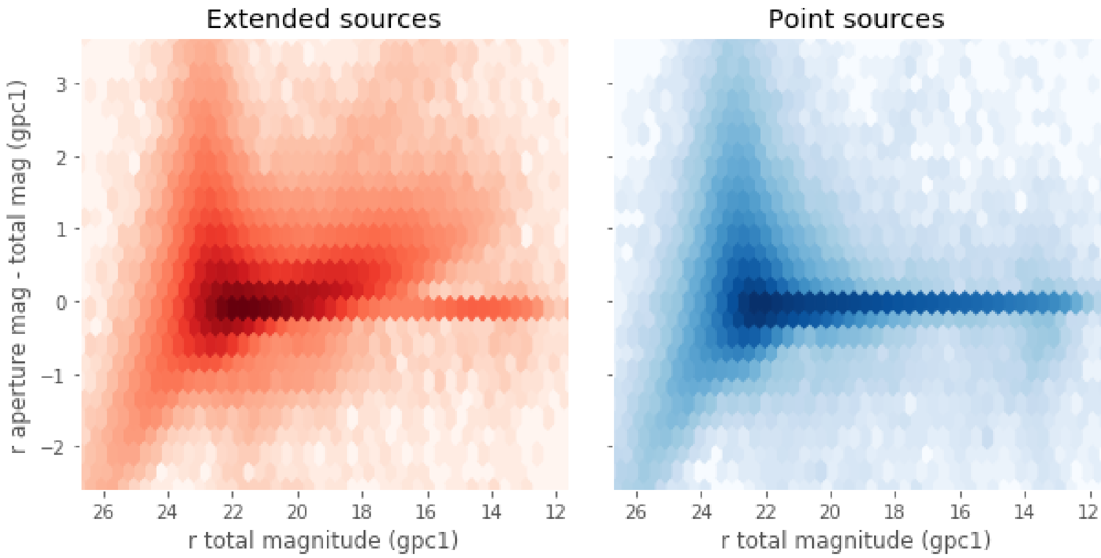


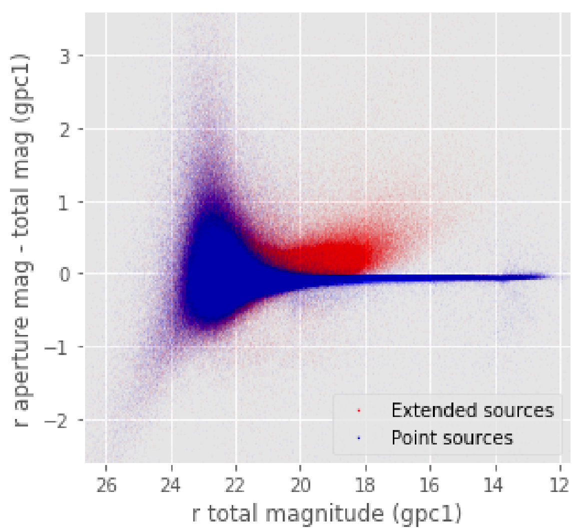

```
UKIDSS Ks-like (total) - 2MASS Ks:
- Median: 0.10
- Median Absolute Deviation: 0.11
- 1% percentile: -0.9586393060930751
- 99% percentile: 0.9195574765912466
```



1.6 IV - Comparing aperture magnitudes to total ones.

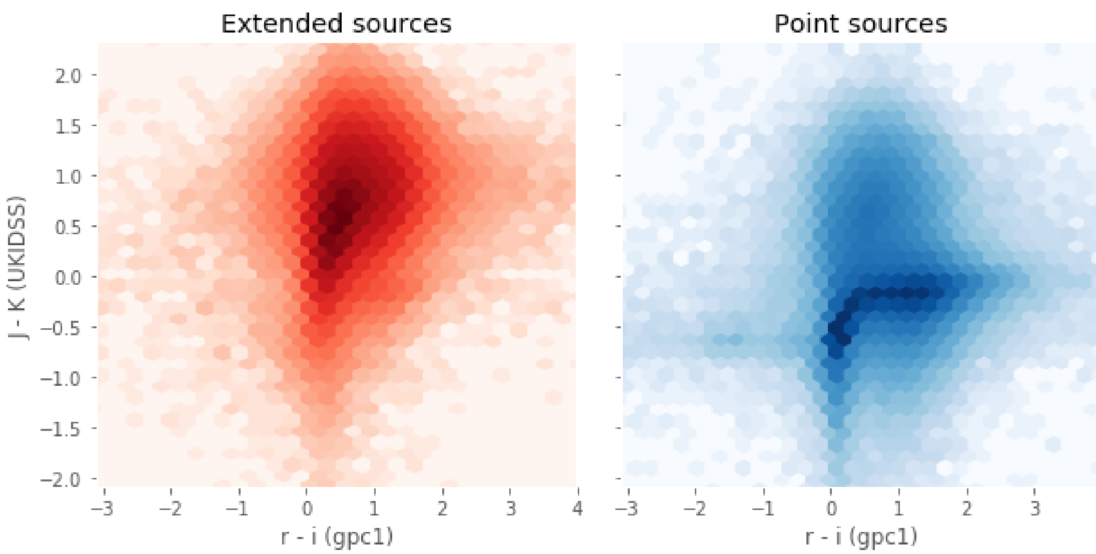
Number of source used: 2979508 / 6759591 (44.08%)

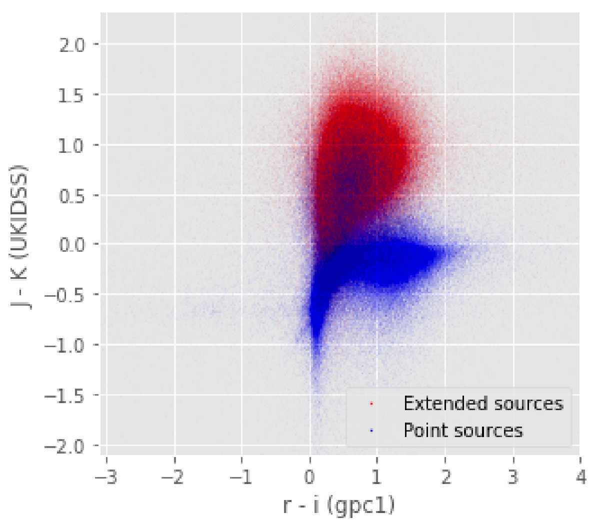




1.7 V - Color-color and magnitude-color plots

```
Number of source used: 1076909 / 6759591 (15.93%)
```


Extended sources
Point sources
J - K (UKIDSS)
r - i (gpc1)
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−3
−2
−1
0
1
2
3
4


J - K (UKIDSS)
r - i (gpc1)
Extended sources
Point sources
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−3
−2
−1
0
1
2
3
4

4_Selection_function

March 8, 2018

1 NGP 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 22:46:23.018267


Depth maps produced using: master_catalogue_ngp_20180219.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

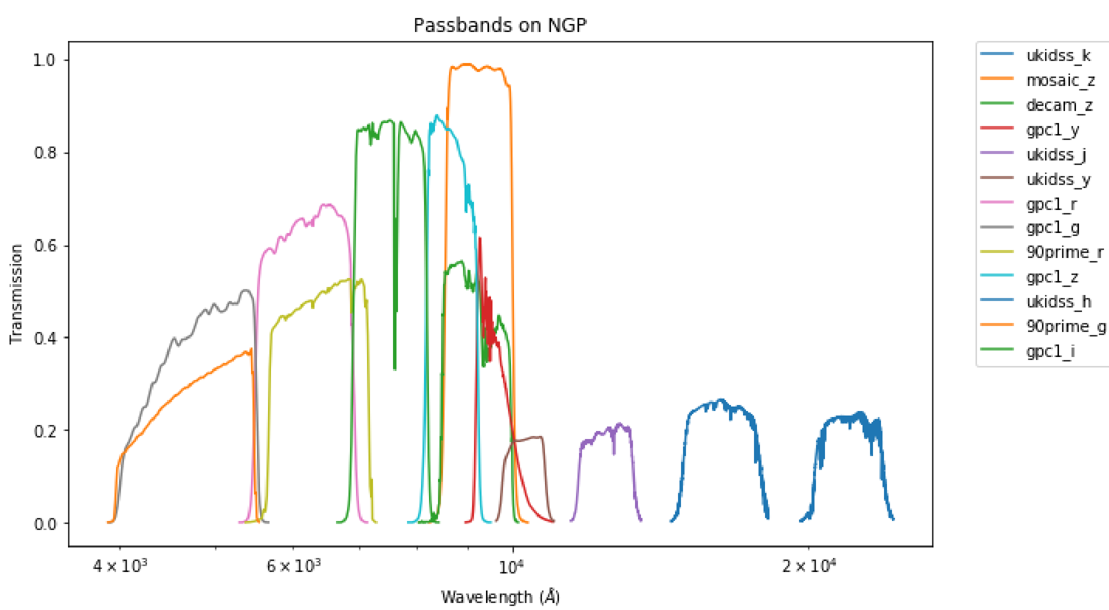
1.5 IV - Overview plots

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]: {'90prime_g',
          '90prime_r',
          'decam_z',
          'gpc1_g',
          'gpc1_i',
          'gpc1_r',
          'gpc1_y',
          'gpc1_z',
          'mosaic_z',
          'ukidss_h',
          'ukidss_j',
          'ukidss_k',
          'ukidss_y'}
```

```
Out[15]: <matplotlib.text.Text at 0x7f946af6c128>
```



1.5.2 IV.a - Depth overview

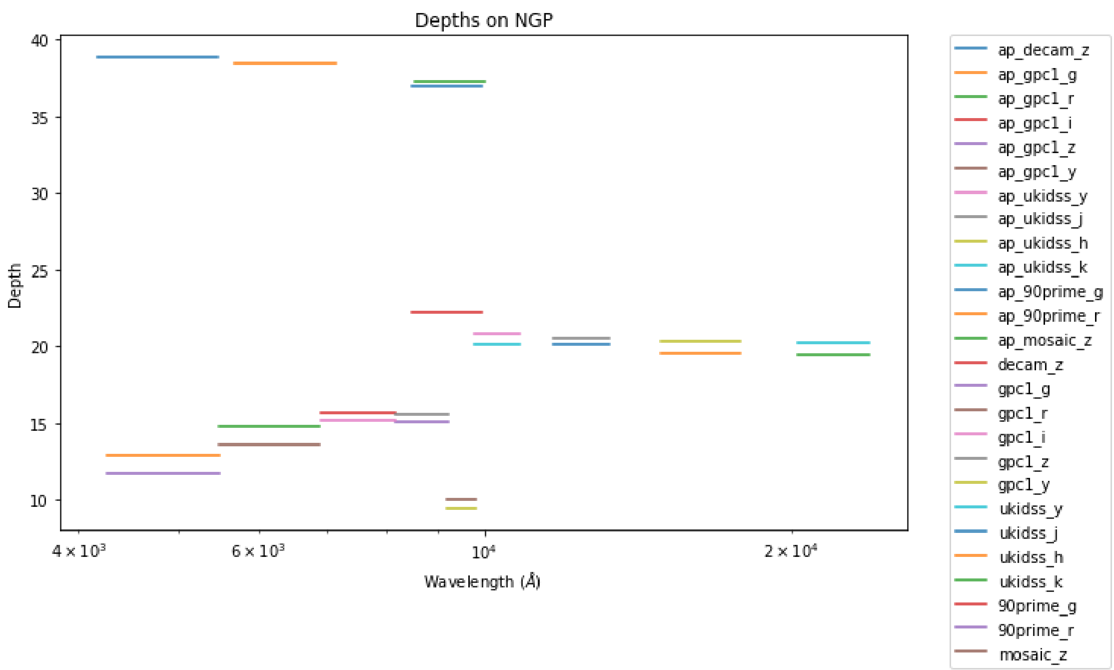
Then we plot the mean depths available across the area a given band is available

```
decam_z: mean flux error: 1.1463661166999373e-06, 3sigma in AB mag (Aperture): 37.55888851050684
gpc1_g: mean flux error: 4807.692928162375, 3sigma in AB mag (Aperture): 13.502355060485058
gpc1_r: mean flux error: 856.2731977410074, 3sigma in AB mag (Aperture): 15.375665987308935
gpc1_i: mean flux error: 400.42818884877096, 3sigma in AB mag (Aperture): 16.200885256176683
gpc1_z: mean flux error: 664.2480392517627, 3sigma in AB mag (Aperture): 15.65137116040058
gpc1_y: mean flux error: 72191.37734621752, 3sigma in AB mag (Aperture): 10.560983543604614
ukidss_y: mean flux error: 3.391916036605835, 3sigma in AB mag (Aperture): 21.381084130189983
ukidss_j: mean flux error: 4.302145481109619, 3sigma in AB mag (Aperture): 21.12298413225701
ukidss_h: mean flux error: 5.362170696258545, 3sigma in AB mag (Aperture): 20.88384527583154
ukidss_k: mean flux error: 5.858597755432129, 3sigma in AB mag (Aperture): 20.787712661019206
90prime_g: mean flux error: 2.112811046117713e-07, 3sigma in AB mag (Aperture): 39.3950452162844
90prime_r: mean flux error: 2.976190103254339e-07, 3sigma in AB mag (Aperture): 39.0230451927253
mosaic_z: mean flux error: 9.432757224203669e-07, 3sigma in AB mag (Aperture): 37.77060022090386
decam_z: mean flux error: 0.9615241289138794, 3sigma in AB mag (Total): 22.749796395475805
gpc1_g: mean flux error: 15224.37419036471, 3sigma in AB mag (Total): 12.250848239044842
gpc1_r: mean flux error: 2651.880278334602, 3sigma in AB mag (Total): 14.148312079359478
gpc1_i: mean flux error: 585.2270004453892, 3sigma in AB mag (Total): 15.788885976121257
gpc1_z: mean flux error: 419.4584464526905, 3sigma in AB mag (Total): 16.150474503150512
gpc1_y: mean flux error: 116445.53948719491, 3sigma in AB mag (Total): 10.041889719972126
ukidss_y: mean flux error: 6.215388774871826, 3sigma in AB mag (Total): 20.723526115430523
ukidss_j: mean flux error: 6.273008346557617, 3sigma in AB mag (Total): 20.71350719972913
ukidss_h: mean flux error: 10.831151962280273, 3sigma in AB mag (Total): 20.120510240507308
ukidss_k: mean flux error: 11.837139129638672, 3sigma in AB mag (Total): 20.024079982728075
90prime_g: mean flux error: inf, 3sigma in AB mag (Total): -inf
90prime_r: mean flux error: inf, 3sigma in AB mag (Total): -inf
mosaic_z: mean flux error: inf, 3sigma in AB mag (Total): -inf


ap_decam_z (8490.0, 9960.0, 1470.0)
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_ukidss_y (9790.0, 10820.0, 1030.0)
ap_ukidss_j (11695.0, 13280.0, 1585.0)
ap_ukidss_h (14925.0, 17840.0, 2915.0)
ap_ukidss_k (20290.0, 23820.0, 3530.0)
ap_90prime_g (4180.0, 5470.0, 1290.0)
ap_90prime_r (5680.0, 7150.0, 1470.0)
ap_mosaic_z (8552.0, 10018.0, 1466.0)
decam_z (8490.0, 9960.0, 1470.0)
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)
ukidss_y (9790.0, 10820.0, 1030.0)
```

```
ukidss_j (11695.0, 13280.0, 1585.0)
ukidss_h (14925.0, 17840.0, 2915.0)
ukidss_k (20290.0, 23820.0, 3530.0)
90prime_g (4180.0, 5470.0, 1290.0)
90prime_r (5680.0, 7150.0, 1470.0)
mosaic_z (8552.0, 10018.0, 1466.0)
```

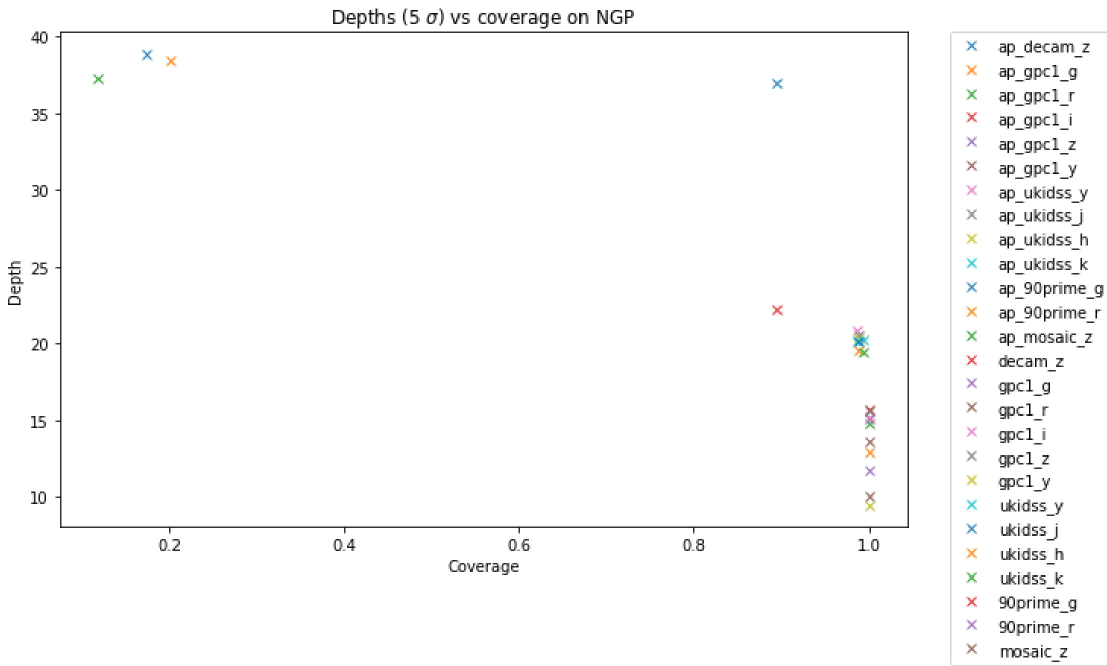
Out[20]: <matplotlib.text.Text at 0x7f9469390240>



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 0x7f946916e0b8>


Depths (5 σ) vs coverage on NGP
Depth
Coverage
40
35
30
25
20
15
10
0.2
0.4
0.6
0.8
1.0
ap_decam_z
ap_gpc1_g
ap_gpc1_r
ap_gpc1_i
ap_gpc1_z
ap_gpc1_y
ap_ukidss_y
ap_ukidss_j
ap_ukidss_h
ap_ukidss_k
ap_90prime_g
ap_90prime_r
ap_mosaic_z
decam_z
gpc1_g
gpc1_r
gpc1_i
gpc1_z
gpc1_y
ukidss_y
ukidss_j
ukidss_h
ukidss_k
90prime_g
90prime_r
mosaic_z