1.1_SSDF

March 8, 2018

1 SSDF master catalogue

1.1 Preparation of SSDF data

This catalogue comes from `dmu0_SSDF`.

The SSDF data consists in two catalogue of IRAC Ch1 and Ch2 fluxes: one for Ch1 detected sources and the other for Ch2 detected sources. For now, we are only using the Ch1 detected sources. **TODO** : We may find a way to merge the two catalogues and select the best flux for each source.

```
This notebook was run with herschelhelp_internal version:
04829ed (Thu Nov 2 16:57:19 2017 +0000)
```

1.2 I - Column selection

```
WARNING: UnitsWarning: 'vega' did not parse as fits unit: At col 0, Unit 'vega' not supported by
```

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

1.3 II - Removal of duplicated sources

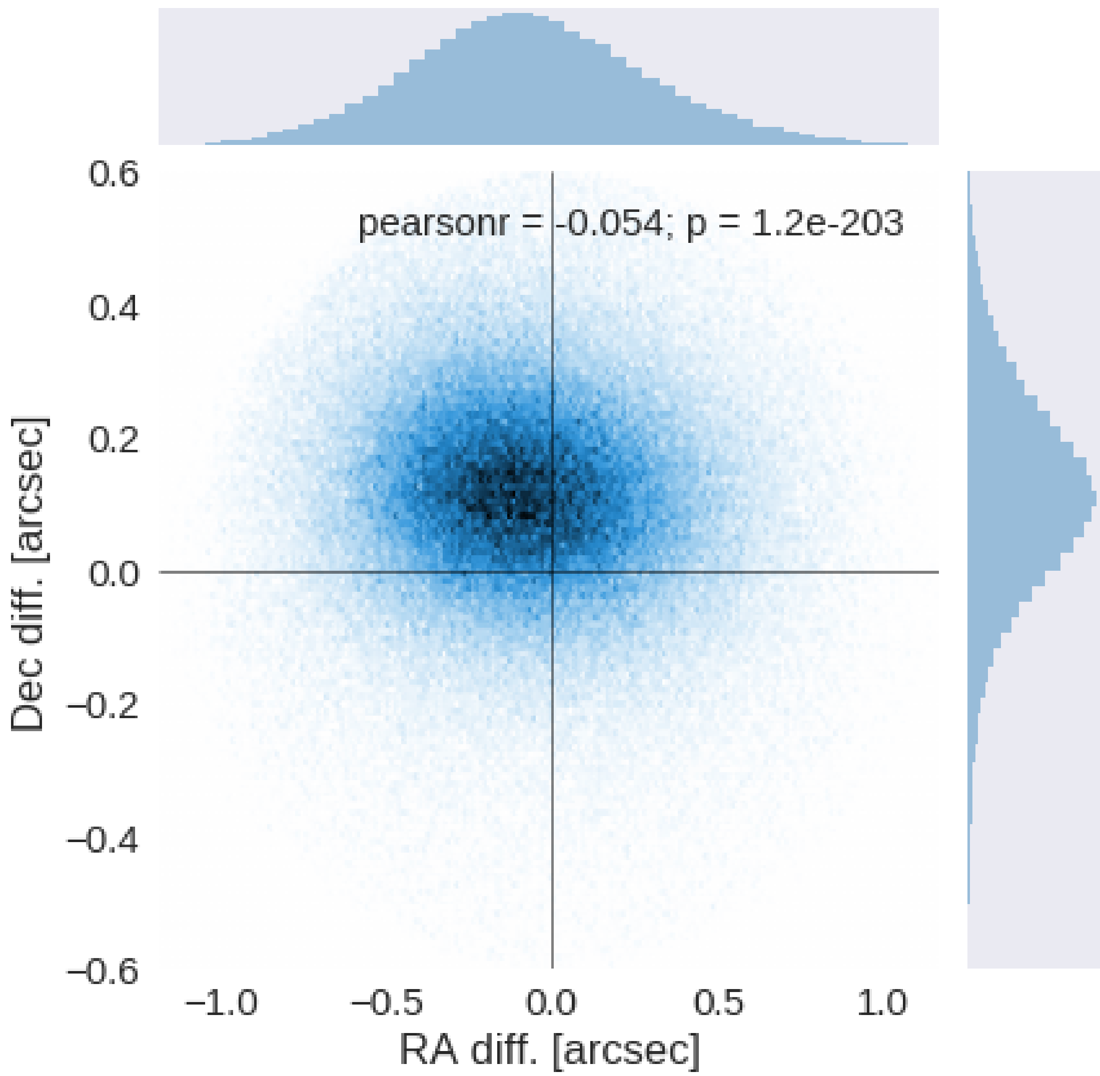
We remove duplicated objects from the input catalogues.

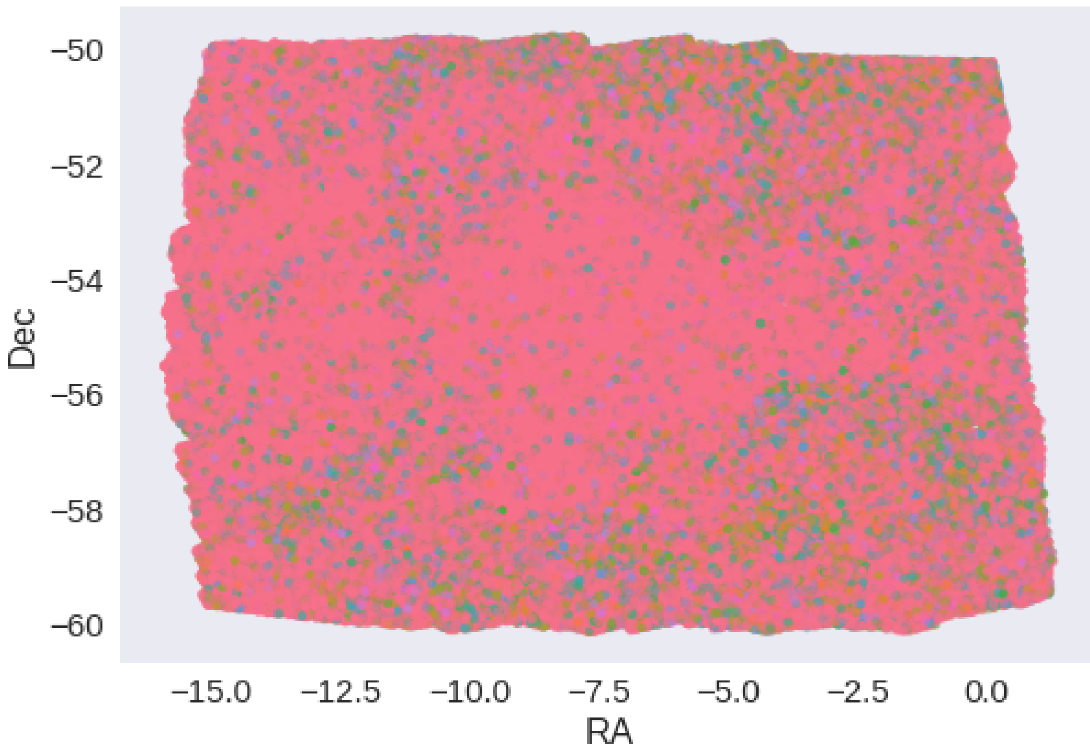
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10
Check the NumPy 1.11 release notes for more information.
  ma.MaskedArray.__setitem__(self, index, value)


The initial catalogue had 5488166 sources.
The cleaned catalogue has 5488141 sources (25 removed).
The cleaned catalogue has 25 sources flagged as having been cleaned
```

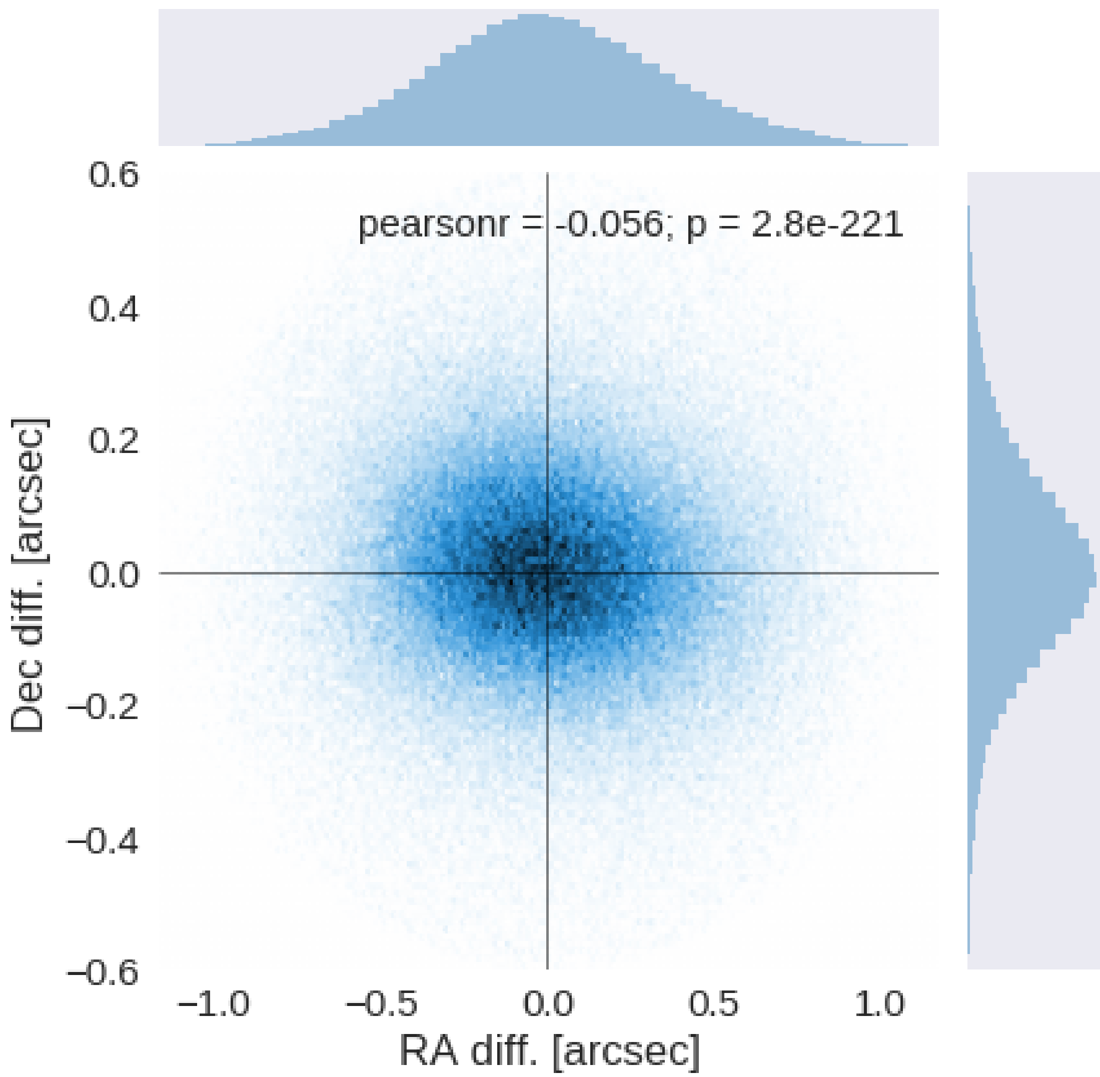
1.4 III - Astrometry correction

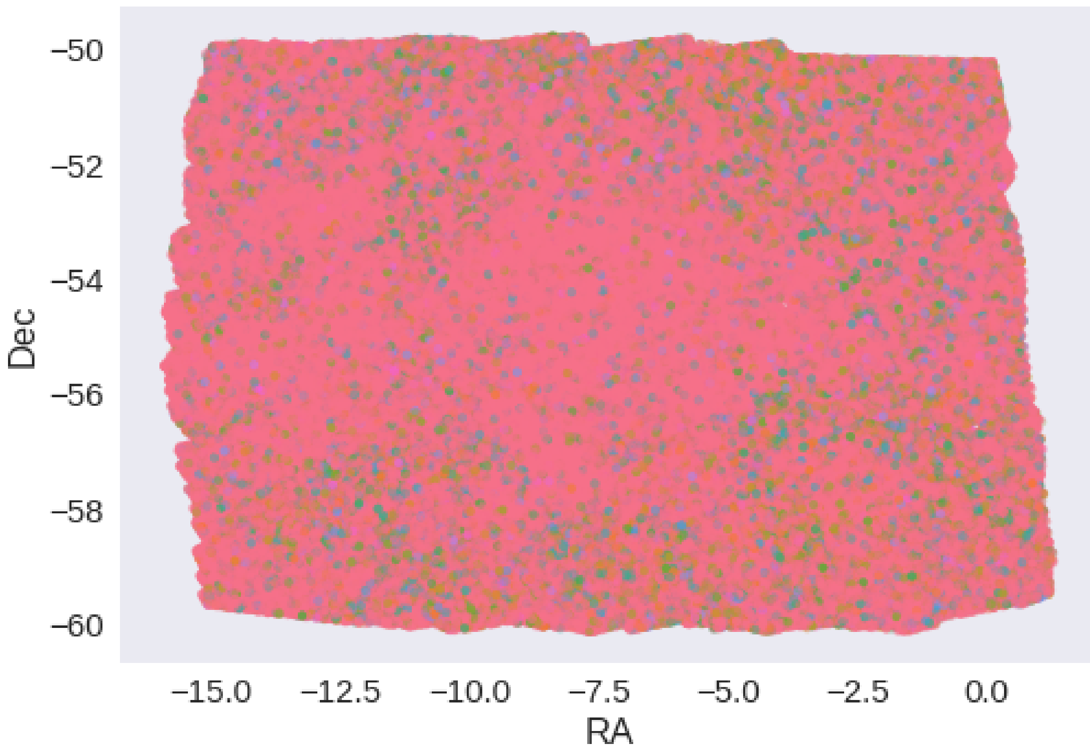
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.






```
RA correction: 0.08155033374350751 arcsec
Dec correction: -0.11236725684966586 arcsec
```


pearsonr = -0.056; p = 2.8e-221
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
Dec diff. [arcsec]
−1.0
−0.5
0.0
0.5
1.0
RA diff. [arcsec]



1.5 IV - Flagging Gaia objects

```
352272 sources flagged.
```

1.6 V - Flagging objects near bright stars

2 VI - Saving to disk

1.2_VISTA-VHS

March 8, 2018

1 SSDF master catalogue

1.1 Preparation of VHS data

VISTA telescope/VHS catalogue: the catalogue comes from `dmu0_VHS`.

In the catalogue, we keep:

- The identifier (it's unique in the catalogue);
- The position;
- The stellarity;
- The magnitude for each band.
- 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.

- Note: on SSDF, the VHS catalogue does not contain Y data.*

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

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

1.3 II - Removal of duplicated sources

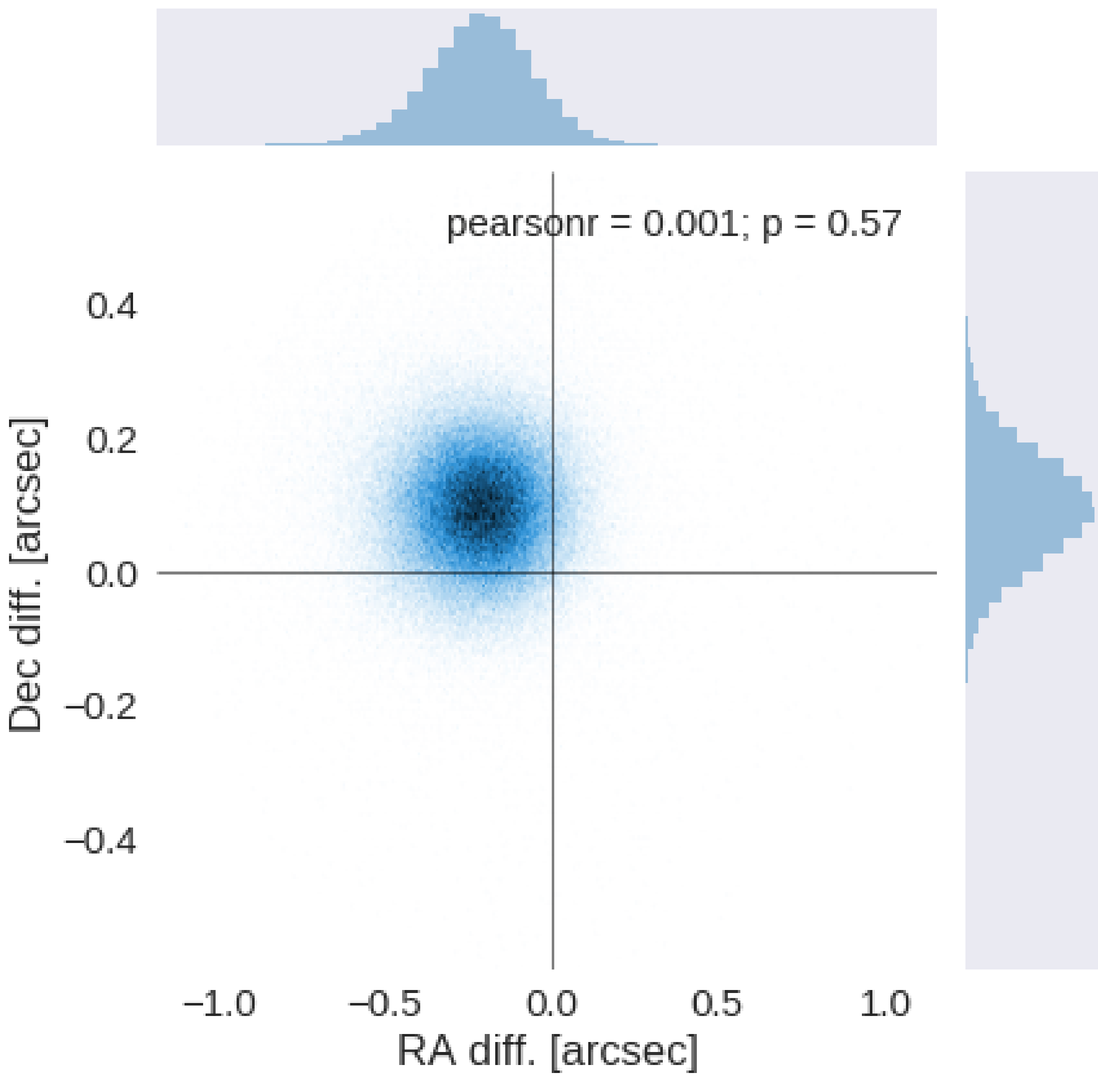
We remove duplicated objects from the input catalogues.

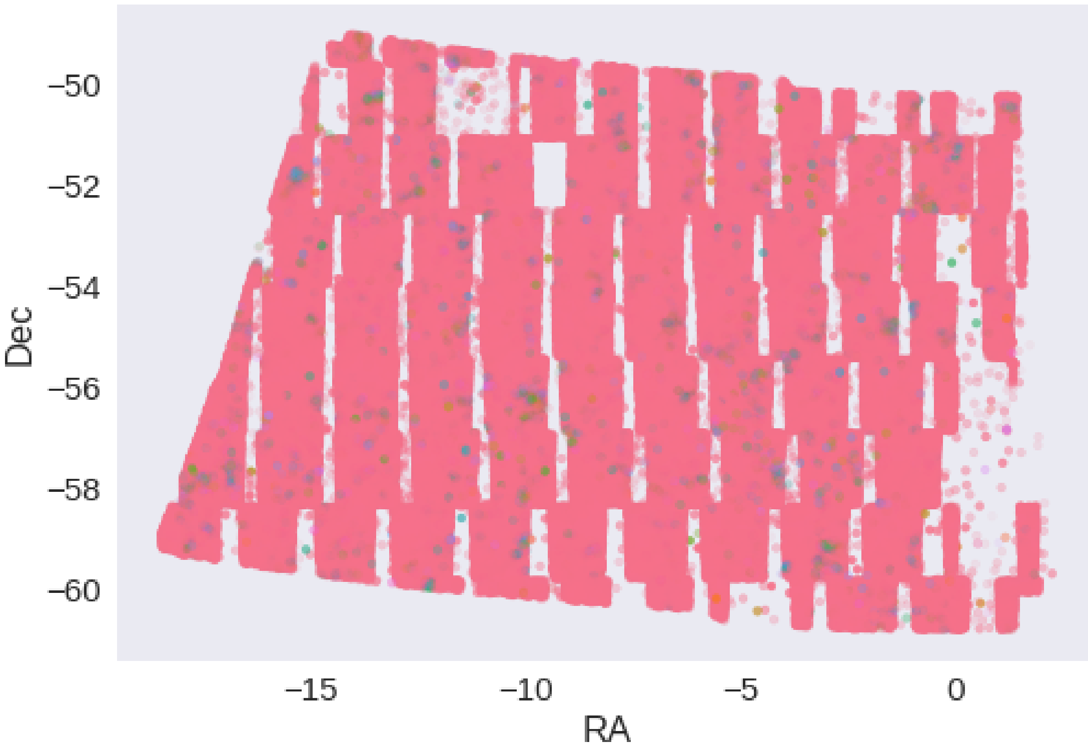
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10
Check the NumPy 1.11 release notes for more information.
  ma.MaskedArray.__setitem__(self, index, value)
```

```
The initial catalogue had 2406318 sources.
The cleaned catalogue has 2406063 sources (255 removed).
The cleaned catalogue has 253 sources flagged as having been cleaned
```

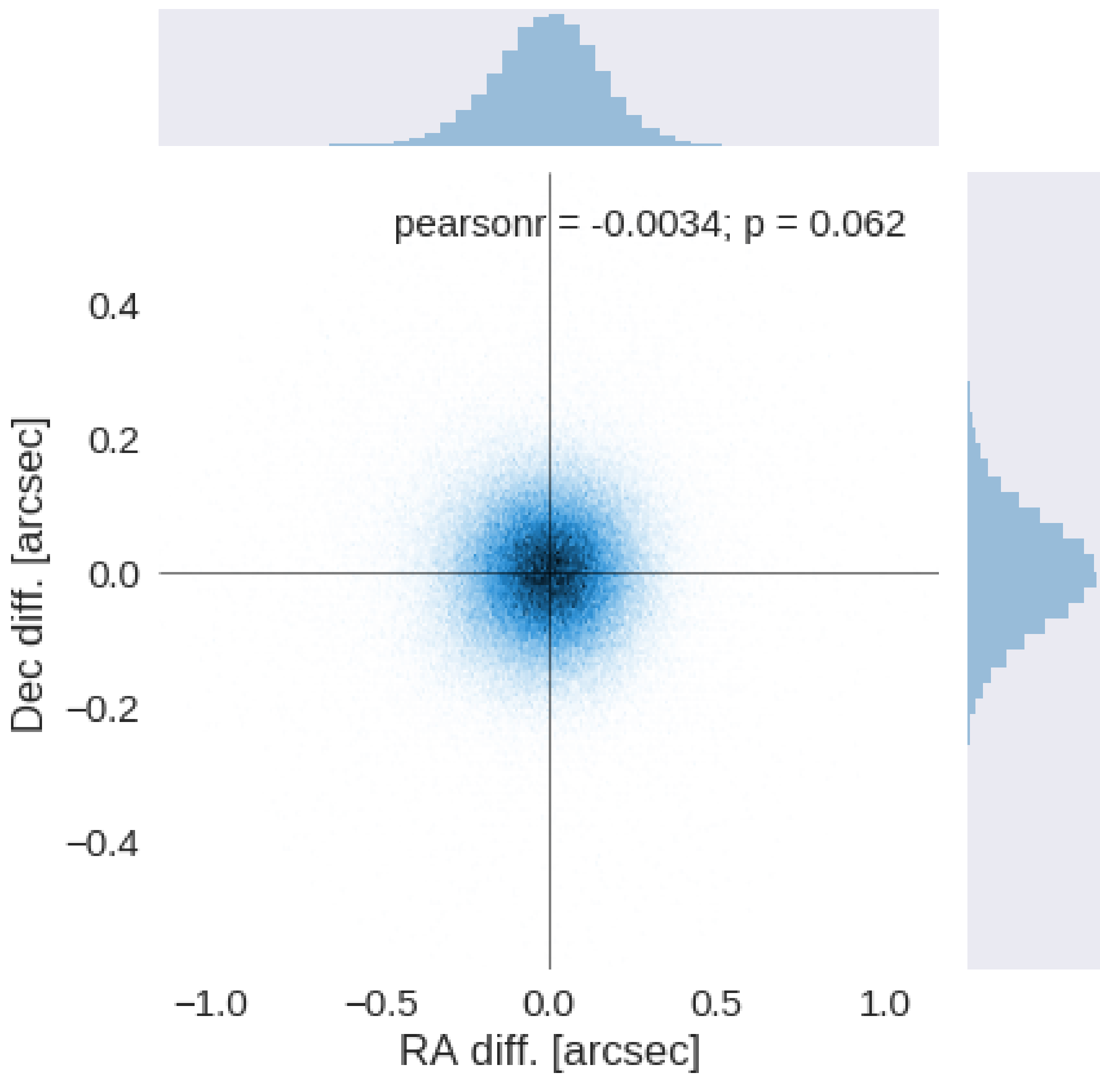
1.4 III - Astrometry correction

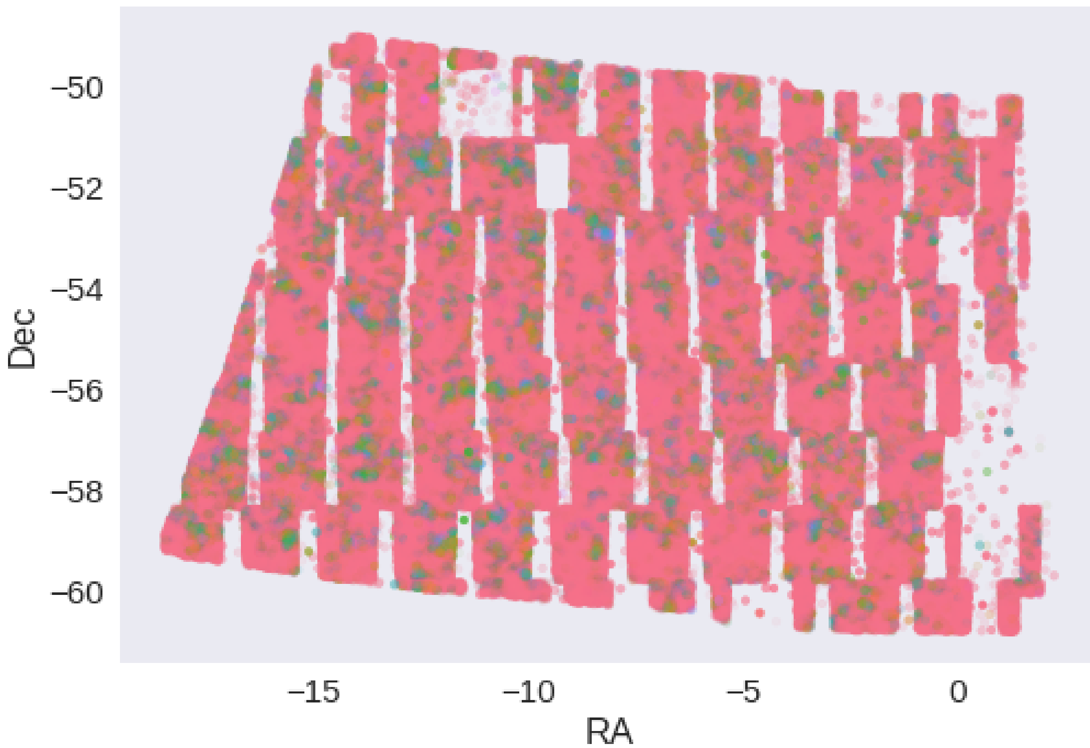
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.






```
RA correction: 0.21025412717108338 arcsec
Dec correction: -0.09461352436233028 arcsec
```


pearsonr = -0.0034; p = 0.062
0.4
0.2
0.0
−0.2
−0.4
Dec diff. [arcsec]
−1.0
−0.5
0.0
0.5
1.0
RA diff. [arcsec]



1.5 IV - Flagging Gaia objects

```
301682 sources flagged.
```

1.6 V - Flagging objects near bright stars

2 VI - Saving to disk

1.3_DES

March 8, 2018

1 SSDF master catalogue

1.1 Preparation of DES data

Blanco DES catalogue: the catalogue comes from `dmu0_DES`.

In the catalogue, we keep:

- The identifier (it's unique in the catalogue);
- The position;
- The G band stellarity;
- The magnitude for each band.
- The auto/kron magnitudes/fluxes to be used as total magnitude.
- The aperture magnitudes, which are used to compute a corrected 2 arcsec aperture magnitude.

We don't know when the maps have been observed. We will take the final observation date as 2017.

```
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:
2018-02-21 16:18:33.821790
```

1.2 1 - Aperture correction

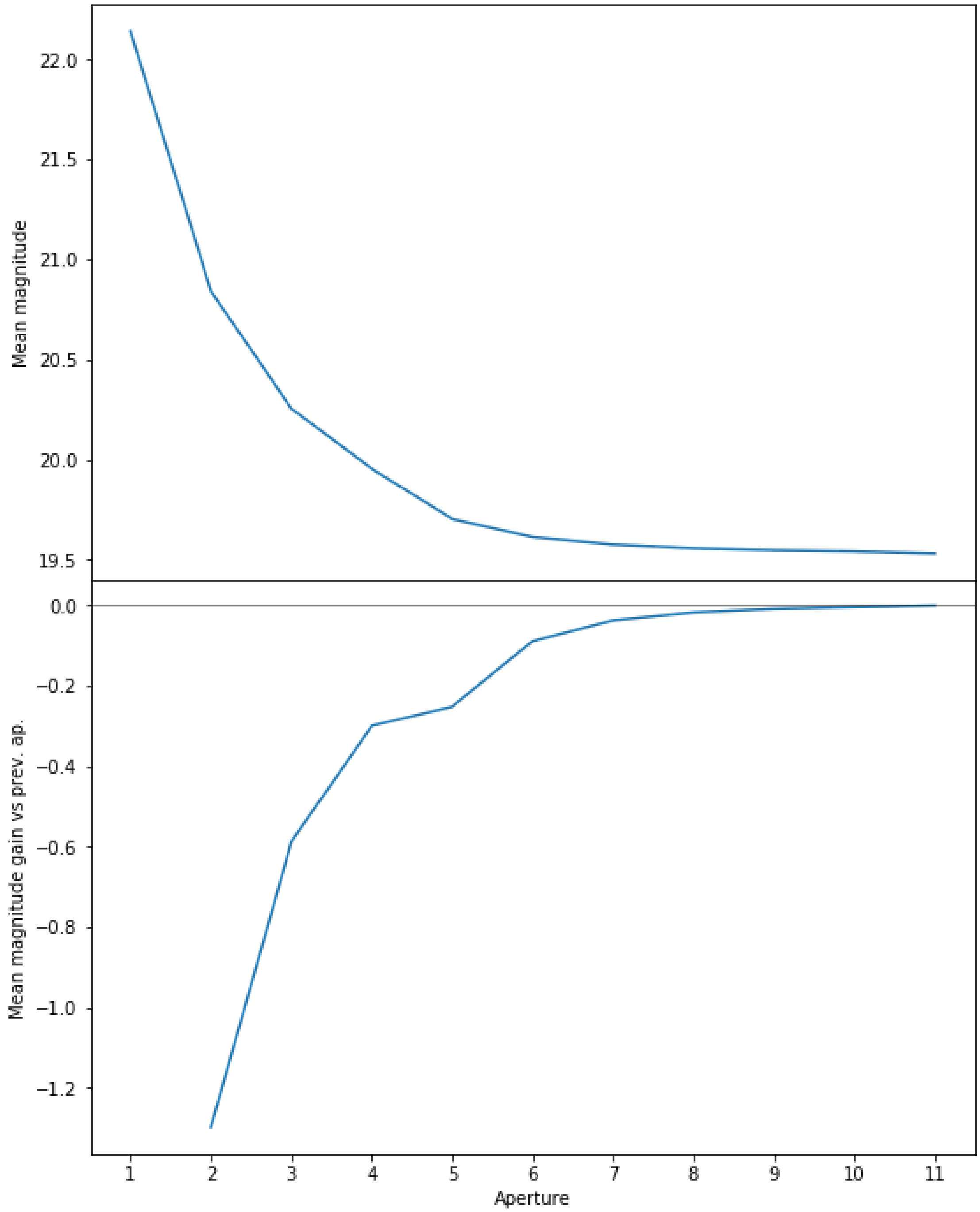
To compute aperture correction we need to dertermine two parametres: the target aperture and the range of magnitudes for the stars that will be used to compute the correction.

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

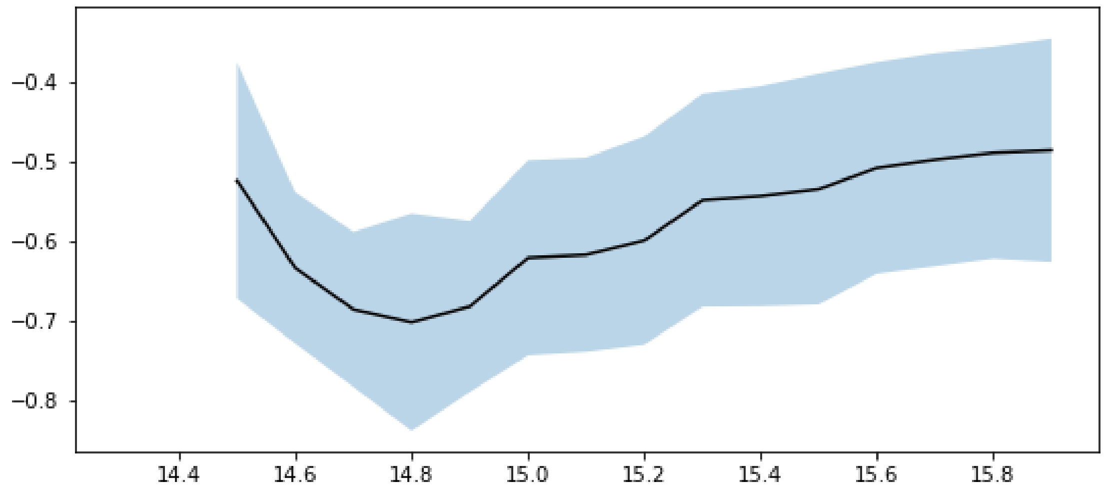
The evolution of the magnitudes of the objects by plotting on the same plot aperture number vs the mean magnitude. The mean gain (loss when negative) of magnitude is each aperture compared to the previous (except for the first of course). As target aperture, we should use the smallest (i.e. less noisy) 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.

1.2.1 I.a - g band



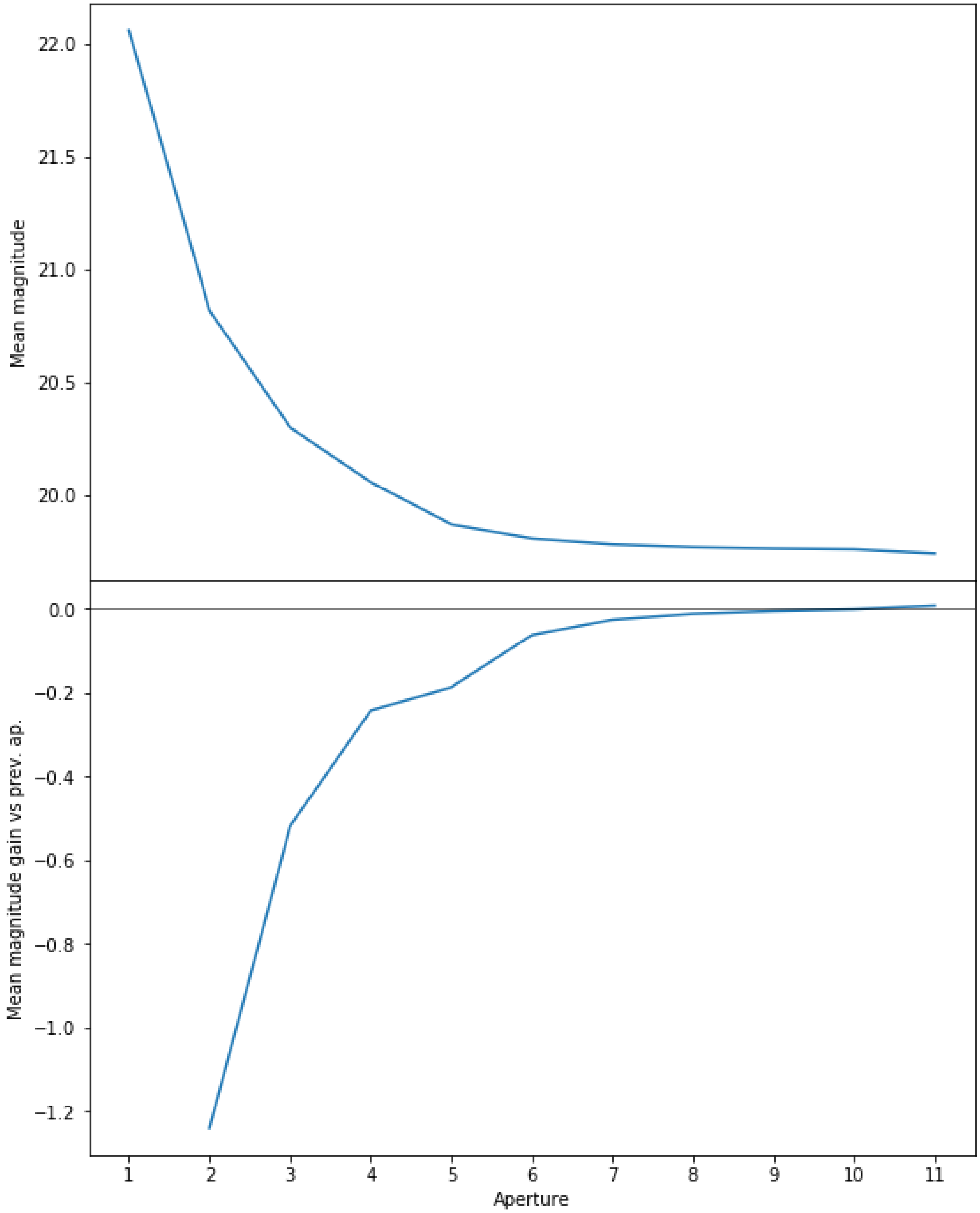
We will use aperture 10 as target.



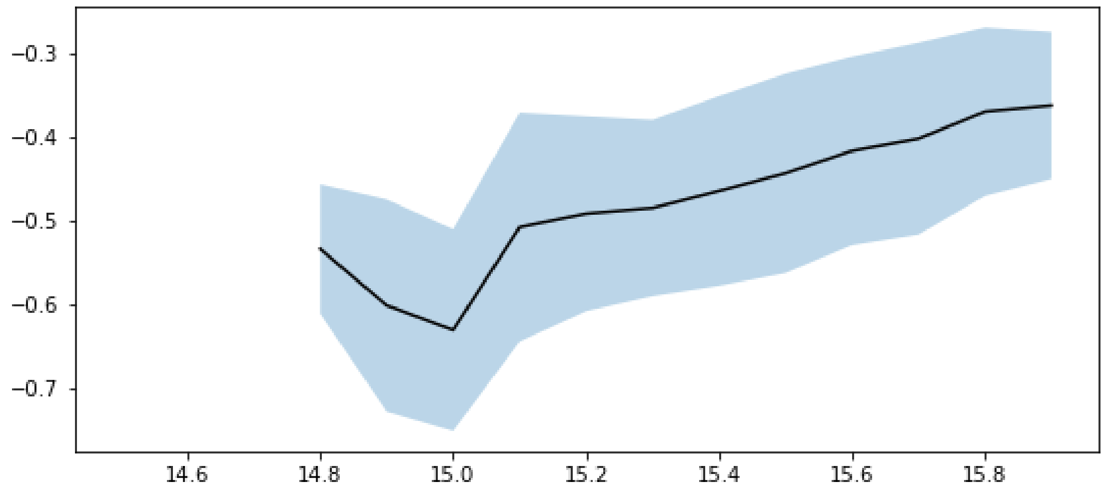
We will use magnitudes between 15.0 and 16.0

```
Aperture correction for g band:
Correction: -0.5084390640258789
Number of source used: 11527
RMS: 0.13857766793996795
```

1.2.2 I.b - r band



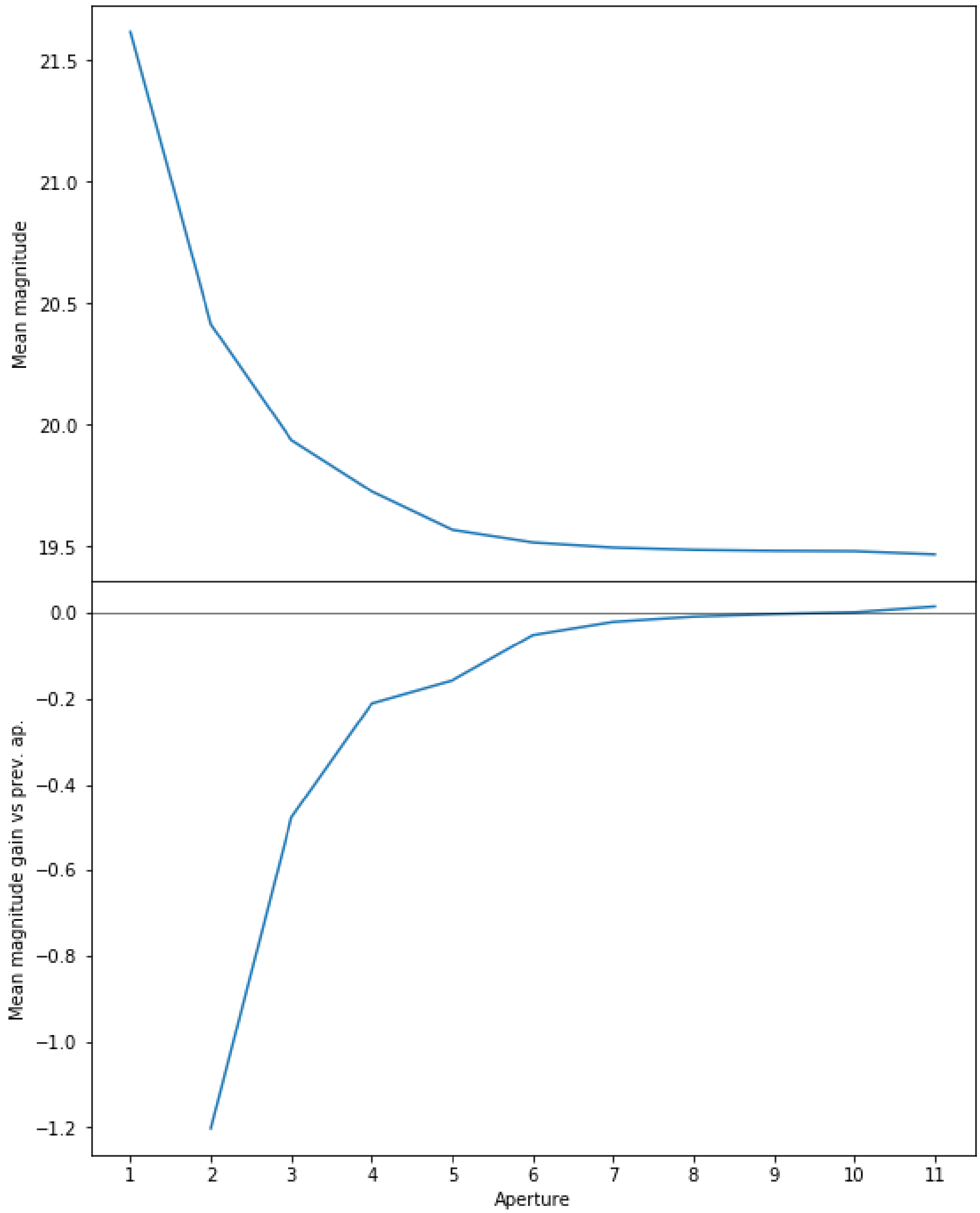
We will use aperture 10 as target.



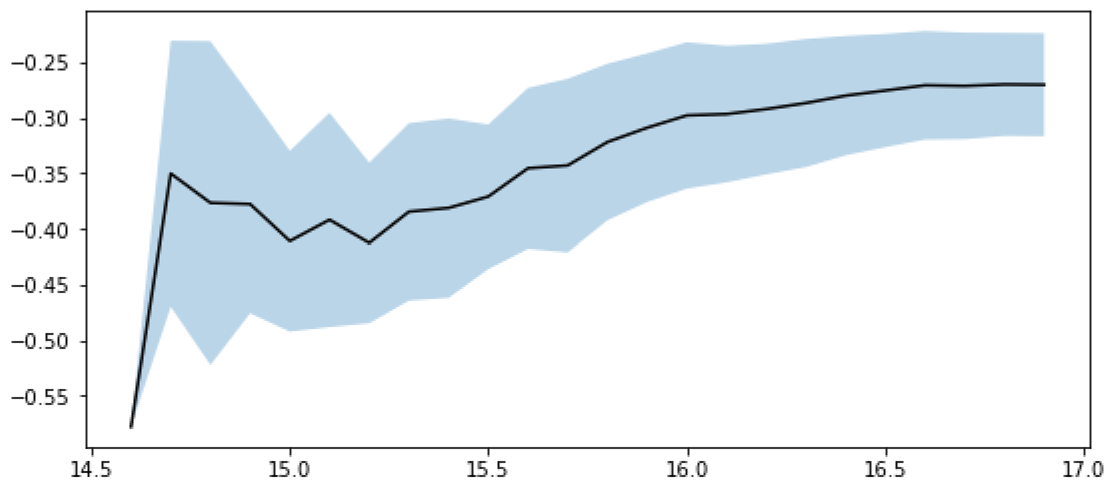
We use magnitudes between 15.0 and 16.0.

```
Aperture correction for r band:
Correction: -0.3870668411254883
Number of source used: 6262
RMS: 0.10930435440532538
```

1.2.3 I.b - i band



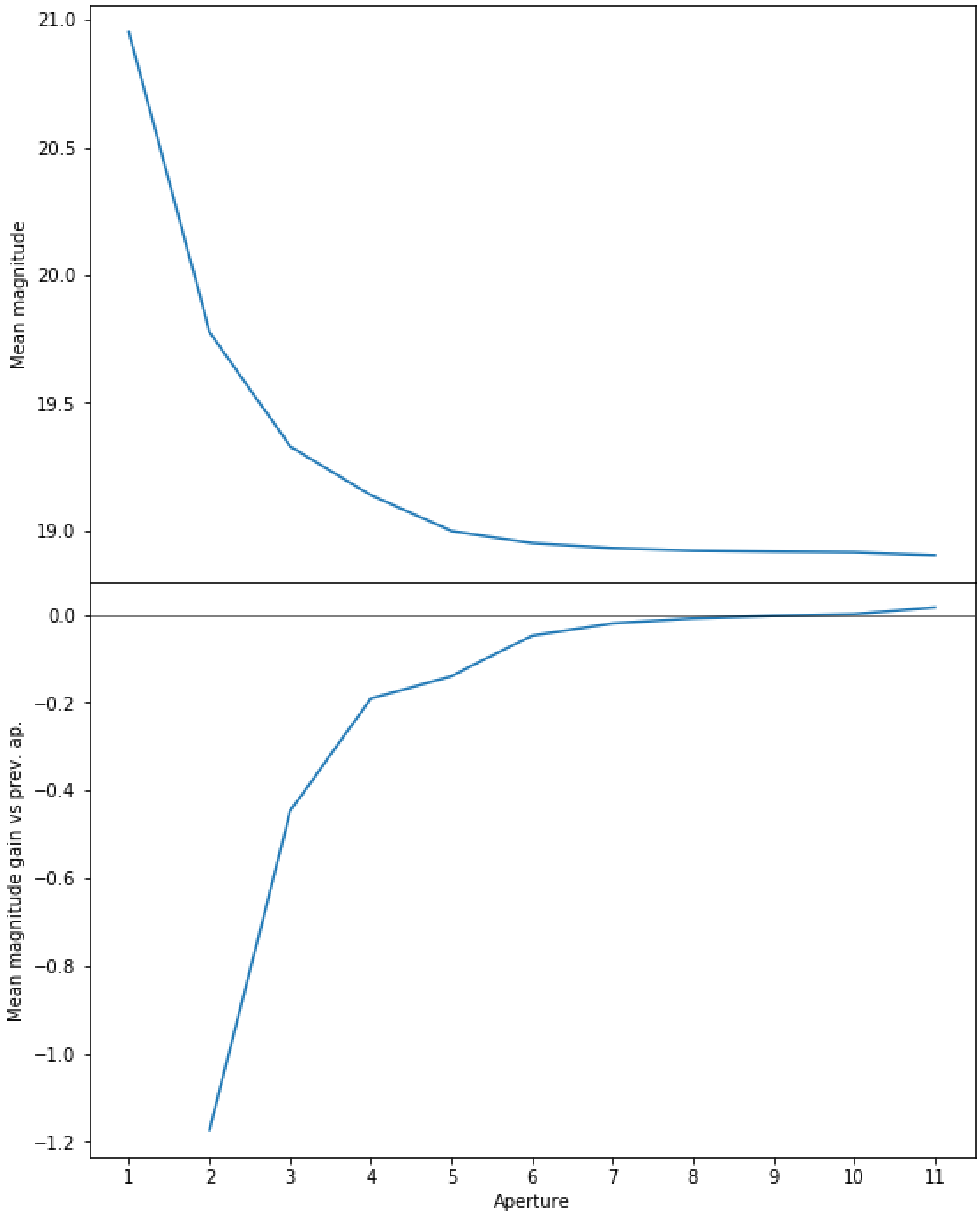
We will use aperture 10 as target.



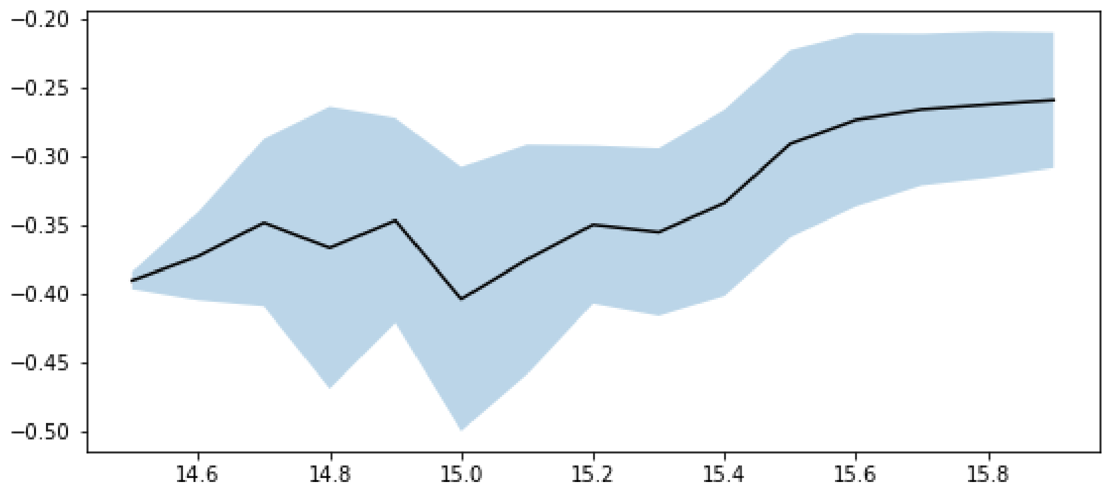
We use magnitudes between 15.0 and 16.0.

```
Aperture correction for i band:
Correction: -0.3250293731689453
Number of source used: 4829
RMS: 0.07326213149972925
```

1.2.4 I.b - z band



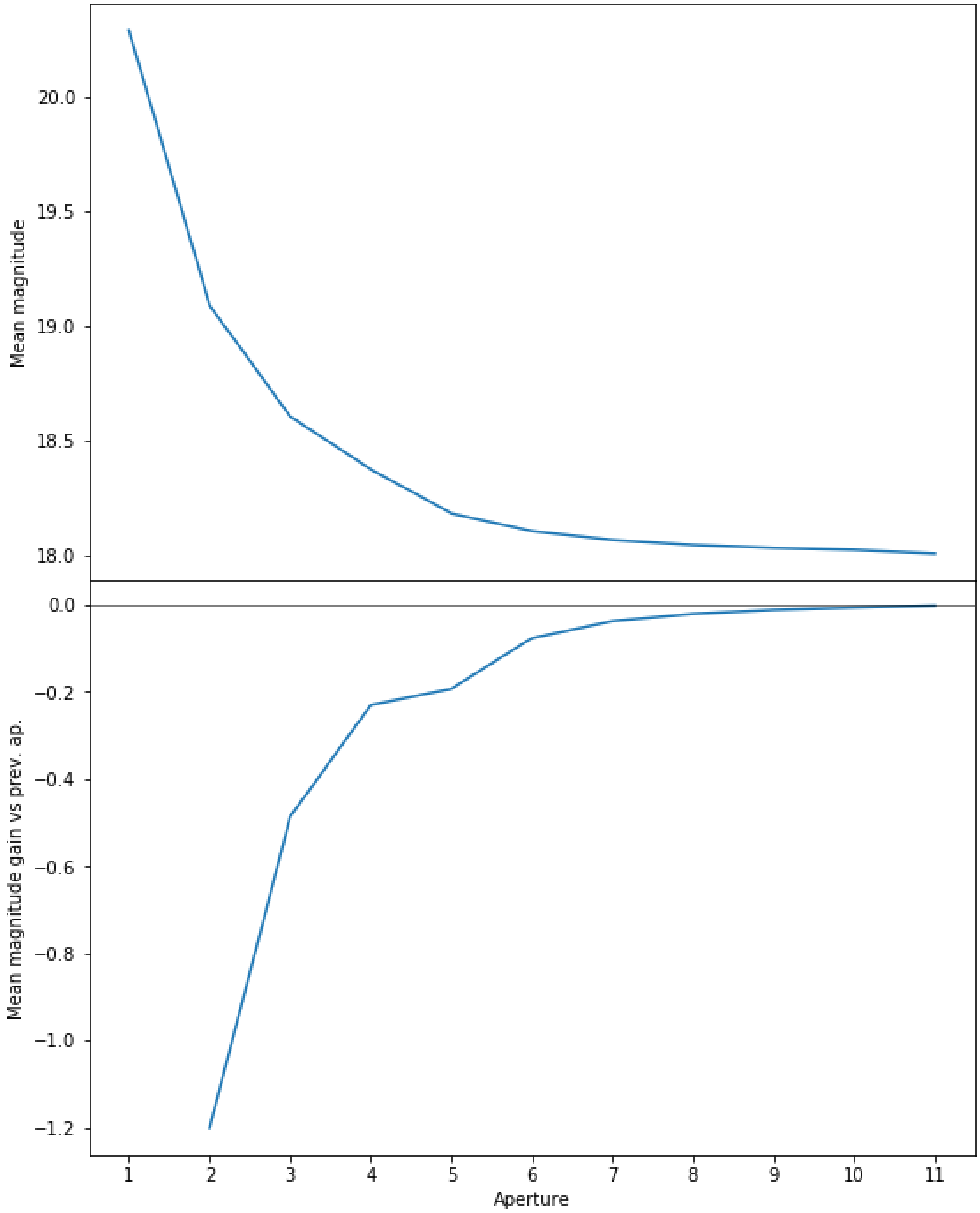
We will use aperture 57 as target.



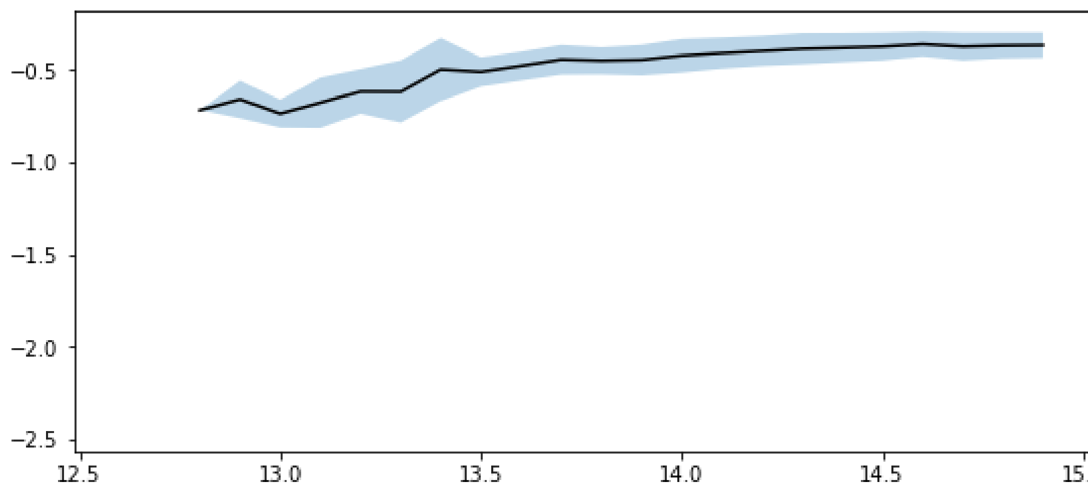
We use magnitudes between 15.0 and 16.0.

```
Aperture correction for z band:
Correction: -0.26555299758911133
Number of source used: 9874
RMS: 0.055136808300606646
```

1.2.5 I.b - y band



We will use aperture 10 as target.



We use magnitudes between 15.0 and 16.0.

```
Aperture correction for y band:
Correction: -0.35900163650512695
Number of source used: 7472
RMS: 0.06888716175884298
```

1.3 2 - 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[24]: <IPython.core.display.HTML object>
```

1.4 II - Removal of duplicated sources

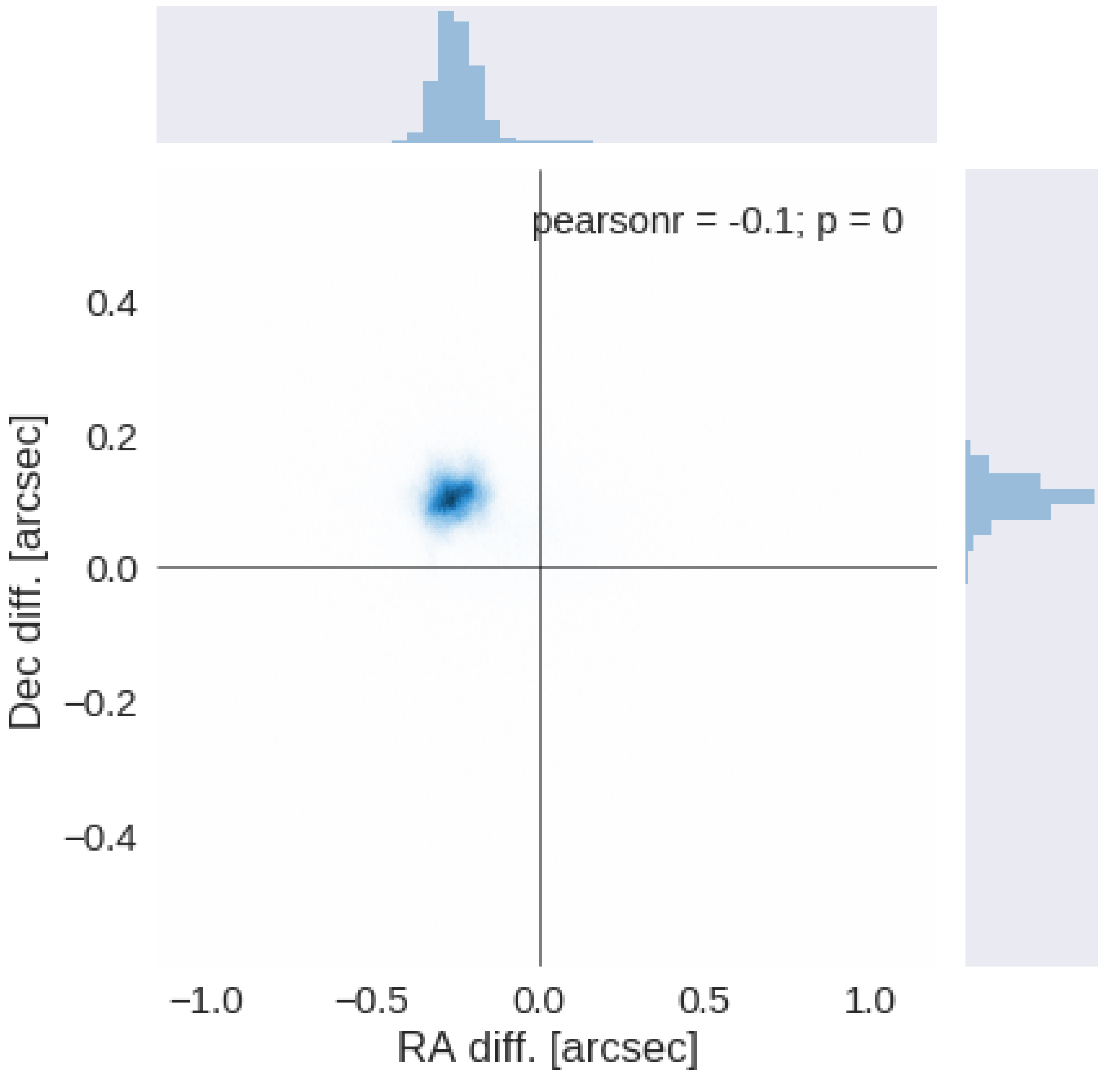
We remove duplicated objects from the input catalogues.

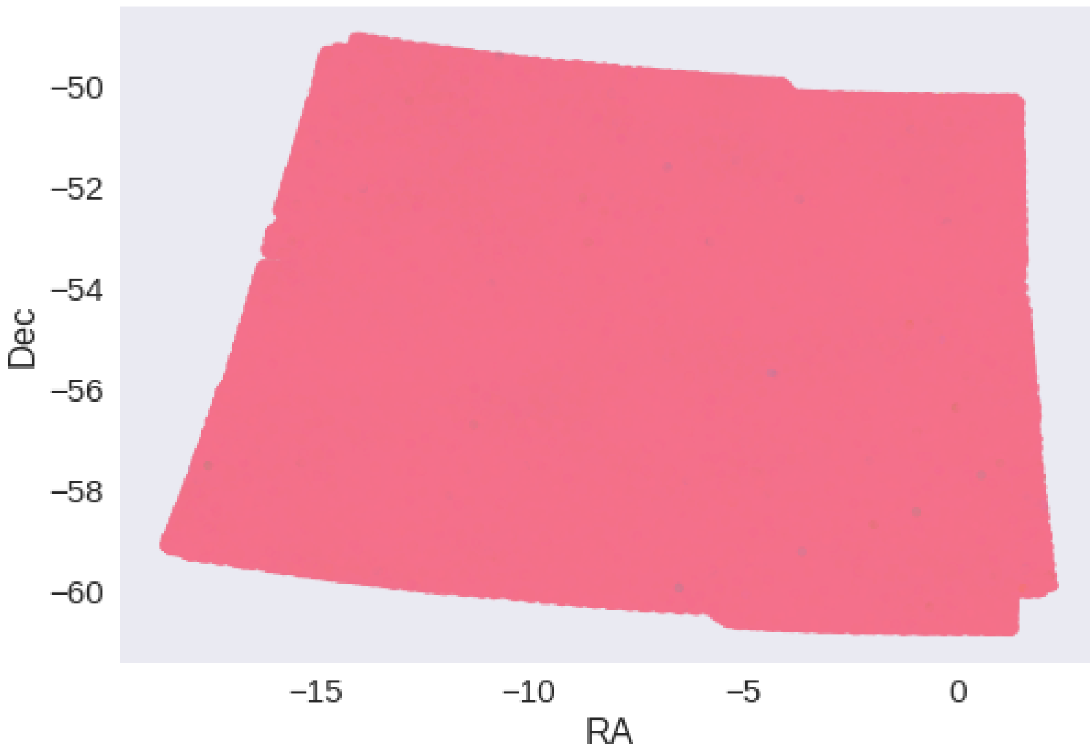
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/astropy/table/column.py:10
Check the NumPy 1.11 release notes for more information.
  ma.MaskedArray.__setitem__(self, index, value)
```

```
The initial catalogue had 9307507 sources.
The cleaned catalogue has 9307386 sources (121 removed).
The cleaned catalogue has 121 sources flagged as having been cleaned
```

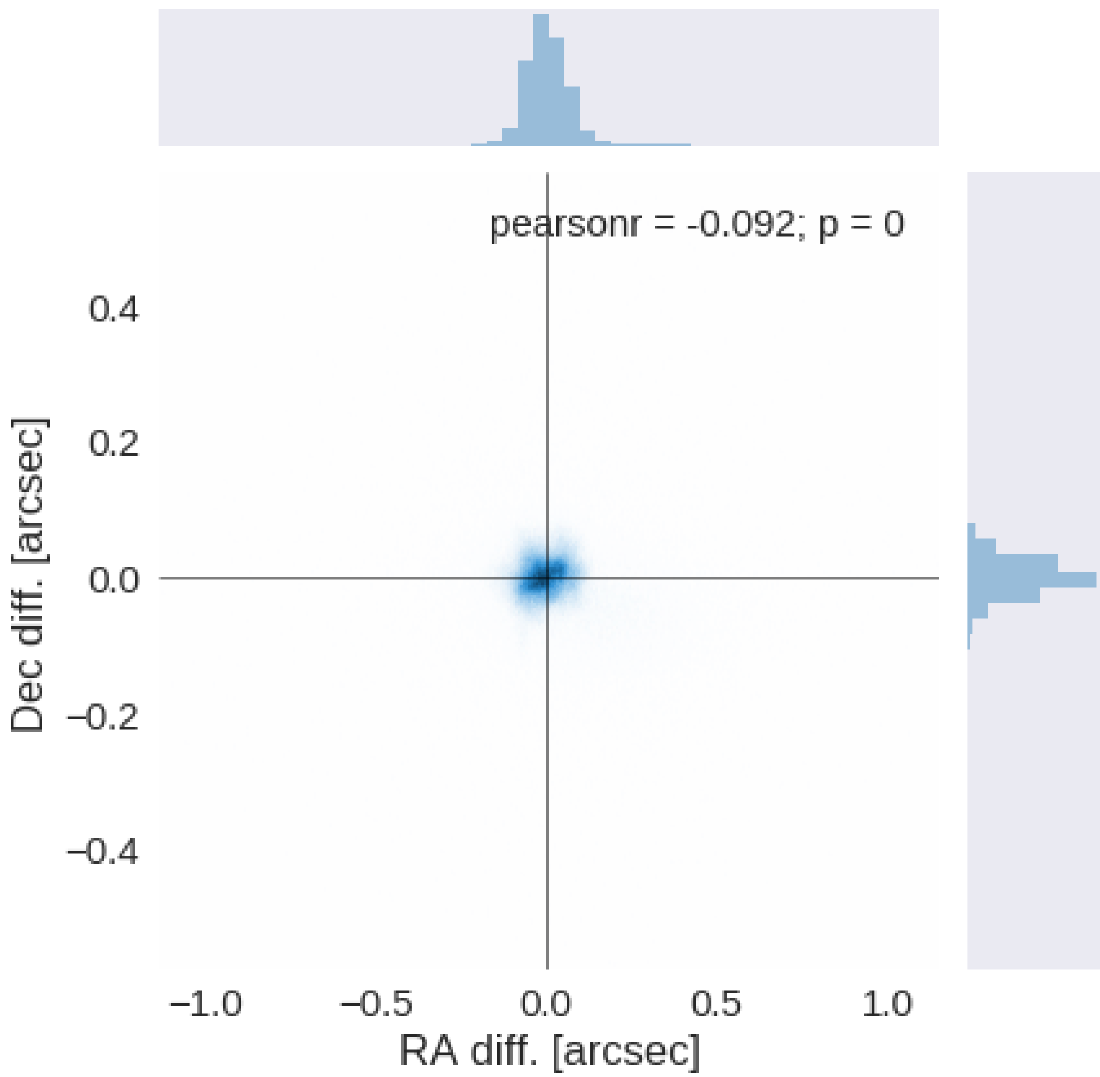
1.5 III - Astrometry correction

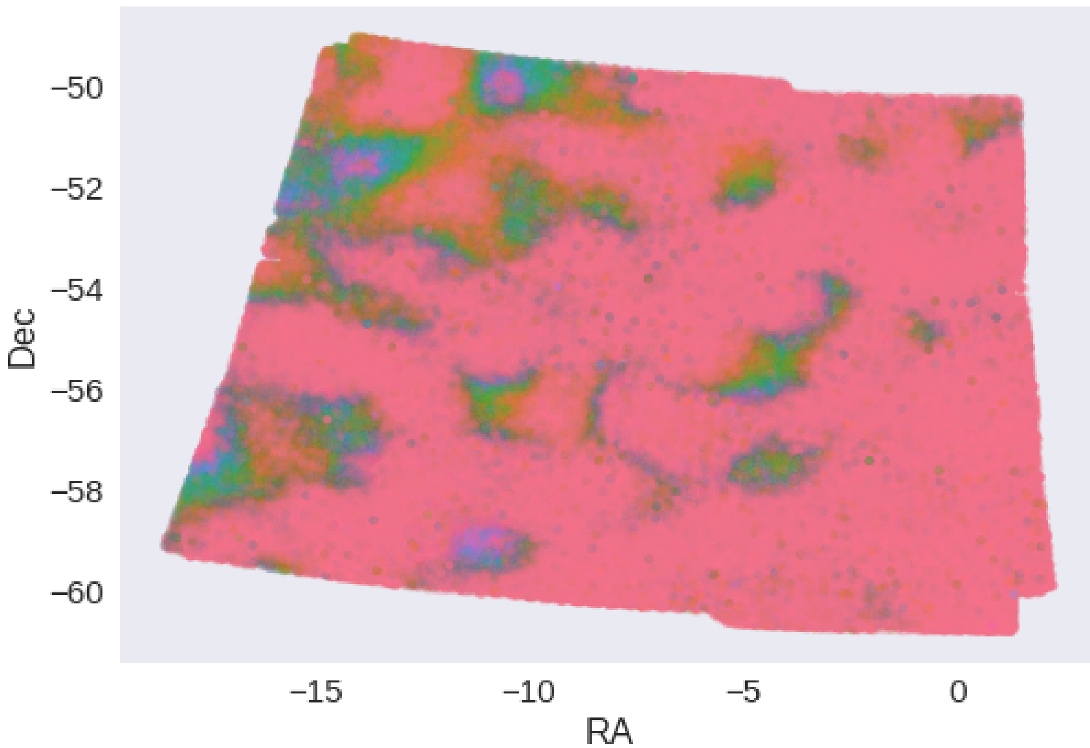
We match the astrometry to the Gaia one. We limit the Gaia catalogue to sources with a g band flux between the 30th and the 70th percentile. Some quick tests show that this give the lower dispersion in the results.






```
RA correction: 0.25125262735627985 arcsec
Dec correction: -0.10440259524102657 arcsec
```


pearsonr = -0.092; p = 0
0.4
0.2
0.0
−0.2
−0.4
Dec diff. [arcsec]
−1.0
−0.5
0.0
0.5
1.0
RA diff. [arcsec]



1.6 IV - Flagging Gaia objects

```
454986 sources flagged.
```

1.7 V - Flagging objects near bright stars

2 VI - Saving to disk

2_Merging

March 8, 2018

1 SSDF master catalogue

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

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

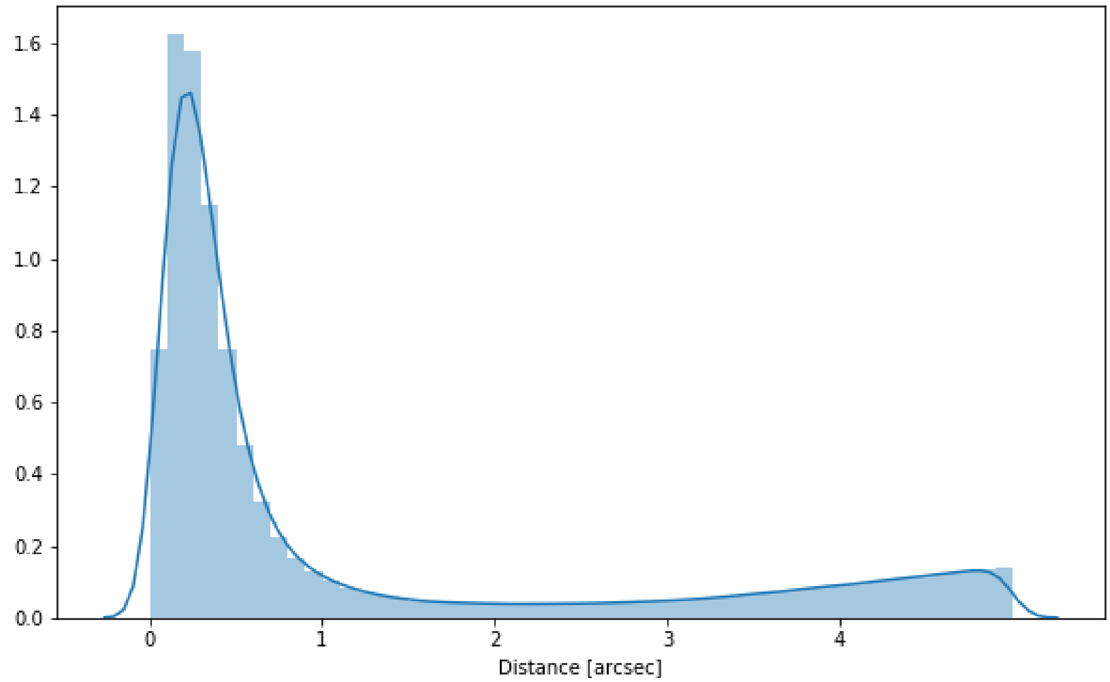
1.1 I - Reading the prepared pristine catalogues

1.2 II - Merging tables

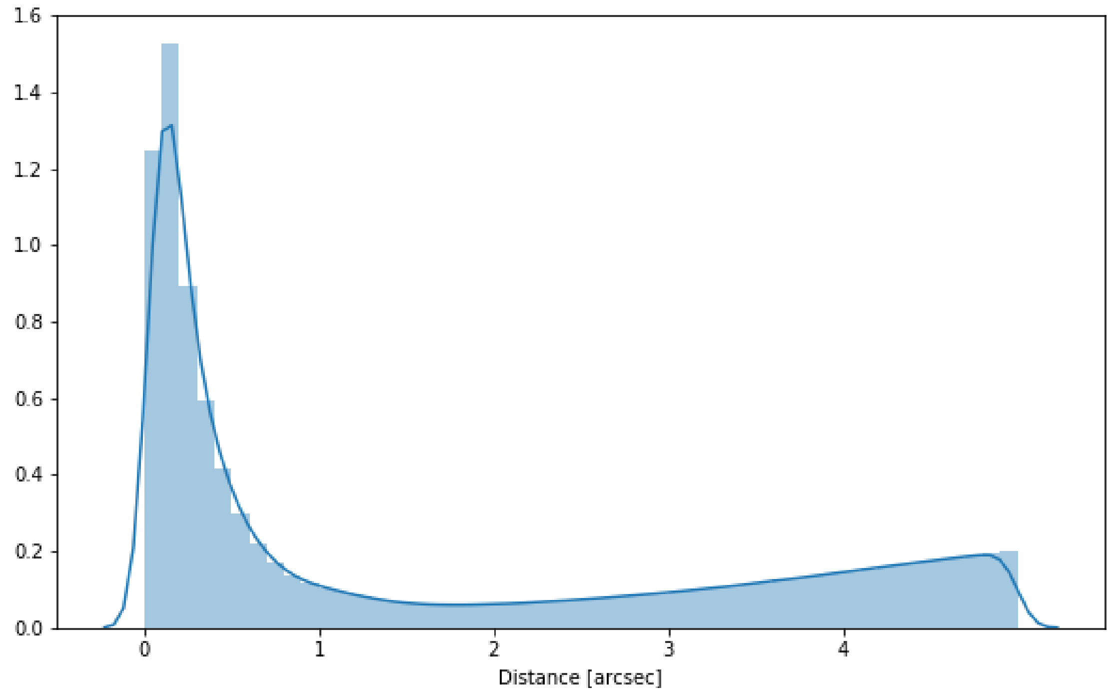
At every step, we look at the distribution of the distances to the nearest source in the merged catalogue to determine the best crossmatching radius.

1.2.1 VISTA-VHS

1.2.2 Add SSDF



1.2.3 Add DES



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 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.

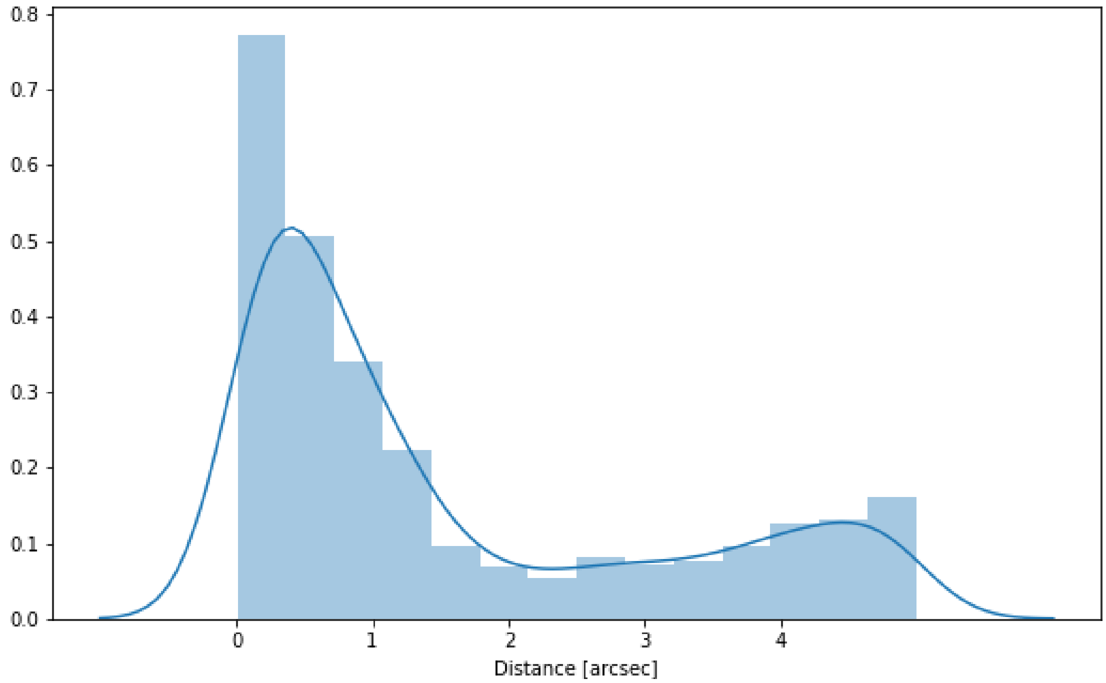
```
vhs_stellarity, ssdf_stellarity, des_stellarity
```

1.4 IV - Adding E(B-V) column

1.5 V - Adding HELP unique identifiers and field columns

```
OK!
```

1.6 VI - Cross-matching with the spec-z catalogue



1.7 VII - Choosing between multiple values for the same filter

There are no duplicate bands.

1.8 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.

1.9 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.

1.10 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.

```
['vhs_id', 'ssdf_id', 'des_id', 'help_id', 'specz_id']
```

1.11 X - Adding HEALPix index

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

1.12 XI - Saving the catalogue

```
Missing columns: set()
```

3_Checks_and_diagnostics

March 8, 2018

1 SSDF 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]
```

```
Diagnostics done using: master_catalogue_ssdf_20180221.fits
```

1.2 0 - Quick checks

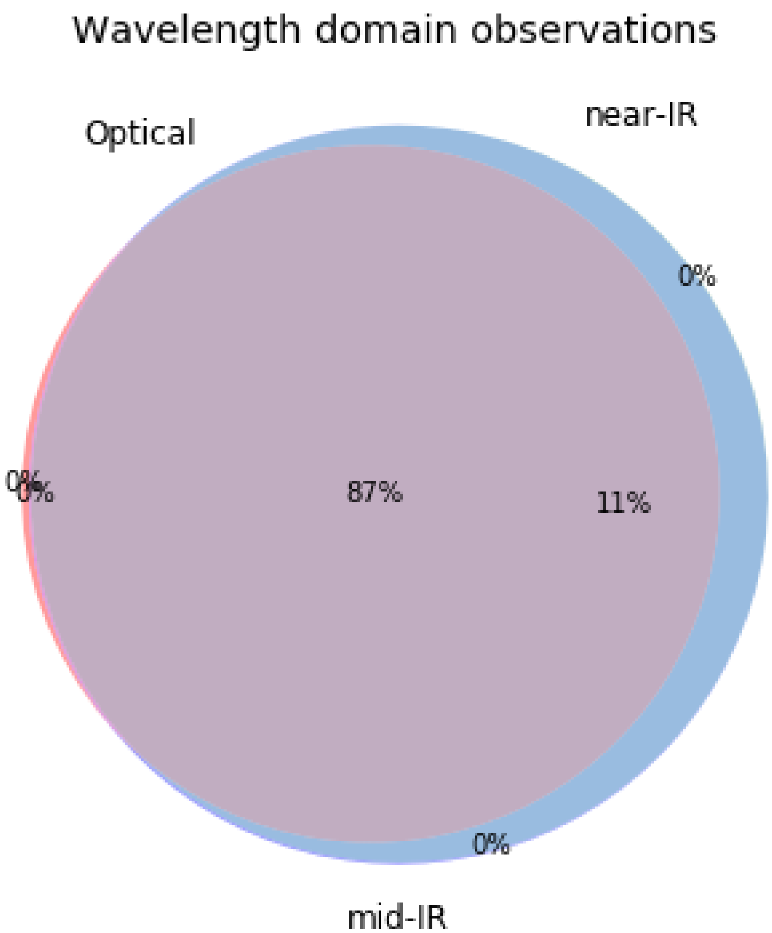
```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/core/numeric.py:301:
  format(shape, fill_value, array(fill_value).dtype), FutureWarning)
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/numpy/core/numeric.py:301:
  format(shape, fill_value, array(fill_value).dtype),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

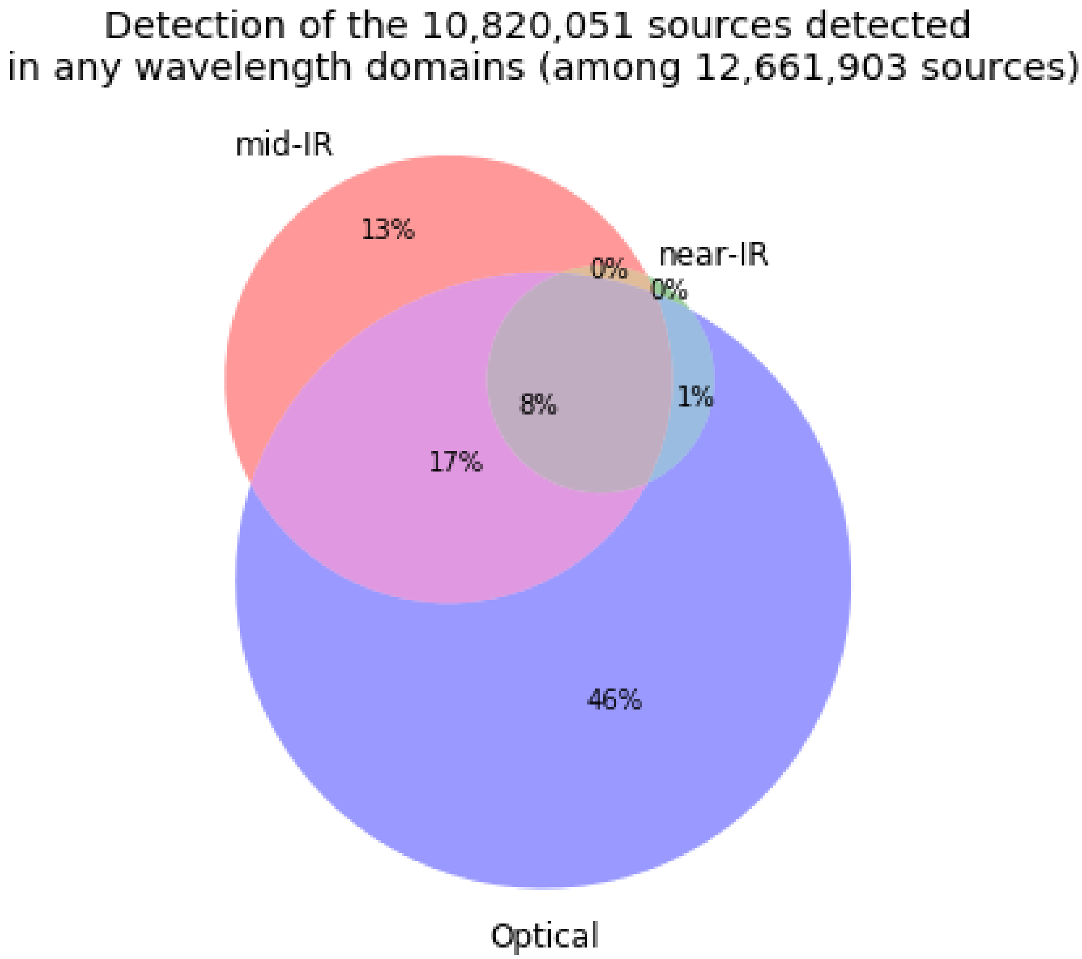
```
Table shows only problematic columns.
```

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

1.3 I - Summary of wavelength domains

```
/opt/anaconda3/envs/herschelhelp_internal/lib/python3.6/site-packages/matplotlib_venn/_venn3.py:
  warnings.warn("Bad circle positioning")
```


Wavelength domain observations
Optical
near-IR
0%
0%
0%
87%
11%
0%
mid-IR

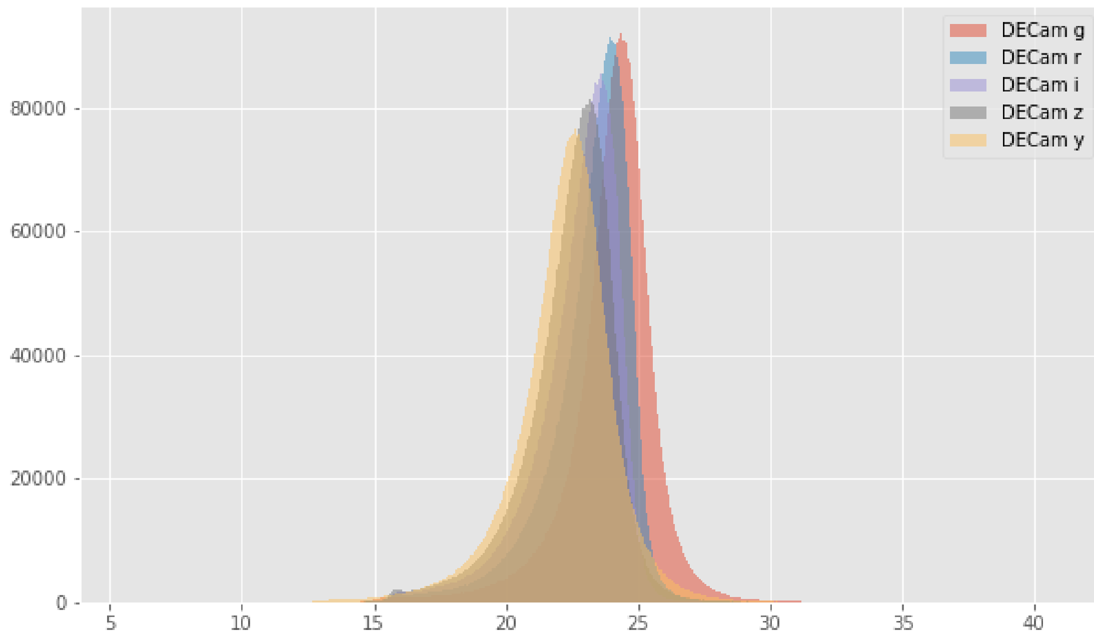


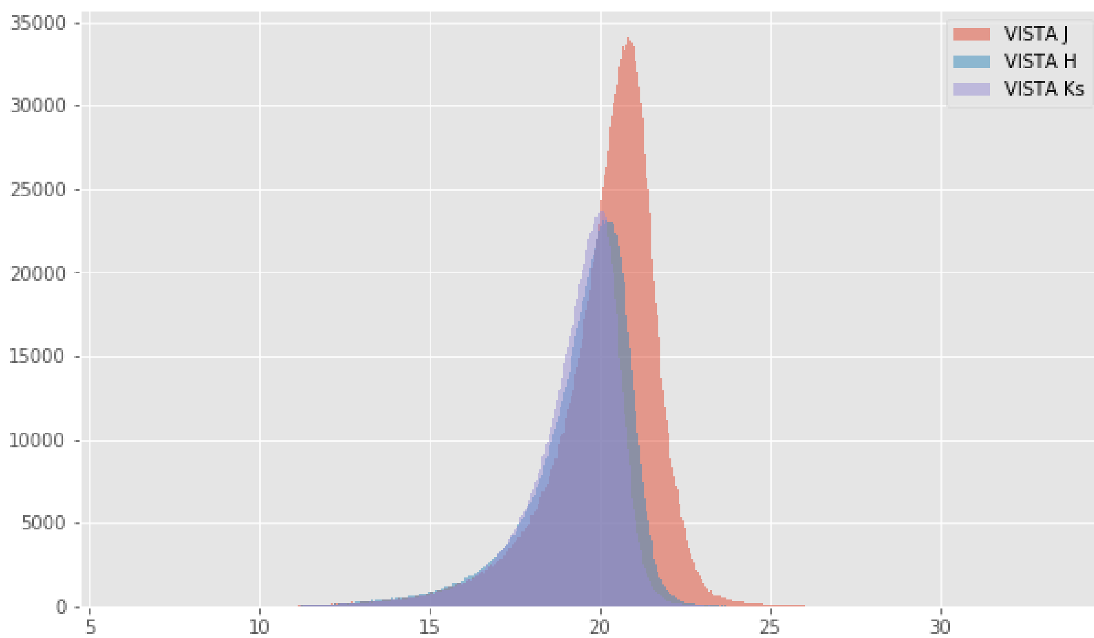
1.4 II - Comparing magnitudes in similar filters

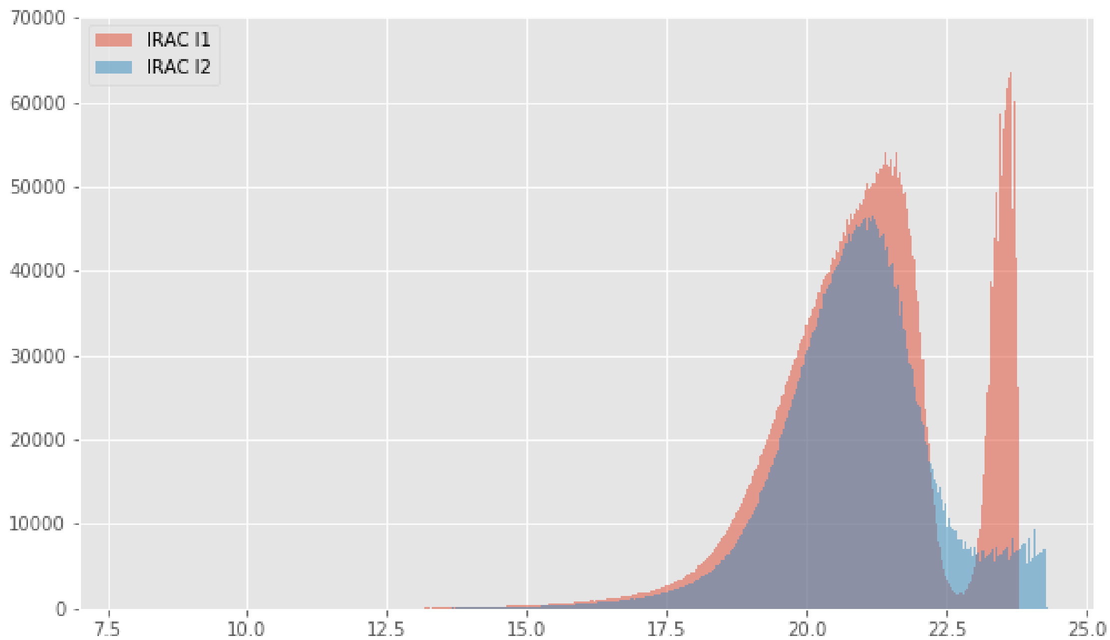
The master list if composed of several catalogues containing magnitudes in similar filters on different instruments. We are comparing the magnitudes in these corresponding filters.

1.4.1 II.a - Comparing depths

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


DECam g
DECam r
DECam i
DECam z
DECam y
80000
60000
40000
20000
0
5
10
15
20
25
30
35
40


VISTA J
VISTA H
VISTA Ks
35000
30000
25000
20000
15000
10000
5000
0
5
10
15
20
25
30



1.4.2 II.b - Comparing magnitudes

There are no similar bands from different instruments.

1.5 III - Comparing magnitudes to reference bands

Cross-match the master list to SDSS.

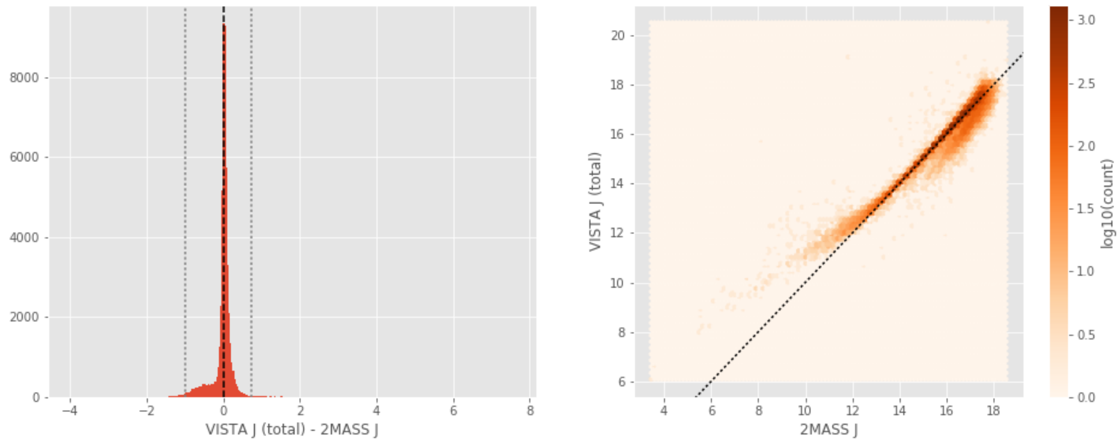
1.5.1 III.b - Comparing J and K bands to 2MASS

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.

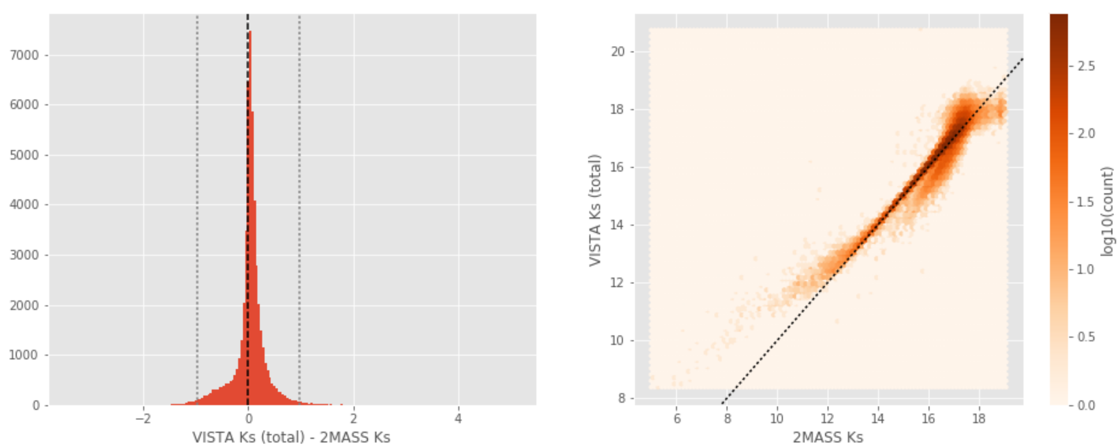
The 2MASS magnitudes are "*Vega-like*" and we have to convert them to AB magnitudes using the zero points provided on this page:

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

```
VISTA J (total) - 2MASS J:
- Median: 0.03
- Median Absolute Deviation: 0.06
- 1% percentile: -0.9993877432157341
- 99% percentile: 0.7214355603120987
```

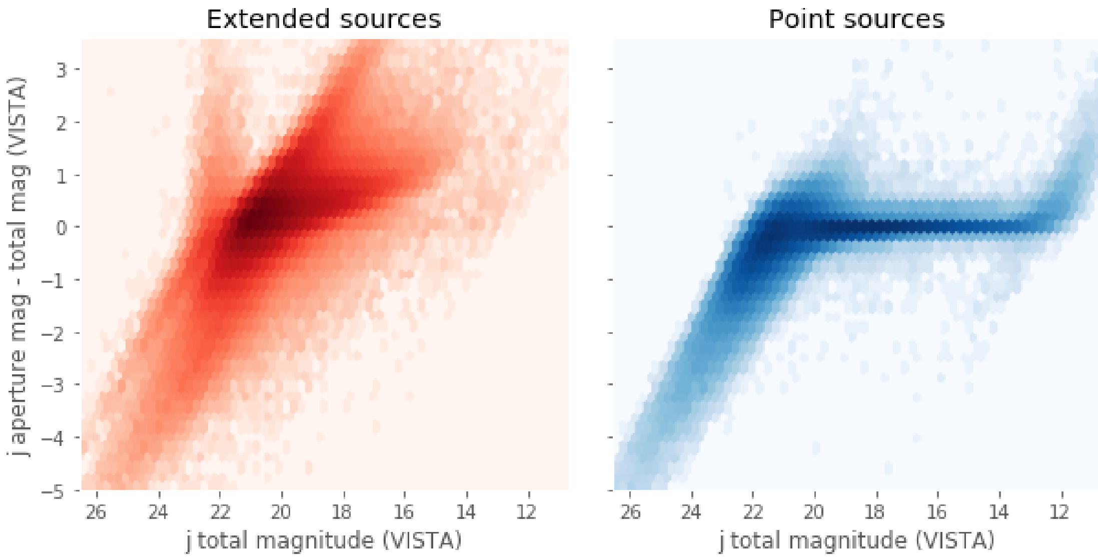


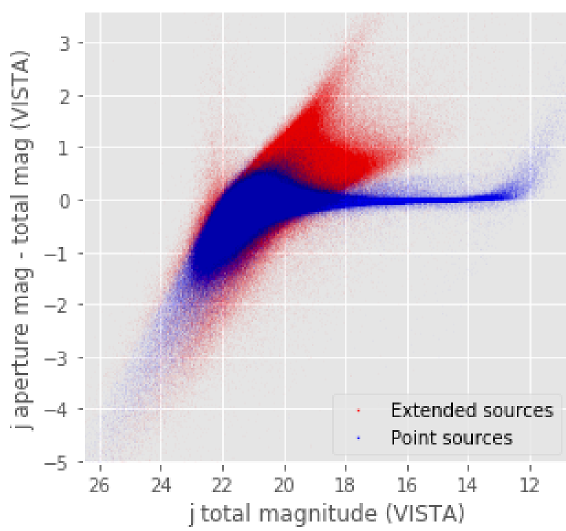

```
VISTA Ks (total) - 2MASS Ks:
- Median: 0.04
- Median Absolute Deviation: 0.09
- 1% percentile: -0.9690106682451767
- 99% percentile: 0.9662497443524783
```



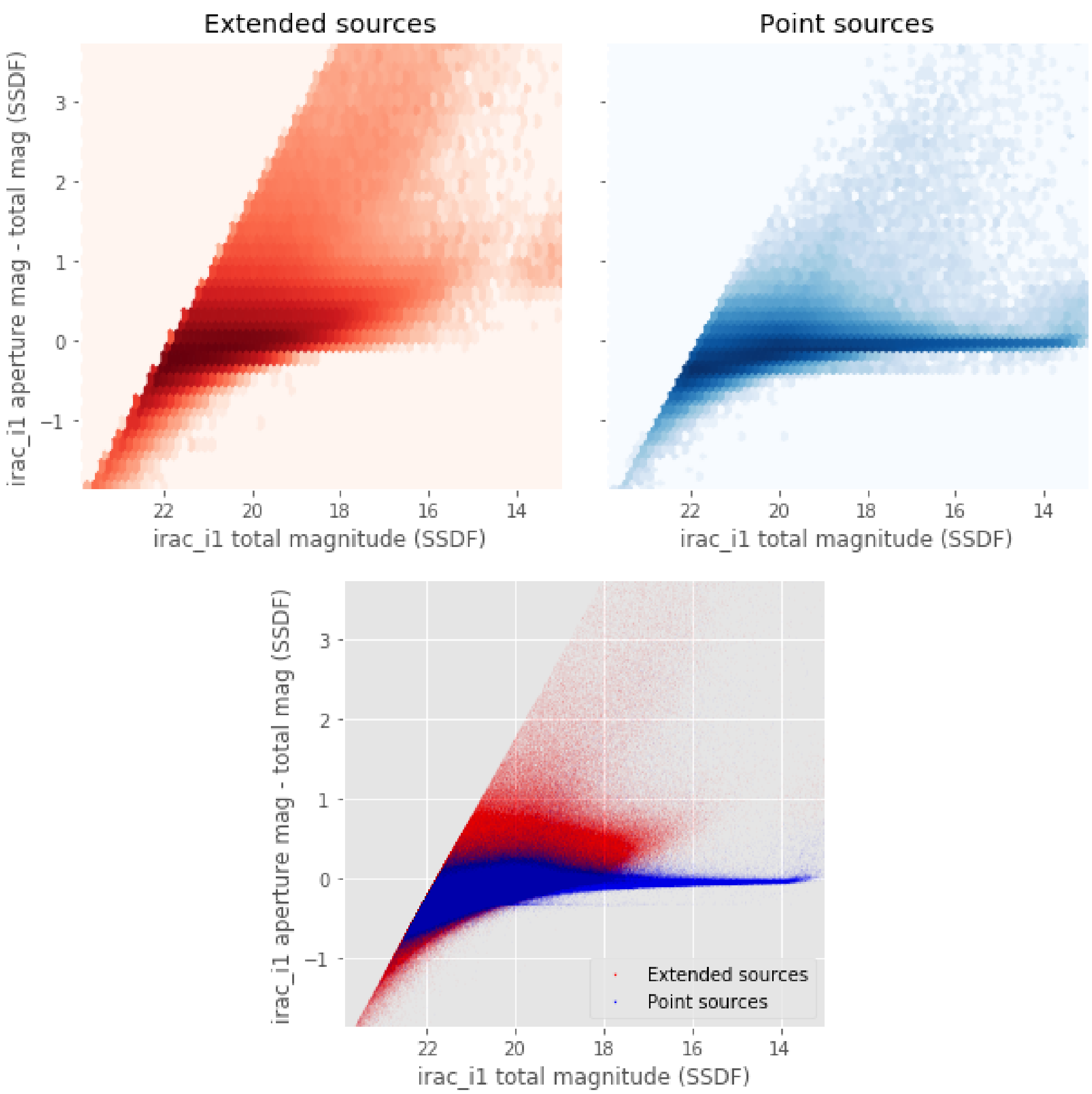
1.6 IV - Comparing aperture magnitudes to total ones.

```
Number of source used: 2057720 / 12661903 (16.25%)
```



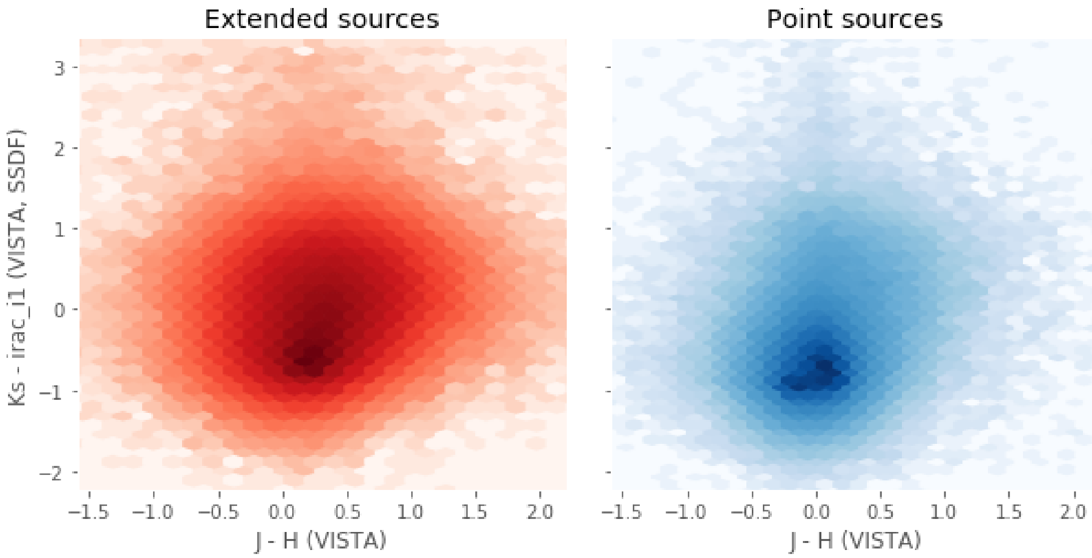


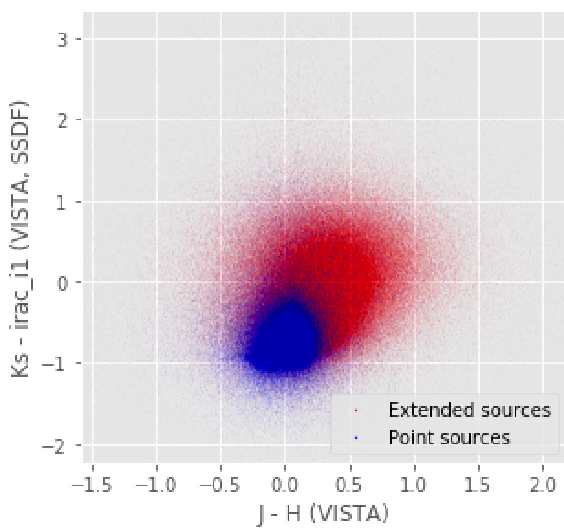
Number of source used: 4239455 / 12661903 (33.48%)



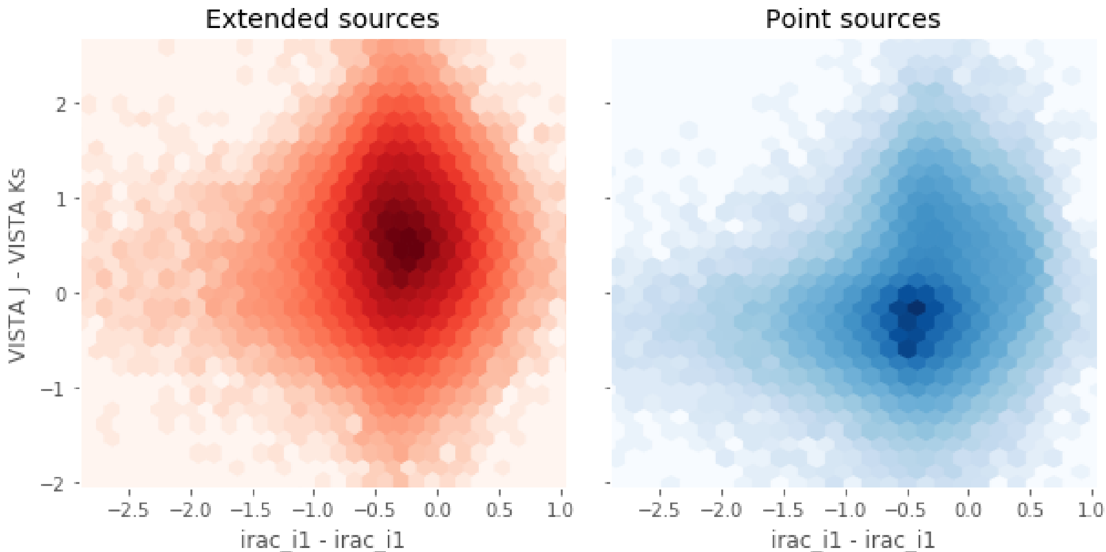
1.7 V - Color-color and magnitude-color plots

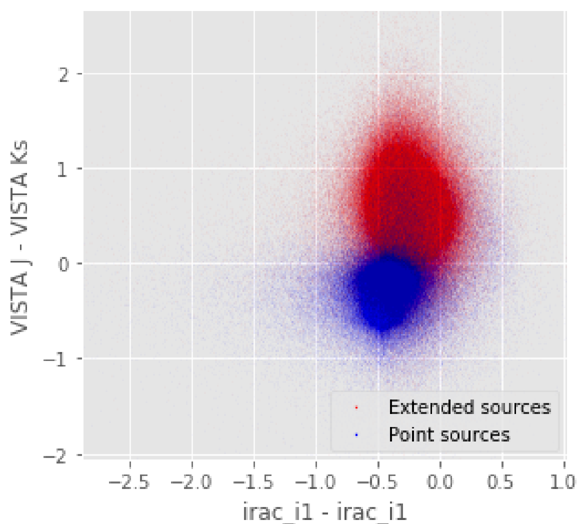
```
Number of source used: 745557 / 12661903 (5.89%)
```





Number of source used: 797742 / 12661903 (6.30%)


Extended sources
Point sources
VISTA J - VISTA Ks
irac_i1 - irac_i1
irac_i1 - irac_i1


VISTA J - VISTA Ks
irac_i1 - irac_i1
Extended sources
Point sources

4_Selection_function

March 8, 2018

1 SSDF 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:
2018-02-28 11:56:29.478992
```

```
Depth maps produced using: master_catalogue_ssdf_20180221.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

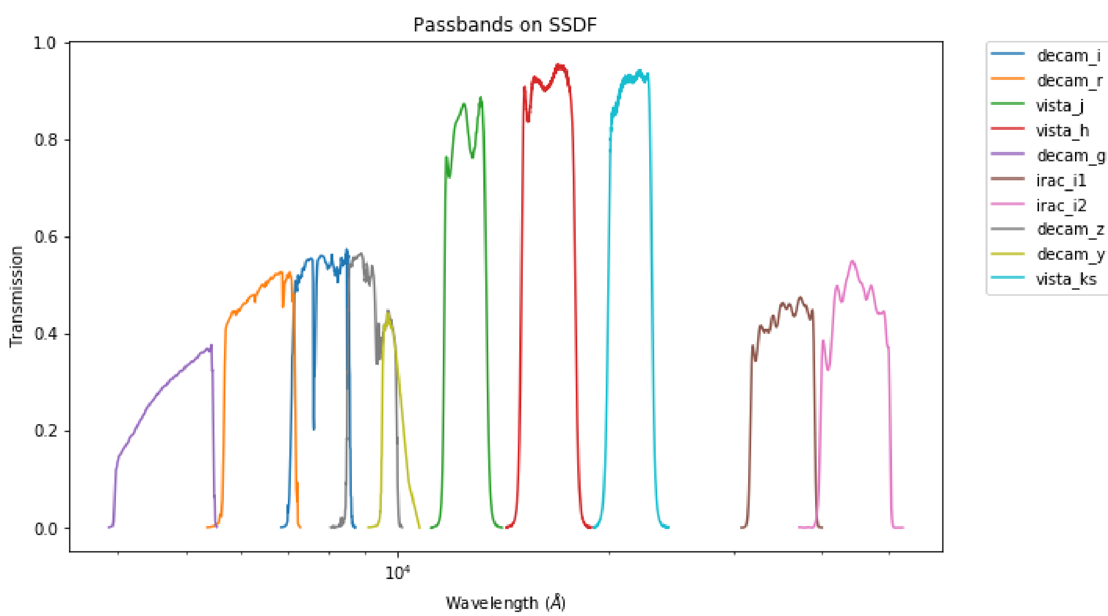
1.5 IV - Overview plots

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]: {'decam_g',
          'decam_i',
          'decam_r',
          'decam_y',
          'decam_z',
          'irac_i1',
          'irac_i2',
          'vista_h',
          'vista_j',
          'vista_ks'}
```

```
Out[15]: <matplotlib.text.Text at 0x7f9e9fd69e48>
```



1.5.2 IV.a - Depth overview

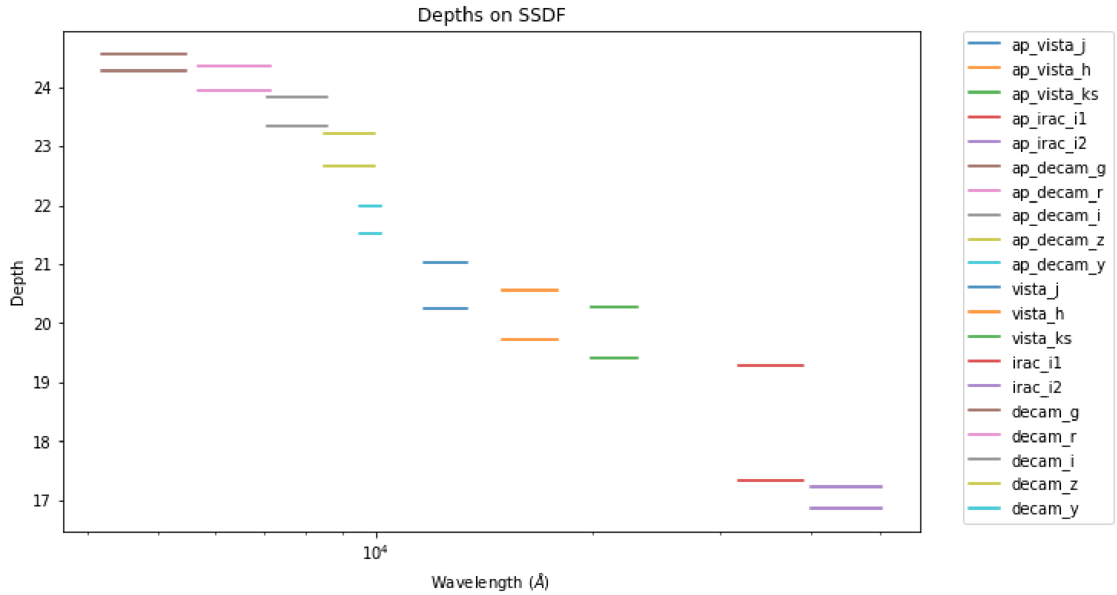
Then we plot the mean depths available across the area a given band is available

```
vista_j: mean flux error: 2.8469159603118896, 3sigma in AB mag (Aperture): 21.57126024530843
vista_h: mean flux error: 4.345226764678955, 3sigma in AB mag (Aperture): 21.112165748362862
vista_ks: mean flux error: 5.582312107086182, 3sigma in AB mag (Aperture): 20.84016157760791
irac_i1: mean flux error: 85.45731353759766, 3sigma in AB mag (Aperture): 17.877823772998816
```

```
irac_i2: mean flux error: 92.01163482666016, 3sigma in AB mag (Aperture): 17.797589995338804
decam_g: mean flux error: 0.10788271051841146, 3sigma in AB mag (Aperture): 25.12481723963301
decam_r: mean flux error: 0.13242671804157846, 3sigma in AB mag (Aperture): 24.90225782326261
decam_i: mean flux error: 0.21237251026390613, 3sigma in AB mag (Aperture): 24.389451111995264
decam_z: mean flux error: 0.3743137466144476, 3sigma in AB mag (Aperture): 23.77410742118841
decam_y: mean flux error: 1.151574241760951, 3sigma in AB mag (Aperture): 22.55396700799789
vista_j: mean flux error: 5.782756328582764, 3sigma in AB mag (Total): 20.801859631785398
vista_h: mean flux error: 9.392768859863281, 3sigma in AB mag (Total): 20.275212775160163
vista_ks: mean flux error: 12.380193710327148, 3sigma in AB mag (Total): 19.97537826306813
irac_i1: mean flux error: 14.00174617767334, 3sigma in AB mag (Total): 19.84174136185556
irac_i2: mean flux error: 130.55184936523438, 3sigma in AB mag (Total): 17.41773929242374
decam_g: mean flux error: 0.14072265950195587, 3sigma in AB mag (Total): 24.83628677767515
decam_r: mean flux error: 0.19366893763945733, 3sigma in AB mag (Total): 24.489546937546088
decam_i: mean flux error: 0.33351221299309886, 3sigma in AB mag (Total): 23.899417507911302
decam_z: mean flux error: 0.6294928812792083, 3sigma in AB mag (Total): 23.209719805243004
decam_y: mean flux error: 1.8069069035025214, 3sigma in AB mag (Total): 22.06485742028314


ap_vista_j (11670.0, 13380.0, 1710.0)
ap_vista_h (15000.0, 17900.0, 2900.0)
ap_vista_ks (19930.0, 23010.0, 3080.0)
ap_irac_i1 (31754.0, 39164.801, 7410.8008)
ap_irac_i2 (39980.102, 50052.301, 10072.199)
ap_decam_g (4180.0, 5470.0, 1290.0)
ap_decam_r (5680.0, 7150.0, 1470.0)
ap_decam_i (7090.0, 8560.0, 1470.0)
ap_decam_z (8490.0, 9960.0, 1470.0)
ap_decam_y (9510.0, 10170.0, 660.0)
vista_j (11670.0, 13380.0, 1710.0)
vista_h (15000.0, 17900.0, 2900.0)
vista_ks (19930.0, 23010.0, 3080.0)
irac_i1 (31754.0, 39164.801, 7410.8008)
irac_i2 (39980.102, 50052.301, 10072.199)
decam_g (4180.0, 5470.0, 1290.0)
decam_r (5680.0, 7150.0, 1470.0)
decam_i (7090.0, 8560.0, 1470.0)
decam_z (8490.0, 9960.0, 1470.0)
decam_y (9510.0, 10170.0, 660.0)
```

```
Out[20]: <matplotlib.text.Text at 0x7f9e9f9770f0>
```



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 0x7f9e9f9c5668>