Alberto Krone-Martins

Open Cluster candidate flagging method challenge

————————————————————————————————

—— **Perseus Recommender Challenge : preparing for the full sky search**

————————————————————————————————

- **Deadline : 15th May 2019**

- **Objective:** To define a very fast recommender strategy to select candidate directions of the sky where we should launch later UPMASK (just to be pragmatic; but it could be any other previously validated method) to search for *bonna-fide* new open clusters and to derive stellar membership lists for each newly detected cluster. A candidate direction is defined as the center of a "data cluster" (equivalent to candidate cluster) in the positional space (here, the "l" and "b" galactic coordinates).

- **Background information:** The recommender strategy will be used in the follow-up paper for https://arxiv.org/abs/1810.05494 . In this follow-up, we plan to search new clusters in the entire sky (or at the very minimum, in the entire plane of the Milky Way).

- **Tradeoff consideration:** it is more important for the recommender strategy results to be more complete than to be more pure, as long as the contamination level is reasonable. Note: with the dataset adopted in this challenge it is not expected that we will discovery many hundreds of new true clusters in regions where we previously knew only a couple.

- **Baseline solution:** the baseline solution was implemented in R and it is very simple, holding a fast HDBSCAN implementation in its core. It works by tilling the dataset in the positional coordinates and running HDBSCAN inside the tiles. To perform the

tiling as fast as possible, it creates regions by first slicing the dataset in equal parts of "dL" degrees in the galactic longitude direction, then by slicing again each slice in the galactic latitude direction to keep a maximum number "maxObjPerBlock" of sources per resulting region. Then it runs HDBSCAN in each region using 20 as the minimum number of objects to define a "data cluster" (minPts=20 in HDBSCAN terminology) and using ("l", "b", "parallax", "pmra", "pmdec") as the clustering dimensions. Finally, this method was executed in a grid spanning different values of "dL" and "maxObjPerBlock" to optimize these tiling parameters, so the method would detect the highest possible number of known clusters while still keeping the number of possible new credible candidate regions to look for new clusters much smaller than the number of original tiles.

We define a credible new candidate region as the (l, b) coordinates of the selected "data clusters" (computed from the median position of the cluster members) with a proper motion dispersion <= 0.2 mas/yr. To estimate the dispersion, we adopt the median absolute deviation.

No further refinement was used in the baseline (e.g. a simple smoothing or regression on the dL vs. maxObjPerBlock grid might allow a selection of a more optimal parameter set, perhaps resulting in slightly better results).

- **Expected outputs :**

1. One txt file with the same name as the proposed recommender strategy, containing:

```
a. The fraction of known clusters in Perseus independently detected as
candidates.

b. The total number of candidates.

c. The total number of credible candidates as defined by a total proper
motion dispersion <= 0.2 mas/yr.
```

2. The recommender strategy output should be a CSV file, with one candidate per line, containing at least the following columns:

```
- clusterN     = cluster candidate Id
- medianL      = median galactic longitude of the candidate cluster
members
- medianB      = median galactic latitude of the candidate cluster
members
- madL         = mad of the galactic longitude of the candidate cluster
members
- madB         = mad of the galactic latitude of the candidate cluster
members
- regDL        = span (max - min) of the galactic longitude of the
candidate cluster members
- regDB        = span (max - min) of the galactic latitude of the
candidate cluster members
- medianPmRa   = median of the candidate cluster members proper motions
in RA
- medianPmDec  = median of the candidate cluster members objects proper
motions in DEC
- madPmRa      = MAD of the candidate cluster members proper motions in
RA
- madPmDec     = MAD of the candidate cluster members proper motions in
DEC
- medianPlx    = median of the candidate cluster members parallax
- madPlx       = mad of the candidate cluster members parallax
- nObj         = number of candidate cluster members
- possibleName = if the candidate cluster is a known cluster, the name
of the known cluster
```

- **Important : Keep communication on Slack.**

————————————————————————

—— Description of the dataset

————————————————————————

The dataset is composed by two files:

- **GaiaDR2-UPMASK-OC-centroids-arxiv-COINGaiatemp.dat.lrz**

md5sum : 8add3a2129be01998c1b9571

datalink:

https://drive.google.com/file/d/1XGFRhJmsNRKdxqsGtLn4yXhbHV-LOF5J/view?usp=sharing

UPDATE [17th April 2019] : Alternatively, you can use the file **PerseusArmDataLGe120Le200_withErrorColumns.fits**, that includes error columns for the astrometric data.
md5sum : 2e9fba47fd742e30c10daca5c7535a61

datalink:

https://drive.google.com/file/d/1QjYrtguuTuAE1LvQVsl_rqOhkeA3k058/view?usp=sharing

- **PerseusArmDataLGe120Le200.csv.lrz**

md5sum : 727c6589f305c4b2bf4d71d8

datalink:

https://drive.google.com/file/d/19EcNyZVEjC0kCmeBdxs9XAQTHXAbyAR-/view?usp=sharing

The csv files are compressed with the lrzip tool. You can easily install this tool using 'apt-get install' in any apt-enabled Linux distribution (e.g. xBuntu), in the Linux extension in Windows, or by using homebrew in OSX. Otherwise, you can also naturally compile lrz yourself.

The **GaiaDR2-UPMASK-OC-centroids-arxiv-COINGaiatemp.dat** file contains information for a list of known clusters in the region.

The **PerseusArmDataLGe120Le200.csv** file contains the Gaia DR2 positions, parallaxes, proper motions and magnitudes in the G, GBP, GRP passbands of the Gaia sources in the region of Perseus. The file was capped in Gmag <= 18, and it doesn't contain error columns. Uncompressed, it should attain ~1.9 GB.

The **PerseusArmDataLGe120Le200_withErrorColumns.fits** file contais the same information as the PerseusArmDataLGe120Le200.csv and additionally it includes ra, dec, and the errors columns of each astrometric column plus the correlations between the astrometric parameters. It attains ~1.9GB, and it is a binnary compressed fits; this format was used in place of the CSV to keep the filesize bellow 2GB.

The SQL query used to produce the PerseusArmDataLGe120Le200_withErrorColumns.fits file directly using the Gaia Archive is simply :

```
SELECT l, b, parallax, pmra, pmdec, phot_g_mean_mag, phot_bp_mean_mag,
phot_rp_mean_mag, ra, dec, parallax_error, pmra_error, pmdec_error,
ra_dec_corr, ra_parallax_corr, ra_pmra_corr, ra_pmdec_corr, dec_parallax_corr,
dec_pmra_corr, dec_pmdec_corr, parallax_pmra_corr, parallax_pmdec_corr,
pmra_pmdec_corr FROM gaiadr2.gaia_source WHERE l >= 120 AND l <= 200 AND b >=
-10 AND b <= 10 AND phot_g_mean_mag <= 18
```

——————————————

—— **Reminders**

——————————————

1. To always keep in mind: Be pragmatic. The objective here is to create a very fast flagging method to flag candidate regions where we would look for new clusters. This method must scale to the entire sky and hundreds of millions of points.

2. Gaia Early-Data Release 3 is arriving in mid-2020. Lets complete this challenge quickly, so we can submit the All-Sky search paper by the end of July-2019 and have it published by end-2019. In this way, we will be able to quickly respond to Gaia EDR3 at the moment it touches the Internet using a previously validated (and hopefully successful) methodology.

3. This is a challenge, not a competition. It might be that at the end we do not decide for one flagging method, but adopt a candidate region flagging set that is composed by the union of the sets produced by different methods. The reasoning here is that each method might be biased towards a selection of candidate regions on its own way, and by joining the results of independent methods, we would improve the completeness of the final candidate region set (while the purity of the final cluster list will likely be assured by UPMASK plus expert human inspection).