Project Rubric

Project Overview

For hobby astronomers available data on astronomical objects and literature related to each object is available in large quantities through portals such as:

- CDS Portal
- Simbad
- SDSS Portal (if the portion of the sky has been covered by SDSS survey)
- NED Portal
- ADS Portal

The time required to go through all the material when planning an observation session for one particular could be reduced by implementing a recommender system that would associate scientific papers to astronomical objects.

A thorough presentation of details on this project is available also at:

- Blog post

- GitHub

Problem Statement

There's plenty of literature available on studies in the field of astronomy and it is time consuming to scroll through the material. SIMBAD service offers for instance a way of scoring objects to scientific papers, but if one looks closer, the criteria is really around that one object (was it mentioned in the paper or just in a table, is in the title, and so on).

What if two objects are similar? may be one has an associated paper that could be useful to understand something crucial about the other. And how about papers in a region of space around an object? How could such a list of papers be put together to reflect my own journey in learning about the sky, such that actually those objects that I plan to see and with which I actually work get relevant associations?

Metrics

As metrics to check the built model following standards have been used:

- Precision – a discussion on this will show that this is not necessary optimal
- Recall – a discussion on this will show that this is helpful, but not enough
- By manually quality check of results – checking for example top 30 results on NGC 3115 yielded about 57% relevance.

In addition SSE was used to assess the fit of the FunkSVD; furthermore a self created score was used to find a balance between number of clusters and size of individual clusters (how many objects get assigned to one particular centroid)

Analysis - data collection, exploration and visualization

The download procedure covers two cases:

CASE I: Download data from SDSS (as described above)

```
python download_sdss.py
python read_sdss.py
python main_reco.py update 'salamander'
python main_reco.py papers 'salamander' ,1237654652032516161'
```

CASE II: Download data from Simbad:

- Get information for object coordinates:
- For example using CDS portal and look for the object: e.g. http://cdsportal.u-strasbg.fr/?target=NGC%202169
- Use **read_simbad.py** script to download and process data (this would download information about objects around M1 – including M1 itself)

```
python read_simbad.py  05 34 31.940 +22 00 52.20
```

To make a recommendation on the object either rebuild the model first or run the recommendation directly:

```
python main_reco.py update 'salamander'
```

And then ask for recommendations:

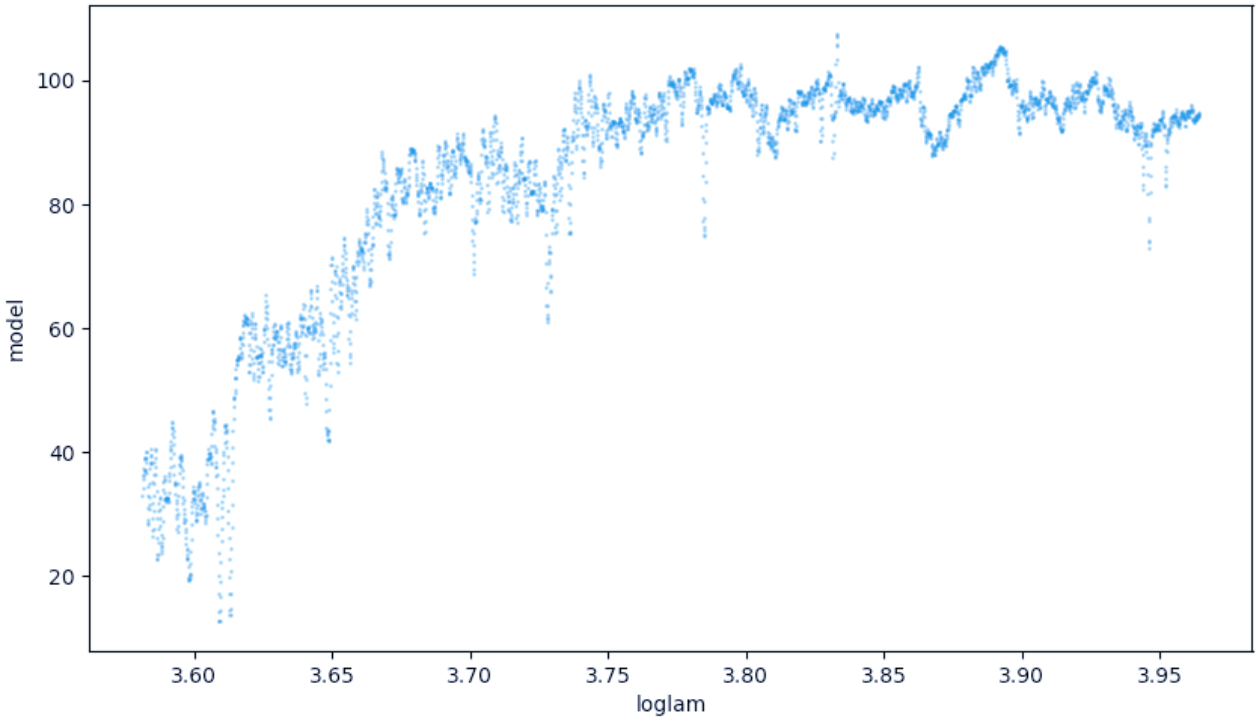
```
python main_reco.py papers 'salamander' 'M 1‘
```

In both cases one can skip the re-building of the model and just run it on the newly downloaded objects. This way the result contains only papers that have been previously processed by the model.

For any FITS file spectra data can be visualized as follows:

```
>>> hdu = read_sdss.load_data(spec_filename="spec-0277-51908-0002.fits")
>>> data = pd.DataFrame(hdu[1].data.tolist(), columns=hdu[1].columns.names)
>>> data.plot.scatter(x="loglam", y="model", s=1, alpha=0.3); plt.show()
<AxesSubplot:xlabel='loglam', ylabel='model'>
```

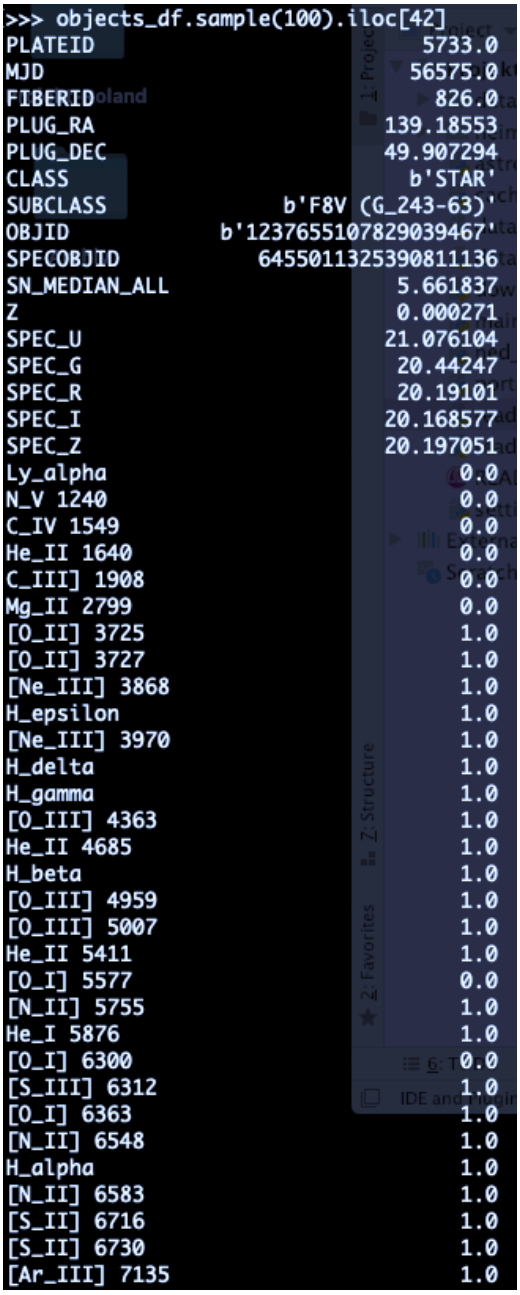
```
>>> hdu[0].header['PLUG_RA']
166.28839
>>> hdu[0].header['PLUG_DEC']
-0.79695882
>>> hdu[0].header['SPEC_ID']
'    311874855962175488'
>>>
```



Absorption and emissions lines are also available in the FITS file:

```
>>> data = pd.DataFrame(hdu[3].data.tolist(), columns=hdu[3].columns.names)
>>> data[["LINENAME", "LINEWAVE", "LINEZ", "LINECHI2"]]
          LINENAME     LINEWAVE     LINEZ    LINECHI2
0   b'Ly_alpha     '  1215.670000  0.000000  -1.000000
1   b'N_V 1240     '  1240.810000  0.000000  -1.000000
2   b'C_IV 1549    '  1549.480000  0.000000  -1.000000
3   b'He_II 1640   '  1640.420000  0.000000  -1.000000
4   b'C_III] 1908  '  1908.734000  0.000000  -1.000000
5   b'Mg_II 2799   '  2800.315184  0.000000  -1.000000
6   b'[O_II] 3725  '  3727.091727  0.033124  88.443237
7   b'[O_II] 3727  '  3729.875448  0.033124  88.074921
8   b'[Ne_III] 3868'  3869.856797  0.033124  47.967651
9   b'H_epsilon    '  3890.151080  0.033124  28.343519
10  b'[Ne_III] 3970'  3971.123187  0.033124  43.589069
11  b'H_delta      '  4102.891631  0.033124  28.613831
12  b'H_gamma      '  4341.684313  0.033124  42.016186
13  b'[O_III] 4363 '  4364.435300  0.033124  36.172642
14  b'He_II 4685   '  4686.991444  0.033124  30.272455
15  b'H_beta       '  4862.682994  0.033124  15.600189
16  b'[O_III] 4959 '  4960.294901  0.033123  31.333370
17  b'[O_III] 5007 '  5008.239638  0.033124  24.371544
18  b'He_II 5411   '  5413.024423  0.033124  27.571526
19  b'[O_I] 5577   '  5578.887704  0.033125  27.015083
20  b'[O_I] 6300   '  6302.046377  0.033125  16.315561
21  b'[S_III] 6312 '  6313.805533  0.033124  21.043619
22  b'[O_I] 6363   '  6365.535420  0.033124  15.256874
23  b'[N_II] 6548  '  6549.858929  0.033124  25.136196
24  b'H_alpha      '  6564.613894  0.033124  19.248644
25  b'[N_II] 6583  '  6585.268445  0.033123  30.240868
26  b'[S_II] 6716  '  6718.294208  0.033124  19.743107
27  b'[S_II] 6730  '  6732.678076  0.033125  15.038312
28  b'[Ar_III] 7135'  7137.757103  0.033125  16.350618
```

The final result of reading data from the FITS files and building a pandas dataframe looks then as follows:



Since this data has already been analyzed and the signal already processed, it would be best to use the flux values from the **model (column ‚model'),** rather than those from column flux. This way one doesn't need to take care of variations due to signal error processing or other factors that could have influenced measurements (although it is still possible to get that). According to the documentation, the values in column „model" were used for calculating redshift, so this is another argument in favor of its usage.

From the header at **index 0**, basic information such as:

- PLUG_RA, PLUG_DEC
- SPEC_ID

Flux data is therefore available in the header data unit (HDU) at position **index 1**, and according to documentation: model is in float32 and represents best model fit used for classification and redshift.

At **index 2**, thus hud[2], further information is available as follows (values are examples for illustration purposes):

- SPECTROSYNFLUX: array([95.50256, 380.8138 , 877.2078 , 1253.3973 , 1637.0831] Flux – these are the values we're looking for to fill up the passbands data
  - Calculating the magnitude in the corresponding passband following formula must be used: 22.5 – 2.5 * LOG_10(SPECTROFLUX)
- Signal to Noise ratio available in SN_MEDIAN_ALL: 61.872798919677734,
- Type of object in field CLASS: b'GALAXY and SUBCLASS
- Redshift is given in field Z: 0.028583228588104248
- Several IDs are provided:
  - BESTOBJID: b'1237654652032516161′, (can access the Explore page http://skyserver.sdss.org/dr16/en/tools/explore/Summary.aspx?id=1237654652032516161) (notice usage of id)
  - ‚SPECOBJID': b' 8624517358456934 40′, (can access spectra plot http://skyserver.sdss.org/dr16/en/get/SpecById.ashx?id=8624517358456934 40)
  - TARGETOBJID: b' 9179435761713′ (notice usage of id)

Obviously making the right assignment to object ID is one of the tasks all data surveys have to manage. For this work BESTOBJID value was used, if available, otherwise the value in TARGETOBJID.

Finally at **index 3** there is a rich data about the emission and absorption lines that were detected for this object.

Regarding passbands they can be ordered as follows; values are in nanometer:

U = 365
B = 445
G = 464
V = 551
R = 658
I = 806
Z = 900
Y = 1020
J = 1220
H = 1630
K = 2190

So one will need to assign possible values to **passbands that are missing**. This will only be possible if some of the values are available, such that for instance mean values between neighbor passbands are possible, e.g. V as mean(G, R).

Methodology

All details presented in this part are executed by rebuilding model as described at blog post

python reco.py update ‚salamander'

In the following part the central idea of collaborative filtering, thus building a matrix user to items, will be implemented; this part will be combined with calculation of probability for association using a classification with an MLP model.

Data preprocessing

Dataset of astronomical objects can be seen as a „**user**“ matrix while the dataset of articles can be seen as „**items**„. Building on this idea, follows that there is a way to build up two matrices and perform a recommendation using the sorting capabilities given by collaborative filtering method.

In order to achieve this several steps were taken:

1. Prepared data on articles as a pandas dataset containing for each scientific paper:
    1. the result of text processing of title, description
    2. Turning keywords associated to articles in dummy variables
2. Prepared data from FITS files or SIMBAD:
    1. By reading the downloaded FITS files
    2. By crawling Simbad website
    3. Spectrum data in passbands U, G, R, Z, I when missing: at first by averaging values

Implementation

There are **thousands** of astronomical objects that come on the list of a hobby astronomer (given that this number depends on the instruments available for observation) and **many more** other that are in the dataset only for analysis purposes (since for most objects there's no way to actually observe them at the eyepiece). In this case the dimensionality of the resulting matrix of features will be very big. This leads to two measures one can implement:

- **PCA** – this will project the vectors for each object to a reduced orthonormal coordinate system; this will result in construction of matrix G
- **Groupings** by clusters, so we could consider the centroids as „users" instead of objects themselves. Later on one can find for each object the closest center and look up for best recommendation for it. This information will be added as feature to the U matrix for later use (collaborative filtering part).

As a result of this, the process starts by building a matrix **U** with following set of features:

- type of object
- redshift
- passband values (u, g, r, i, z)
- emission and absorption lines

The size of matrix U, thus dimension m, grows dynamically (and personally biased) according to user's needs, anytime when SDSS or Simbad downloads are executed. Due to the fact that Simbad will not have all data for features such as passband, emission and absorption lines, a naive trial was used to replace those values by mean. Later added an improvement to calculate means between passbands. A further improvement of this would be to replace NaN values by looking at types of objects and only then perform replacement. Another version would require a much more advanced technique by replacing the emission lines values by templating, since these values are known for standard type of astronomical objects; the challenge here would then be to adjust values for redshift.

Out of text data for articles, there are following types of features available:

- title, abstract – can be transformed to vectors by NLP techniques
- keywords – will be treated as binary variables (for example: „galaxies" tag)

As a result the matrix **A** will also have a high dimensionality and thus **PCA** will be best fitted at this point to reduce the dimension resulting in matrix **A***. From this point on both matrices **G**, described earlier, and **A*** can now be combined such that for each record from matrix **G** (initial matrix **U** of objects) all combinations from matrix **A*** get associated.

Very quick one will get again to dimensionality problem of **M**, since most of the associations haven't been found to be so; one knows already if one object is associated to one scientific paper, if it was so found on NED at download time.

One method to reduce dimensionality of matrix **M** is to construct all associations between objects (matrix **G**) and articles (matrix **A***) according to found data and **only then append by an arbitrary factor**, e.g. 3, more data associations. The reason one should not have half-half of the data labeled as 1 and 0 is because data is intrinsically not balanced.

In the end we have a target variable (1 if association was given, 0 otherwise). This leads to the possibility to build up a classifier that will predict the probability of association between one centroid object (clustered **G** matrix) and potential articles represented by processed vectors in **A***.

On one side the positive aspect of this method is that it contains a bias based on own research studies, thus only those regions of the sky get included where sky observing sessions take place, on the other side there is a bias and dependency on NED output; this second aspect is not something one can adjust in a practical manner, since too much crawling only leads to damaging NED servers performance, possibly getting IP banned. Notice that the **M** matrix is based on PCA form of **U** (now **G** matrix) for all objects.

Worth mentioning at this point, that for this the model used is a classifier MLP that helps calculate the probability of association.

The next part the process includes building up a **D** matrix as a user-item matrix and apply SVD.

The idea in using collaborative-filtering method would be to only use the 1s for associations we know of and null values for all 0s were one doesn't know for sure:

- matrix **D** contains the centroids of the resulted clustering on matrix **G**
- Null values are used if there is no association between centroid and scientific paper
- FunkSVD gets applied to determine intermediary U, VT matrices
- Finally the resulting matrix gets SVD-decomposed and predictions of articles for each centroid are then those values on rows in the resulting matrix after constraint on latent features got applied (e.g. 15 in this case).

The last step of the procedure is combining both results from **MLP** and **SVD**. On one side, MLP predicts a probability, then SVD sorts scientific articles for each object and delivers scores that are around value 1.0 if association is expected and 0.0 if not.

Running a complete process including model building from collected data, one can see that predicting the association isn't a trivial task, for several reasons:

- precision of the voting model is relatively low, because many of potential articles are not associated; the classifier provides probabilities by employing a threshold for ‚associated' at 50% while the SVD method provides a sorting of papers per object. The precision isn't a very good metric for assessing performance, because NED did not provide really links to all possible articles (from other regions of sky), but rather concentrate on object itself. This issue is easily seen if one looks up the recommended papers which are studies on sample of objects with focus on, for instance, early-type galaxies, motion of galaxies,

metallicity parameters, and so on; in these cases often studies use a sample of galaxies – the object itself fits the sample, but it's not part if it.

- recall has a fairly higher value (in the sense „over a random model"), which means many of associated objects do get included in the final results

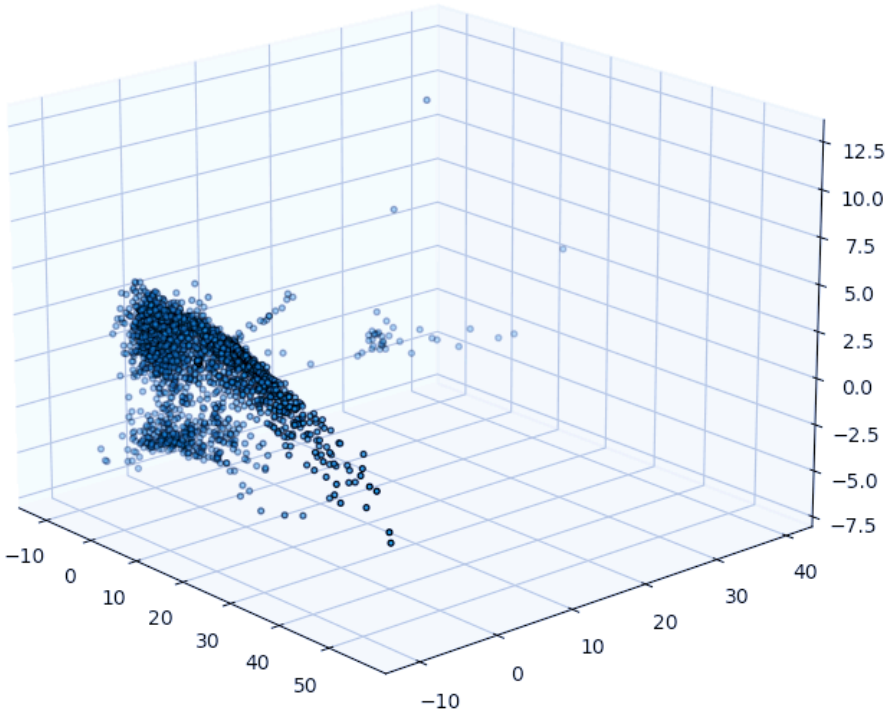
Both PCA results with matrix **G** and **A*** show cluster structures, which enforces the decision that was made to use clustering in order to group objects in the orthonormal space defined by the principal components.

Running a complete process including model building from collected data, one can see that predicting the association isn't a trivial task, for several reasons:

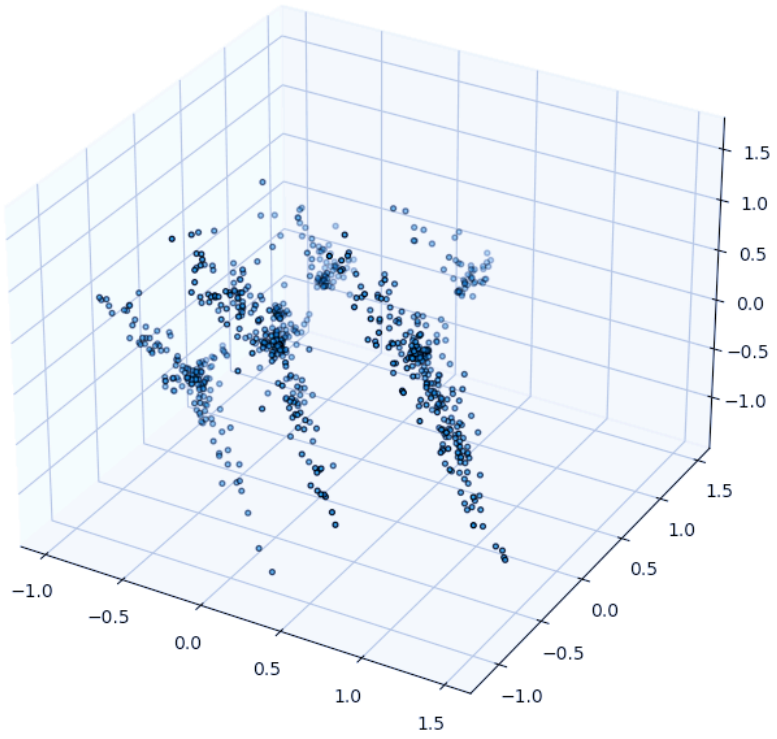
- precision of the voting model is relatively low, because many of potential articles are not associated; the classifier provides probabilities by employing a threshold for ‚associated' at 50% while the SVD method provides a sorting of papers per object. The precision isn't a very good metric for assessing performance, because NED did not provide really links to all possible articles (from other regions of sky), but rather concentrate on object itself. This issue is easily seen if one looks up the recommended papers which are studies on sample of objects with focus on, for instance, early-type galaxies, motion of galaxies, metallicity parameters, and so on; in these cases often studies use a sample of galaxies – the object itself fits the sample, but it's not part if it.
- recall has a fairly higher value (in the sense „over a random model"), which means many of associated objects do get included in the final results

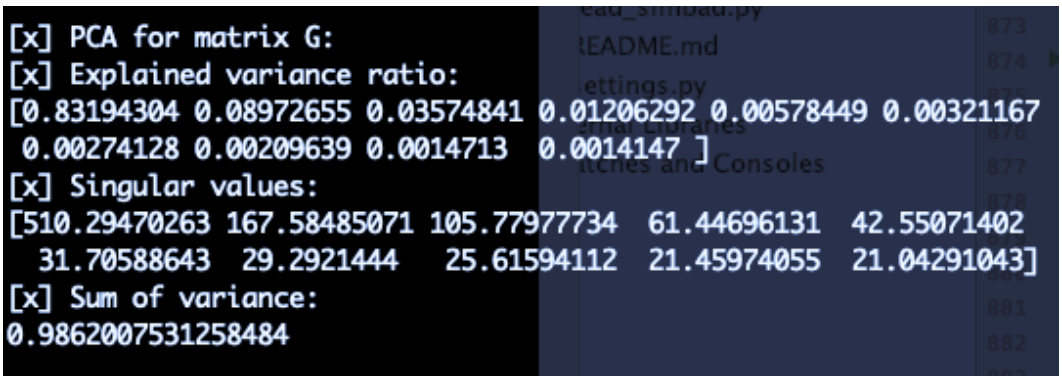
Both PCA results with matrix **G** and **A*** show cluster structures, which enforces the decision that was made to use clustering in order to group objects in the orthonormal space defined by the principal components.

First 3 principal components for G

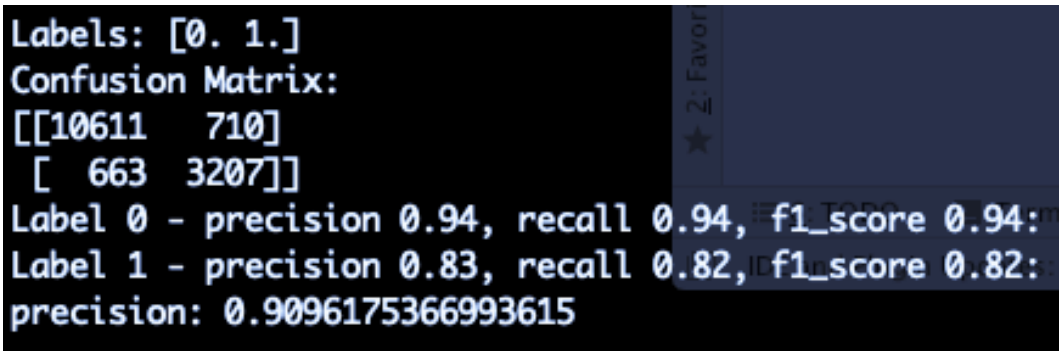


First 3 principal components for A_star



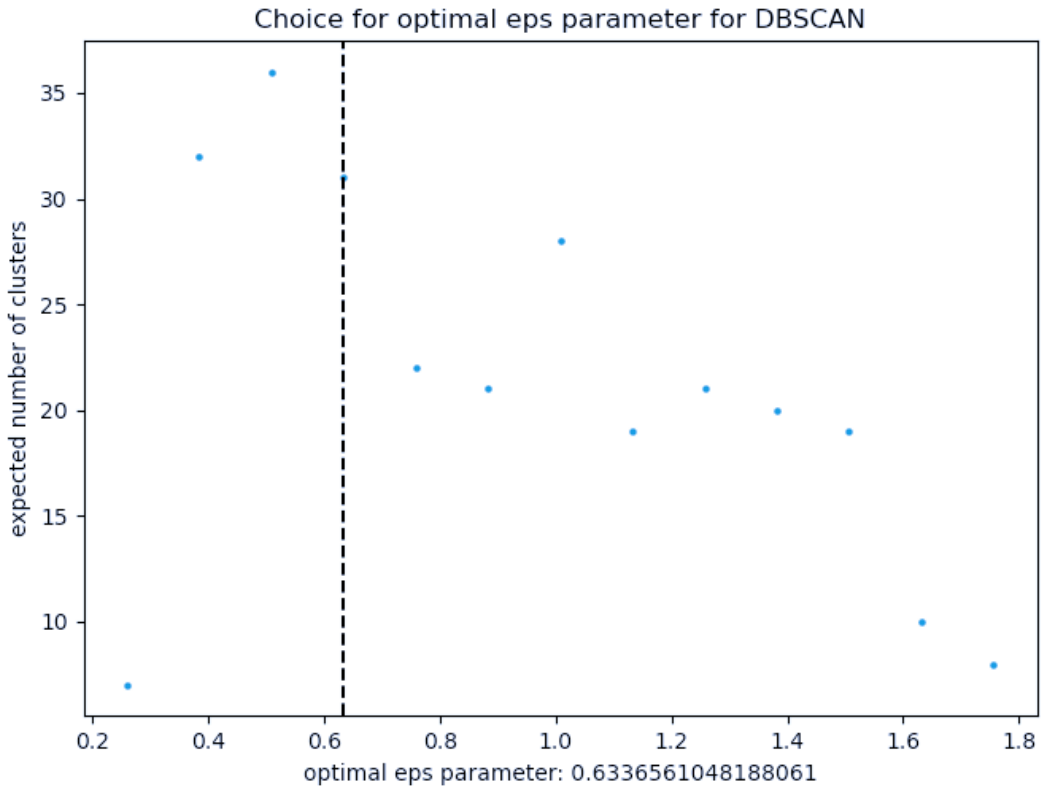


The MLP performs well, but one needs to notice that what is being predicted is the probability of having an association as found on NED which is inherently biased (due to the download procedure). So a more qualitative approach is needed to assess the effectiveness of the method. A somewhat better metric would be the recall, since the model should capture a portion of actual associations. This value lies for the MLP at about 80%.



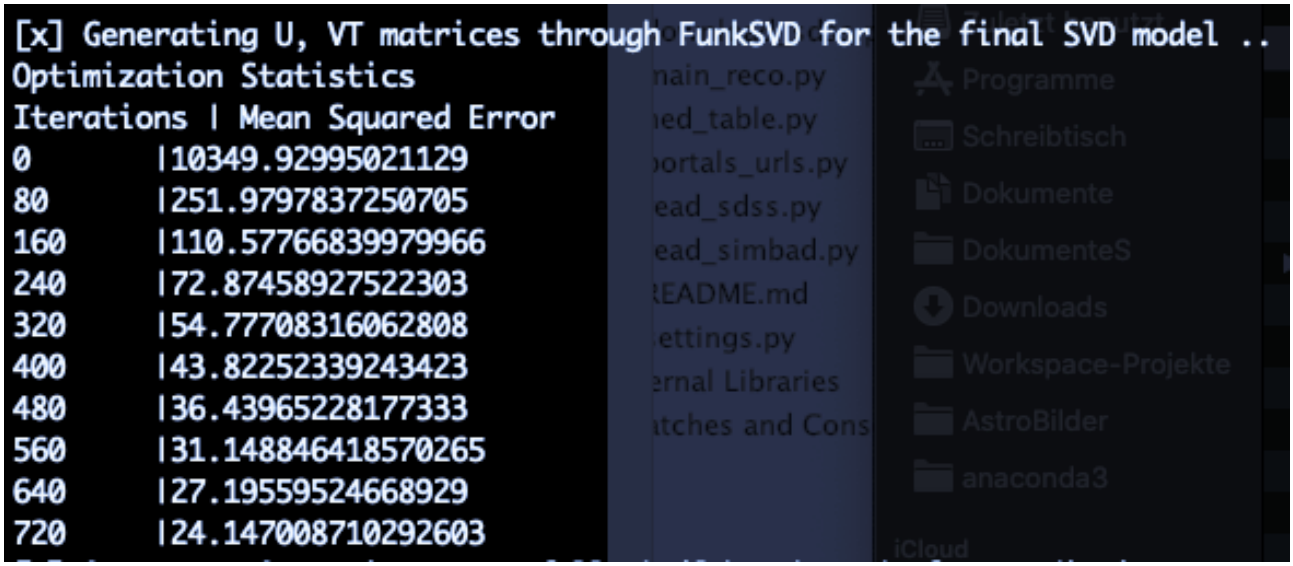
```
**********************************************************************************
article_index    316.0
mlp_proba          1.0
associated         0.0
Name: 2, dtype: float64
{'description': 'We report on the detection of a third massive star component '
                'in the σ Orionis AB system, traditionally considered as a '
                'binary system. The system has been monitored by the IACOB '
                'Spectroscopic Survey of Northern Massive Stars program, '
                'obtaining 23 high-resolution FIES@NOT spectra with a time '
                'span of ~2.5 years. The analysis of the radial velocity '
                'curves of the two spectroscopic components observed in the '
                'spectra has allowed us to obtain the orbital parameters of '
                'the system, resulting in a high eccentric orbit (e ~ 0.78) '
                'with an orbital period of 143.5 ± 0.5 days. This result '
                'implies the actual presence of three stars in the σ Orionis '
                'AB system when combined with previous results obtained from '
                'the study of the astrometric orbit (with an estimated period '
                'of ~157 years). Based on observations made with the Nordic '
                'Optical Telescope, operated on the island of La Palma jointly'
                'by Denmark, Finland, Iceland, Norway, and Sweden, in the '
                'Spanish Observatorio del Roque de los Muchachos of the '
                'Instituto de Astrofisica de Canarias.',
 'keywords': ['Simón-Díaz, S.',
              'Caballero, J. A.',
              'Lorenzo, J.',
              'binaries: spectroscopic',
              'galaxies: star clusters: individual: σ Orionis',
              'stars: early-type',
              'stars: individual: σ Ori AB',
              'Astrophysics - Solar and Stellar Astrophysics'],
 'link': 'https://arxiv.org/abs/1108.4622',
 'refcode': '2011ApJ...742...55S',
 'title': 'A Third Massive Star Component in the σ Orionis AB System'}
```

An example for predicting articles for „ sig Ori“; in this case the prediction actually gives an article that was not associated but obviously fits well with this region of the sky.*



For the implementation of clustering a search for a best fit parameter eps of DBSCAN algorithm was written. The idea was to find best balance between number of clusters and the size of individual clusters (the more well spread the better).

Finally using FunkSVD to obtain approximation of matrices U and VT leads to filling up those null values where associations are not given. Followed by an SVD approximation using 15 latent features the result is a matrix that can be used to make recommendations for centroids, thus for all objects in PCA space (given by G matrix).



The final score of the voting model considers the average of output from above presented two models:

- **MLP** predicts a probability of association
- **SVD** (originating from FunkSVD method) predicts a list of top associations
- the **voting model** considers both values by averaging to a final score

Prediction of new objects, thus unknown to the model can be difficult, but this is how it could be done:

- the object needs to have been at least downloaded from SIMBAD or SDSS
- The nearest centroid can be calculated, meaning value for field ‚group'
- required portion of matrix M can be built just for this object
- Row matrix data for G gets calculated by applying PCA model learned for G
- Classification by MLP results in first score
- Classification using the nearest centroid and access to the **UVT** matrix results in second score
- final score is the average model

This scenario is rather unrealistic, as one could just build the model upfront using the new papers downloaded for the objects of interest and leave it to the model to sub-select top k of all articles available on local machine. Nevertheless for curiosity purposes, one can ask *what papers previously downloaded can be associated to a new specific object?* thus never seen before by model. The disadvantage of this method is the A* matrix will not be updated with the new records, so the SVD part of the model will only give back data from the past.

One would then use the model on any object (either newly downloaded or known by model) as follows:

```
python main_reco.py papers 'salamander' 'NGC 3115' 30
```

or

```
python main_reco.py process 'salamander' 30 'NGC 3115'
```

This will give back 30 recommendations for desired object
(the **process** command parameter can take an object list and output a CSV file).

A **qualitative check was tried out and for** NGC 3115 about 57% of the recommendations were actually „correct“, in the sense that they treat aspects that can be related to this particular object.

During planning a session, one would most probably have several objects on his list. So the way to go about it, would be to generate one list of all related papers, for instance top 3 per object and then gather all data in a CSV file:

```
python main_reco.py process 'salamander' 3 'NGC 2566' 'NGC 2207'
'NGC 2974' 'NGC 2559' 'NGC 2292' 'NGC 2613' 'NGC 3115'
```

Refinement

Several improvements needed to be taken during the implementation of the above presented method:

- Replacement of missing values by mean for spectra was improved by taking advantage of values that the object already has in other passbands. So if values are available für G and I then R will become the average of neighboring passbands

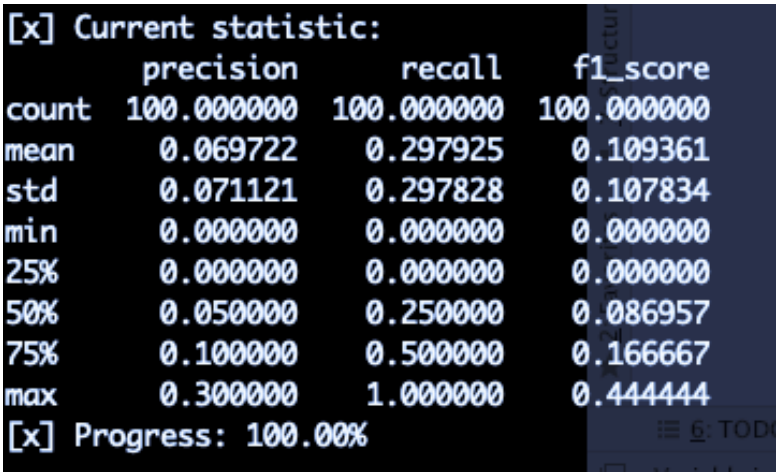
- Several trials needed to be done during the models for PCA for both matrices G (of objects) and A* (of papers). The reason was at the beginning more data was downloaded so the number of components needed to be adjusted otherwise the explained variance would be too low (less than 80%).

- On first trial the null-values were not used, but instead 0s were employed. To take advantage of the FunkSVD method capabilities, that is to be able to predict what the value for an unknown association might be, without assuming directly that it is zero, null values were employed afterwards.

Results

Model Evaluation and Validation

For assessing performance of the model, as stated above, it is not necessary the best way to use precision as a metric. Recall should be good (in the sense ‚better as random') to at least indicate that the actual associations from NED, which are legitimate, are captured by the model. But a **qualitative check** should also show that new articles are related, even if they are not getting an association from NED (either because we only downloaded top N articles or because NED didn't actually list them as related papers). This can indeed happen, as some of the papers, although not speaking about the object itself do treat subjects that are relevant to it. For instance one paper might not treat parameters of NGC 3115 but might talk about evolution of early-type galaxies (in this case type S0), such as Rapazzo 2017 et. al.



Justification

The reason this method works is because any small detail get a weight into the final model; such details could be the type of object (galaxy, QSO), its redshift (which would probably lead to QSO literature and evolution of galaxies), type of spectra (for instance literature on blue-stragglers can be related to stars that have high fluxes in the blue spectrum at high velocities, thus higher redshift)

So depending on how many articles are included in the resulting recommendation of top k papers and on parameter threshold for probability of association, one can validate the model on existing data which makes it clear that the task of predicting the right associations isn't trivial:

Conclusion

Reflection

We’ve came from downloading data from SDSS or Simbad, processing it, using association to scientific papers from NED and ADS to creation of a repository with data on astronomical objects in pandas dataframes as well as, through NLP techniques, processed data on those papers that are associated to them according to mentioned services.

By building up a classifier with an MLP one can now calculate a probability for association between any object and any paper followed by an adaptation of the user-item idea from collaborative filtering method by setting known associations to 1s while non associations to null values, so that by usage of FunkSVD method followed by SVD decomposition one is able to sort all papers for any object by a DBSCAN cluster „proxy“. In order to reduce dimensionality and keep building a manageable user-matrix on local machine PCA techniques were employed. Finally a voting model was used to calculate a score based on values from both prediction classifiers (one gives the order of papers, the other improves position indexes by probability of association). Validation of the model was conducted and both good and bad aspects of these mechanisms were addressed as well as a quality check on the main example of NGC 3115 was conducted leading to an approximated relevance of 57%.

As could been seen collecting data, cleaning and building up the connections between astronomical objects and scientific papers can get quite complex. This results in a model that is not easily testable in a fair way, since constraints and biases limit data collection as well as the process of labeling data.

Nonetheless the resulting model, as it turns out, delivers interesting results and enough material is made available to further once's own learning process in a step by step manner, with each sky observing session.

Another interesting aspect is naturally the ability of this process to perform even better with time, since it can only become more personalized, as objects and articles are added to repository.

Improvement

Several future improvements that could be mentioned here are:

- Add more features such as velocity, distance
- Add information about size in degrees of object
- Label data by building a feedback loop on those articles that get selected; best place would be directly output in CSV then by manually marking those papers that one has read
- Remove recommended articles from repository, if one already knows them
- Perform better replacement of missing values in flux series. This can mean to use templates.
- Remove papers that only present survey data
- Improve voting model by weighting of both results

The best place to start would be add new features, this would improve the results especially for those objects downloaded from Simbad, since some of them might only have just a redshift value and a type (when spectroscopic data is missing). The second step could be to build in the feedback loop by reading those manually processed CSVs and read out what was actually considered „interesting“ and what not. This would then help the model to better calibrate the final results.

A third step would be to improve on the scoring model by finding optimal weights on both classifier MLP and SVD matrix. Both models have their strengths and weaknesses, which were discussed, so one could balance the effects, for instance, by strengthening the MLP score while reducing the influence of the SVD score. Such improvements can make the current model even more personalized in its recommendations.

Bibliography

- Centre de Données astronomiques de Strasbourg (CDS, Simbad, VizieR): http://cdsportal.u-strasbg.fr/
- astrophysics data system (ADS): https://ui.adsabs.harvard.edu
- NASA/IPAC Extragalactic Database (NED): http://ned.ipac.caltech.edu
- Sloan Digital Sky Survey (SDSS): https://www.sdss.org/
- Efficient Photometric Selection of Quasars from the Sloan Digital Sky Survey: II. (Richards et al. 2008)
- Spectroscopy of red Quasars (Suk Joo Ko, 2019)
- Photometric System (Wikipedia)
- A Comparison of Six Photometric Redshift Methods Applied to 1.5 Million Luminous Red Galaxies (Abdalla et al. 2008)