**Description of the ML project for the John Bryce course 7718-5**

“*Prediction of QSO vs. others using the SDSS database*”

By Oved Dahari, 5-Jan-2021

Summary

Machine Learning is used to predict the identification of Quasi-Stellar Objects in the general catalogue of the Sloan Digital Sky Survey. A few ML classification models were tested on ~150k classified samples, and the best model proved to be Random Forest, with parameters for best results obtained by using a grid-search cross-validation technic. Three resulting models with different class-weights values are provided to the user, for choice of best precision, recall or accuracy. The model of choice is used to identify QSO candidates from any region in the sky chosen by the user, and the resulting predictions are written to disk.

The Scientific Background

The celestial objects called Quasi Stellar Objects (QSOs) were discovered the 1960’s, and remain one of the mysterious astronomical phenomena yet to be fully explained. See [www.en.wikipedia.org/wiki/Quasar](http://www.en.wikipedia.org/wiki/Quasar). After the discovery of the first QSO named 3C 273 (called Quasar at the time, for Quasi Stellar Radio object), hundreds more were found in various searches. However, a huge leap forward was made when *millions* of QSO’s were discovered by the Apache Point Observatory in New Mexico, as part of the Sloan Digital Sky Survey (SDSS) project. See [www.sdss.org](http://www.sdss.org).

The reason it took so long to discover the *first* QSO is its distance – some 2.4 billion light years away. It remains the closest QSO to Earth, although similar (weaker/smaller) objects were found in the nuclei of closer galaxies. Hence these objects are rare in the local universe, but were more common in the past. In fact, most of them are so luminous, that they outshine most galaxies, and at the far reaches of the universe they become the majority of *observed* objects (where telescopes detect just the most luminous objects).

The Sloan Digital Sky Survey (SDSS).

The SDSS survey consists of wide-field imaging of the night sky in 5 filters or ‘colors’ (each field is photographed 5 times), see Appendix A. Subsequently, a few hundred objects are selected in each field for follow-up spectroscopic observation. The spectrum is required for further scientific analysis. It provides the redshift of the objects, and enables their classification. As of 2018 when the latest data release (DR16) was published, approximately 35% of the entire celestial sphere was covered. Some areas of the sky are blocked by the Milky Way, and others (southern) areas are not accessible from New Mexico.

The nature of the survey provides spectra for only a small portion (about 5 percent) of the objects which were imaged. Therefore, when classification is needed for *all* objects in a specific field, it may be done using their ‘colors’ only – the intensities of the objects as measured in the 5 color images.

The SDSS website provides access to various catalogues. The imaging catalogue is called ‘PhotoObj’, where the 5 color intensities are listed for each object. As of today (Data Release 16), some 1 billion(!) objects are listed in that catalogue. Another catalogue – ‘SpecPhoto’ - lists all objects which have reliable spectra (produced in the follow-up spectral study), and hence it contains classification to three classes: stars, galaxies, and QSOs.

Studying the two catalogues, I have found another classification that appear in both: the ‘type’ of the object, depending on its light distribution in the image. As it turns out, all objects in the SpecPhoto catalogues are either type 3 (extended) or 6 (like a point source). Since most QSOs at high redshift are expected to look like a point source (hence the original name), the ‘type’ may prove to be a good feature separating them from galaxies (but not from stars).

I have also found, that the PhotoObj catalogue contains repetitions, due to overlap of sky coverage during the observation process. However, using a parameter called ‘mode = 1’, I selected only ‘Primary’ targets. Since in the SpecPhoto catalogue the targets are all Primary (mode = 1), this practice avoids duplicates and eliminate potential skews in predicting QSO-candidates.

In summary, my software uses the SpecPhoto catalogue for training and testing the model, and the PhotoObj catalogue for a search QSO candidates in the field selected by the user.

Feature Engineering

The data in the catalogues contain astronomical quantities called ‘apparent magnitude’. These are relative unitless numbers where the difference between two magnitudes and is given by the formula:

where *I* is the intensity of the object. The formula shows that if an object 1 is 100 times more luminous that object 2, its magnitude is 5 units *less* that the magnitude of object 2 (for historical reasons).

Like all astronomical images, the intensities are calibrated by comparison to “standard” stars.

The observed (or apparent) magnitude of an object is different than its real (or absolute) magnitude by two factors: (a) its distance from Earth, since the intensity is proportional to the inverse of the distance squared; and (b) the “reddening” of the colors by interstellar absorption, or “extinction”. See <https://en.wikipedia.org/wiki/Extinction_(astronomy)>. The extinction is *calculated* by the galactic coordinates of the object, using a model of the galactic interstellar medium (which causes the absorption). In the SDSS catalogues the extinction-correction values are given for each filter, so they can be applied to each color magnitude directly, by subtraction.

As for the distance to the object, it cannot be determined by imaging alone, and hence the PhotoObj catalogue does not contain it. Therefore, comparing absolute (real) magnitudes of two objects is not possible, and only *relative* colors can be used to compare quantities between objects. As we see from the formula above, the *difference* between the magnitudes (called hereafter ‘color index’) does not depend on the distance to the objects.

As we have 5 colors in the tables (*u, g, r, i*, and *z*), we can construct 10 combinations of color indices. For example, ‘u-g’ is defined as , where the magnitudes in the *u* and *g* filters were first corrected for extinction. Adding the ‘type’ categorial feature (see above), we get 11 features for training the models.

Further Considerations

The spectra of QSOs are markedly different from the spectra of galaxies (see the Appendix). They contain broad emission lines, and an underline continuum which extends far into the UV. While the emission lines do not significantly influence the intensity observed with broad-band filters, the excess in the UV may provide a strong difference in the ‘u-g’ color index. However, the *apparent* (observed) spectra of QSOs are strongly skewed by their significant redshift. For example, for QSOs (or any other object) with a redshift of 2, the ‘u’ band is shifted into the ‘g’ band(!). Additionally, as seen in the Appendix, beyond (shorter of) the Hydrogen Lyα emission line at 1215 Å, there is strong absorption in the spectrum, caused by the Lyα absorption by the material between Earth and the object. Therefore, we expect significant reduction in the ‘u’ and ‘v’ magnitudes for objects with redshifts larger than 2.5 or so, where Lyα is redshifted into the ‘u’ or even the ‘g’ band. Hence when a mix of QSOs at different redshifts are examined together, the difference of their colors from other objects (galaxies, stars) may not be so obvious.

The spectrum of galaxies in more uniform, in general (see Appendix A), although elliptical galaxies are redder than spirals.

The spectra of individual stars (which are always present in any direction in the sky) depend strongly on their surface temperature, and hence span a wide range of colors.

All considered, the classification based on color indexes (and ‘type’) is complicated, and therefore calls for machine learning technics.

Retrieving the data from SDSS

As is customary with American national scientific data, it becomes public (world-wide) after a few years of the observations. The same is true for the SDSS survey.

When the SDSS data base became large, the management of the project decided to put it in a ‘cloud’, providing the users with Jupyter Notebooks to access the data via the SkyServer website. See <http://skyserver.sdss.org/dr16/en/home.aspx>. It uses a Python library called SciServer.py.

In this (John Bryce) project, the requirement is a stand-alone Python program that can retrieve the data directly. Therefore, I tried to install the SciServer library on my computer (either by the ‘pip’ tool, or by PyCharm) without success. The library could not be found. After corresponding with the SDSS people directly (via e-mail), they directed me to the GitHub website, where I could find the SciServer sub-programs, four of which proved necessary for data retrieval. These four programs are included in my project’s package. To access the cloud, the data-retrieval function uses my own current username and password. However, in case these will not be valid in the future, the program asks the user to register with the SkyServer site, and then provide his username and password to proceed.

Once my Python program is logged-in to the cloud, it uses SQL queries to retrieve the data from the SpecPhoto or the PhotoObj tables, as needed.

The training set

As explained above, the training set it taken from the SDSS SpecPhoto catalogue. It turns out that the catalogue contains ~300 objects per square degree of the sky. I have decided that a ~150k sample should provide a reliable training set, and hence I needed a region (solid angle) of ~350 square degrees of the sky. To allow for good sky overage I have selected 8 non-overlapping regions of ~7x7 square degrees each, differing by their galactic latitudes (20-90, 10 degrees apart).

See the description of equatorial and galactic coordinates here: [www.en.wikipedia.org/wiki/Celestial\_coordinate\_system](http://www.en.wikipedia.org/wiki/Celestial_coordinate_system)

The SQL query returned ~150k objects, of which ~32k (~20%) are classified as QSOs. Some 2350 of the QSOs were classified as ‘extended’ (type = 3), all of which have redshift below 2 (and hence are *relatively* close to Earth).

Examining the training features

Using various Pandas and Seaborn tools like pairplot and histplot, no obvious separation of QSOs vs. all others was found among the 11 features. Also, no obvious correlation was found for any color index in the pairplot diagrams.

To get a quantitative measure of the separation qualities of the color indices (features), I have used the following formula for each color index:

Where *std* is the standard deviation, and *f* is the “separation factor” of the data. When *f* is close to zero, no separation is apparent, and when it is larger than 1, the separation is considered good.

As it turns out, the *f* value for all 10 indices ranged from 0.3 to 0.9, and none of them is close to zero. Therefore, I decided to use *all* 10 features in the training of the models, in addition to the ‘type’ categorical feature (see above). Interestingly (as per the discussion above), the UV filter (‘u’) does *not* provide a strong color index separator. For further analysis, I have examined the attribute ‘feature\_importances’ of the Random Forest model. Indeed, the most important features in that model are ‘u-r’ and ‘u-i’, with ‘u-z’ in third place. Apparently, the Lyα absorption for high-redshift QSOs offsets the UV-excess of low-redshift ones to some extent, but not completely.

Running various ML models

This is a classification problem, with 10 continuous (real numbers) features, and one categorial (two classes) feature. In all, I have tested six different ML models: kNN, Logistic Regression, SVM, Decision Tree, Naïve-Bayes-Gaussian, and Random Forest. All models were trained on the whole 150k sample, using the GridSearchCV method (see sklearn.model\_selection) to find the best parameters for each model. The SVM model was found to be too slow for the grid search, so it was compared to the Random Forest model on a smaller data set, and found to be inferior.

The results are given in the following table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Param1** | **Param2** | **Param 3** | **Best Score** |
| kNN | n\_neghbors = 30 | weights = distance | Leaf\_size = 10 | 0.9306 |
| LR | C = 0.001 | -- | -- | 0.9018 |
| GNB | -- | -- | -- | 0.8388 |
| DT | min\_sam\_leaf = 2 | max\_depth = 5 | Criterion = gini | 0.9232 |
| RF | n\_est = 80 | min\_leaf = 10 | Max\_depth = 150 | 0.9325 |

It turns out that the Random Forest model provides the best result.

For further possible improvement, I’ve tested the Random Forest model with *clustering* the features first, and using an RF model on each cluster separately. No improvement was achieved with 3-10 clusters. Also, I tried using the three original classes (‘GALAXY’, ‘QSO’, and ‘STAR’), with no significant change in the results for the QSO class, when compared to the 2-class model (‘QSO’ vs. ‘Others’). Also, I tried the PCA method to reduce the number of the color-index features from 10 to 4. While indeed the 4 PCA features covered >99% of the variance, they provided no improvement in predicting QSOs on the test data (see below).

Once the model was chosen (Random Forest), and the parameters selected, it was trained on the whole 150k-samples dataset. Additionally, I’ve decided to provide the user with 3 options: high precision, best result, and high recall. This was achieved by use of the parameter ‘weights’ of the model. Hence the program will produce (and use) 3 models, which are saved on disk.

The resulting models were tested on new data from the SpecPhoto catalogue. It contains five 7x7-degrees regions which do not overlap with the training dataset. The results are given in the following table:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Region ra** | **High Precision Model** |  | **Best accuracy**  **Model** |  | **High Recall**  **Model** |  |
|  | Precision | Recall | Precision | Recall | Precision | Recall |
| 190 | 0.91 | 0.53 | 0.79 | 0.80 | 0.69 | 0.93 |
| 175 | 0.91 | 0.45 | 0.81 | 0.74 | 0.72 | 0.91 |
| 160 | 0.90 | 0.45 | 0.79 | 0.75 | 0.67 | 0.90 |
| 140 | 0.92 | 0.53 | 0.84 | 0.73 | 0.68 | 0.82 |
| 120 | 0.93 | 0.64 | 0.84 | 0.86 | 0.71 | 0.94 |
| Average | 0.91 | 0.52 | **0.81** | **0.77** | 0.69 | 0.90 |
|  |  |  |  |  |  |  |
| All regions | 0.92 | 0.54 | **0.82** | **0.78** | 0.69 | 0.89 |

Hence the values in the last row were used to inform the user about the precision/recall of the model of choice (with about 2-3% uncertainty).

Although the GridSerachCV algorithm provided the best parameters, I have found that smaller values used with the Random Forest model provided the same results, while reducing the training time and the size of the models when saved on disk. Therefore, I used n\_estimators = 30, and max\_depth = 30.

Finally, the models are used to search for QSO candidates from the PhotoObj catalogue, in a region selected by the user. I include the value ‘predict\_proba’ in the resulting list of QSO candidates. This will help the user in designing his next line of research, i.e. to start with candidates with the highest probability of being QSOs.

A word of caution: The selection of objects by the SDSS team for follow-up spectral observations induce *selections effects* which may influence the actual use of the ML models for prediction of QSO objects among the general (PhotoObj) field.

Running the Python program

The details of running the program are listed in the sdss\_readme.txt file.

At first, if the models do not exist on disk, the program downloads the training data (if not on disk), and then ‘train’ the three models with parameters as indicated above. The downloaded data and the models are written to the user’s disk.

Then the program enables the user to test the models on classified data (from the SpecPhoto table), from a region in the sky selected by him.

Then for searching QSO candidates, the user selects the region in the sky and the model (see the 3 options described above). The selected model is applied to the data, and the list of suspected QSOs is created. The list is sorted by right ascension and declination, and is saved to a .csv file (its name is chosen by the user). It is designed for the scientist who wants to use the data to search for QSOs. See the sdss\_readme.txt file.

Comparing with other studies.

In general, the use of ML for astronomical research is relatively new. In 2011 a new society was founded, called IAIA or International Astronomical Informatics Association, by my colleague Prof. George Djorgovski from UCLA, among others. See [www.astroinformatics.info/astroinfo](http://www.astroinformatics.info/astroinfo).

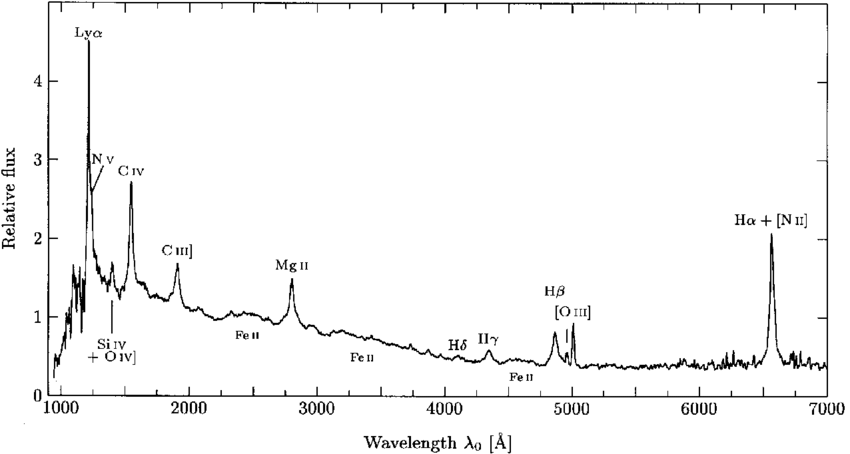
I have found only one ML paper which is specifically similar to my project, by V. Acharya et al, 2018 (*Current Science*, Vol. 115, No. 2). In that article, the authors used cloud services to train their models (on ~100k objects), and to apply the results to the whole data archive (some 1 billion samples). They used only 4 color indices (and not 10), and did not use the ‘type’ information that I used. They tested 3 different models: kNN, SVM, and Random Forest. Their results are somewhat inferior to mine: in their best (Random Forest) model, they achieve precision of 80%, with recall of 72%.

Another study was published by Bai et.al 2019 (*The Astronomical Journal*, 157: 9). However, in this study the data is a combination of the SDSS and the Chinese LAMOST project (see [www.lamost.org](http://www.lamost.org)). They arrived at better prediction results, but direct comparison with my project is not possible.

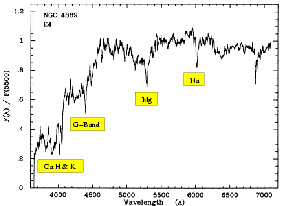
Appendix A: Examples of the spectra

The spectra of a typical QSO (in the rest frame), elliptical and spiral galaxies are shown below. Spectra of stars differ widely (mostly due to their surface temperature), and hence there is no point in displaying them in this Appendix.

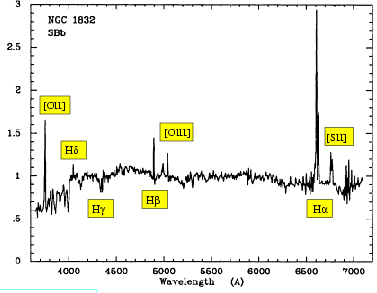
The Appendix also include the ‘spectrum’ of the 5 filters, to demonstrate their specific spectral transmission.



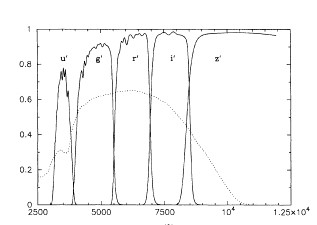
A spectrum of a typical QSO in the rest frame



A spectrum of a typical elliptical galaxy



Spectrum of a typical spiral galaxy



The SDSS filters