Astronomical Point Source Classification through Machine Learning

Idel R. Waisberg

December 13, 2013

Abstract

We investigate the performance of a variety of classification algorithms to astronomical point source classification. Specifically, to classify objects from the SDSS survey into main sequence/red giant stars, white dwarfs or quasars using photometric data across 5 bands. We found that k-nearest neighbors and a multinomial SVM using a Gaussian kernel were the algorithms with the best performance, and dependence on training size data and feature importance were further investigated.

Introduction and Objective

With the Sloan Digital Sky Survey (SDSS), which provides a catalog with around 230 million objects in the visible band and covering 1/4 of the celestial sphere, the amount of data available for astronomers has become, if one can dare say, astronomical. A particular problem is the classification of point-like sources, due to the lack of visible structure of these objects. These include different types of stars (main sequence, red giants, white dwarfs), but also more complex structures (such as quasars - supermassive black holes surrounded by an accretion disk).

The most accurate way to classify point-like sources is through high-resolution spectroscopy i.e. by analyzing atomic transition lines. However, this somewhat complex process can easily take a large amount of time given the great number of sources provided by the survey. An interesting question is whether an accurate classification of point-like sources can be made through lower resolution photometry i.e. measurement of light intensities for a few different bands that each span a large wavelength range. This is a much less complex and robust measurement than spectroscopy, and could therefore provide a desired alternative to the latter. [3,4]

The aim of this project is to investigate different machine learning algorithms to classify astronomical point-like sources into 3 possible classes (quasars, main sequence/ red giant stars and white dwarfs) by using photometric data in 5 different bands (u (ultraviolet), g (green), r (red), i and z (very-near infrared)), available from the SDSS survey.

Data Structure and Features Selection

Both training and test data were obtained from Data Release #7 of the SDSS survey. [1,2]. Test data consists of 2500 quasars (qsos), 4000 main sequence and red giant stars (m/r stars) and 981 white dwarfs (wds) obtained from a region spanning $\pm 25^\circ$ in declination and $\pm 1h$ in right ascension from the author's Zodiac sign Virgo constellation ($(RA, dec) = (13h, -4^\circ)$). The training data consists of 3000 qsos, 7000 m/r stars and 3000 wds from sky regions excluding the test data range. Also, only sources for which $15 < mag < 21$ and for which the uncertainty in the measurement is $< 0.2mag$ in all the photometric bands considered are obtained, in order to exclude too bright or too faint sources, and to ensure that the measurement is reasonably accurate. The classes of all the objects in the test and training data have been previosly confirmed by spectroscopy.

Because the brightness of an object varies with the distance from Earth, the magnitudes (roughly the log of luminosity) provided by the photometry measurements are not themselves good indicators of the class of a point-like source. Rather, it is the ratio of luminosities (difference of magnitudes) – called color indices – between adjacent photometric bands that matter, since they indicate how one "color" predominates over the other. [5] In this case, we are provided with 5 photometry measurements in the near-visible range; hence, there are 4 color indices (u-g, g-r, r-i, i-z) that can be constructed. The most straightforward input vector of features in this case is therefore 4-dimensional. Figure 1 shows the three bivariate color-color plots for the test data, using the known classification solution.

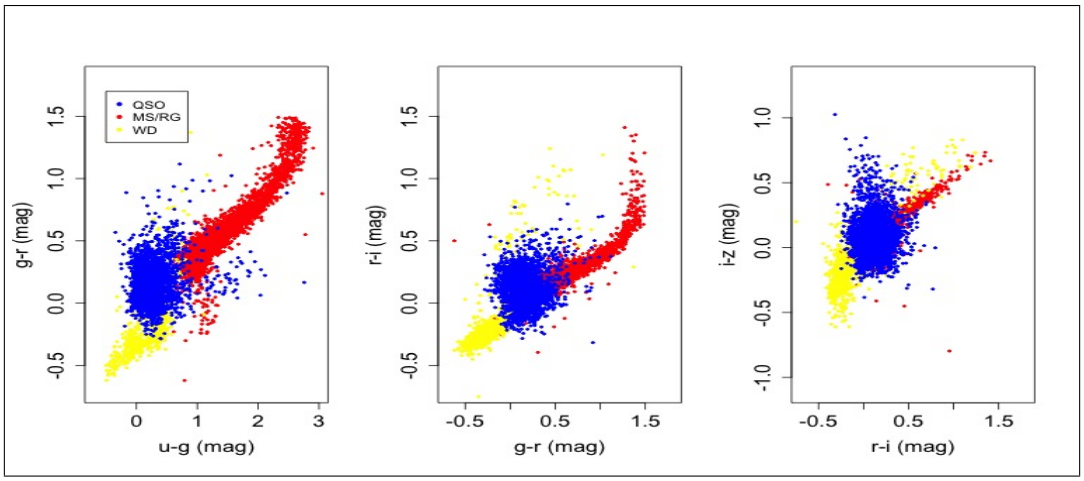

Figure 1: Test Data represented in bivariate color-color diagrams.

Classification Algorithms - Results and Analysis

The aim of the machine learning classification algorithms is to explore complex or unknown relations between features and labels to classify new data, without a complete understanding of the physical laws that might deterministically connect them. Below, we discuss different classification algorithms and investigate their classfication performance.

Unsupervised Learning: K-means clustering

We start with one of the simplest models for classification: k-means clustering, with 3 centroids in this 4-dimensional space. The result, shown in Figure 2, shows that this is a poor classification algorithm (accuracy $\approx 25.26\%$); for instance, it divides the main sequence structure in the leftmost bivariate plot into the 2 classes and is unable to distinguish between qsos and wds.

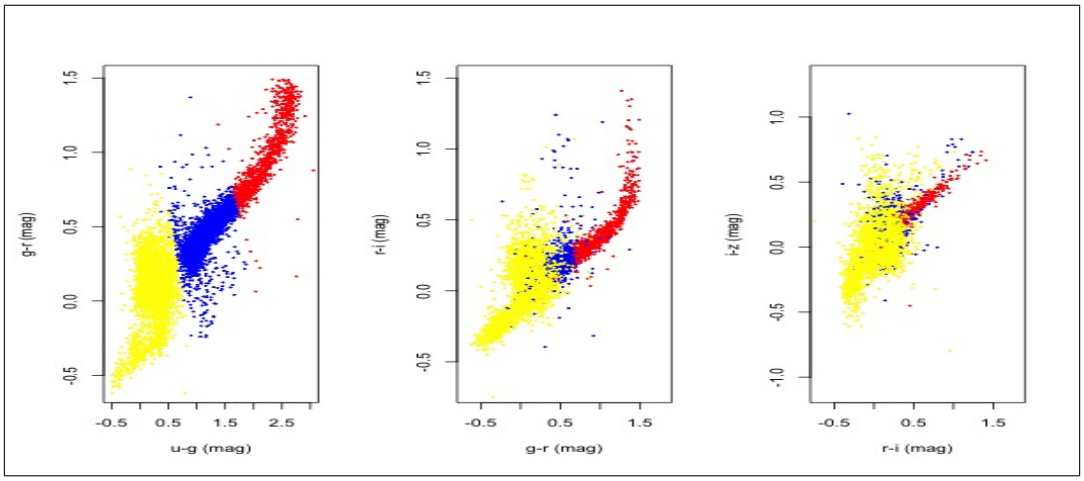

Figure 2: Bivariate color-color plots for test sample resulting from k-means clustering with 3 centroids.

Given it is an unsupervised learning algorithm, kmeans can only take into account the spatial distribution of the data, and ignores the different relations between features that vary depending on the label. Supervised learning tries to explore such relations through the training examples. Figure 3 below is a parallel coordinates plot for a section of the training data (40 randomly selected objects of each class), and clearly shows different trends in feature relations depending on the class label (e.g. $g - r$ and $r - i$ tend to

be directly proportional for m/r stars and inversely proportional for qsos). Therefore, it is expected that supervised learning algorithms would have a much better classification performance.

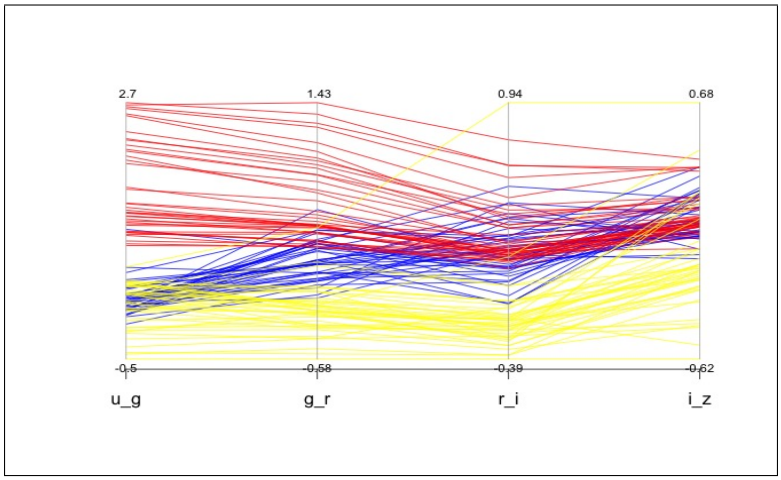

Figure 3: Parallel coordinates plot for training data.

Supervised Learning: MLR, GDA, knn and SVM

Here, we investigate the performance of supervised learning algorithms for classification: Multinomial Logistic Regression (MLR), Gaussian Discriminant Analysis (GDA), K-nearest neighbors (knn) and Support Vector Machines (SVM).

GDA was applied assuming the 3 covariance matrices are the same (linear decision boundaries). Knn was applied with k=4 neighbors. SVM was applied with a cost $C = 100$ and three types of kernels: Gaussian ($K(x, z) = \exp^{||x-z||^2}$), Linear (($K(x, z) = x^T z$) and a second degree polynomial ($K(x, z) = (x^T z)^2$). In this case, because there are three classes, a multinomial SVM algorithm is needed. This algorithm is based on the regularized SVM algorithm presented in [6] using a one-versus-one approach i.e. given k classes, $C(k, 2)$ binary classifiers are constructed (3 if $k = 3$) and the class selected by the most classifiers is chosen.

Table 1 shows the performance of each model on both the training and test data.

Table 1: Supervised learning algorithms' accuracies (%) on training (tr) and test (ts) data.

-	ta (m/r stars)	ta (qsos)	ta (wds)	ta (overall)	ts (m/r stass)	ts (qsos)	ts (wds)	ts (overall)
MLR	99.93	85.5	83.4	92.78	97.58	81.56	82.87	90.29
GDA	98.50	85.13	86.57	93.36	96.90	81.28	87.36	94.49
Knn	99.9	97.07	97.37	97.67	97.33	94.64	97.25	96.42
SVM (Gaussian)	99.99	98.5	96.97	98.95	95.48	96.72	96.64	96.04
SVM (Linear)	99.91	93.70	82.03	94.35	97.18	90.84	81.85	93.02
SVM (Polyn.)	96.13	86.63	77.80	89.71	93.73	80.32	78.59	87.26

The table shows that all the algorithms achieved a good performance in the classification of m/r stars. Overall, the accuracy in classifying qsos and wds was much lower; this is expected given the considerable overlap between these two classes in the bivariate color-color diagrams. Indeed, the confusion matrix for the algorithms show a high degree of confusion between these two classes.

The two best overall performances were given by Knn and SVM (Gaussian kernel) algorithms. At first, this might seem curious given the simplicity of the former algorithm and the complexity of the latter. However, this might be expected, since the models constructed by both algorithms are heavily determined by the Euclidean distances between the training examples. Changing the cost value C of the SVM algorithm and increasing the number of neighbors k of the Knn algorithm did not alter the performance significantly.

Table 2 shows the confusion matrix for Knn and SVM (Gaussian kernel) when the model is applied to

Table 2: Confusion Matrices for Training (ta) and Test (ts) data for Knn and SVM (Gaussian Kernel). Lines refer to predicted classification, whereas columns refer to actual (real) classes.

Knn, ta	m/r star	qso	wd
m/r star	6993	28	21
qso	3	2912	58
wd	4	60	2921

Knn, ts	m/r star	qso	wd
m/r star	3893	25	3
qso	60	2366	24
wd	47	109	954

SVM (Gaussian), ta	m/r star	qso	wd
m/r star	6999	12	21
qso	0	2955	70
wd	1	33	2909

SVM (Gaussian), ts	m/r star	qso	wd
m/r star	3819	13	3
qso	126	2418	30
wd	55	69	948

both the training and test data. The two algorithms performed very similar when the model was applied to the training data; comparing the performance on the test data, they perform quite similarly for the classication of wds, but the SVM (Gaussian kernel) performs superiorly for the classification of qsos and inferiorly for the classification of m/r stars.

Further Investigations

Cross-validation studies were done to further assess the performance of Knn and SVM (Gaussian kernel). A leave-one-out cross validation on the training set for Knn resulted in an accuracy of 98.1%. Given the higher complexity of the SVM algorithm, a 10-fold cross validation was performed in this case, resulting in an accuracy of 98.24%. The SVM (Gaussian kernel) was therefore chosen to be used for further investigations of the data. Figure 4 shows the bivariate color-color plots of the classification result of the SVM (Gaussian kernel) on the test set. It is very close to the true classification shown in Figure 1.

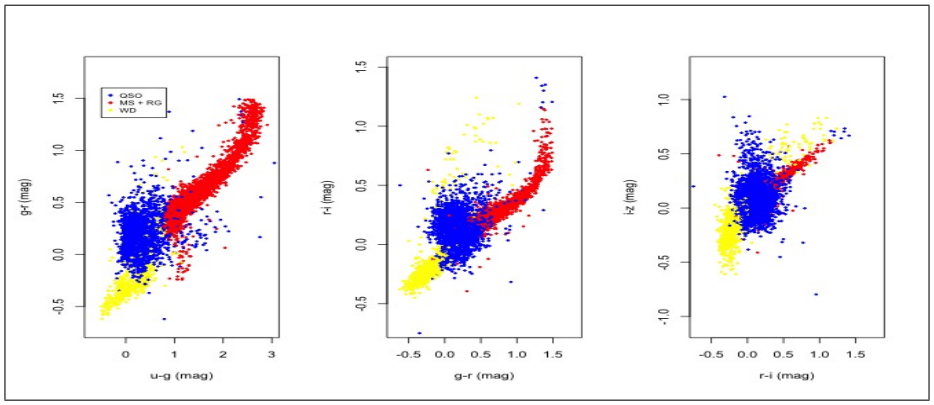

Figure 4: Classification result of SVM (Gaussian kernel) on the test set.

Figure 5 shows how the training and test errors change as a function of the training size (m) when SVM (Gaussian kernel) is applied. The training error increases slightly as m is increased, whereas the test error decreases until approximately $m = 20000$ and then further increases. The large gap between training and test errors suggests the algorithm does not suffer from high bias, so that an increase in the number of features is not likely to increase the performance. The fact that the test error increases with m soon after the operating point suggests that the algorithm does not suffer significantly from high variance, and therefore decreasing the number of features is also not likely to increase the performance of the algorithm. Therefore, we conclude we are close to the optimal performance of the algorithm at the operating point, given the information and physical knowledge available.

Although machine learning algorithms can function quite well without the knowledge of the underlying mechanisms or physical laws concerning the data, their results can provide insight into the data that might initially not have been obvious. Table 3 shows the performance of the SVM (Gaussian Kernel) algo-

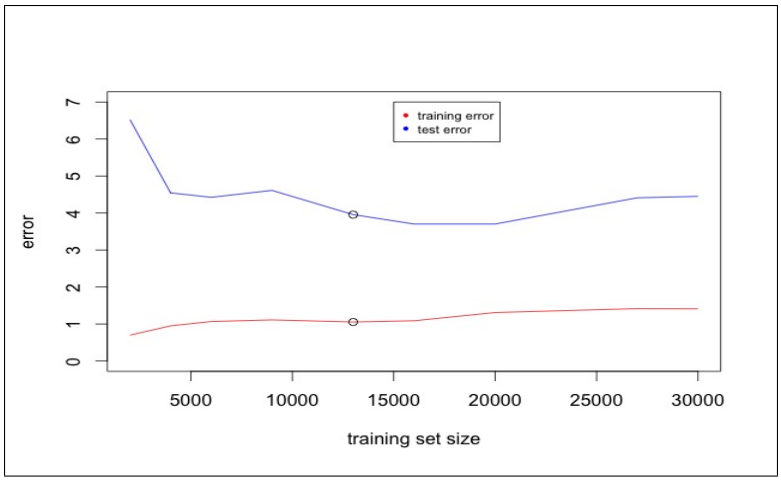

Figure 5: Training and test error dependencies on the training data size for SVM (Gaussian kernel) algorithm. The black dots refer to the point where the previous results in this paper were obtained.

Table 3: Left: performance of SVM (Gaussian kernel) given each feature used alone. Right: ablasive analysis of the same algorithm and features.

-	training accuracy	test accuracy
$u-g$	83.02	82.70
$g-r$	84.27	79.44
$r-i$	77.54	69.44
$i-z$	72.79	64.90

-	training accuracy	test accuracy
add $i-z$	98.95	96.04
add $r-i$	96.93	94.32
add $g-r$	92.77	91.02
$u-g$ (baseline)	83.02	82.70

rithm using only one of the 4 features for classification, whereas Table 5 shows the result of ablasive analysis performed on this algorithm (i.e. the features are added one by one from bottom to top and the increase in the performance assessed). From the tables, it is possible to see that $g-r$, $u-g$, $r-i$ and $i-z$ are, in decreasing order, the most important features for the classification of astronomical point sources in main sequence/red giant stars, white dwarfs and quasars, which might provide previously unavailable insights into the different physical structures and characteristics of these astronomical bodies.

Further Work

There are several possibilites for future work in this topic, among which:
- Use more complex classification algorithms such as neural networks and classification trees;
- Include more object types in the dataset: more high-redshift quasars, brown dwarfs (they appear in redder bands of the SDSS survey), and spatially extended galaxies;
- Use physical information to add additional features and/or weights and prior probabilites to the algorithms, and check if performance is improved.

References

[1] http://cas.sdss.org/astro/en/tools/search/sql.asp
[2] http://vizier.u-strasbg.fr (Vizier Catalogue, SDSS-DR7 quasar catalog (Schneider+, 2010) and SDSS DR7 white dwarf catalog (Kleinman+, 2013))
[3] Feigelson, Eric D. *Modern Statistical Methods for Astronomy With R Applications* Cambridge, 2012
[4] *Advances in Machine Learning and Data Mining for Astronomy* Chapman & Hall
[5] Phillips, A.C. *Physics of Stars* Wiley, 2010
[6] Ng, A. *CS 229 lecture notes. Part V - Support Vector Machines*