Identifying Quasars in Large-Scale Spectroscopic Surveys

technische universität dortmund

Fabian Gieseke[1] fabian.gieseke@tu-dortmund.de
Kai Lars Polsterer[2] polsterer@astro.ruhr-uni-bochum.de
Andreas Thom[1] andreas.thom@tu-dortmund.de

Peter Zinn[2] **Dominik Bomans**[2] **Ralf-Jürgen Dettmar**[2] **Oliver Kramer**[1] **Jan Vahrenhold**[1]

[1]Chair of Algorithm Engineering, Faculty of Computer Science, Technische Universität Dortmund, Germany [2]Astronomisches Institut, Department of Physics and Astronomy, Ruhr-University of Bochum, Germany

RUHR UNIVERSITÄT BOCHUM RUB

Abstract

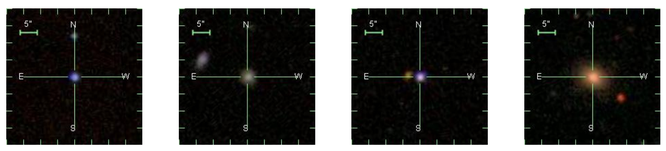
We present automated classification approaches to discriminate quasars from other objects based on spectroscopic data. The performance of our approaches is evaluated on a manually labeled training set obtained from the spectroscopic information available in the Sloan Digital Sky Survey. Our experiments indicate that feature extraction and meaningful peak detection approaches significantly improve the classification performance.

Motivation

Classification Task: Identifying Quasars

The (semi-)automatic analysis of data sets has become an increasingly important issue for researchers in Astronomy [1, 2]. This is especially true for massive data sets obtained from, e.g., the Sloan Digital Sky Survey [3] which is based on raw data of about 60 TB. From a Machine Learning perspective, a variety of problems in Astronomy can be formulated as supervised (e.g. classification, regression) or unsupervised tasks (e.g. clustering, dimensionality reduction). We describe the use of supervised learning techniques to identify quasi-stellar radio sources (quasars) based on spectroscopic data.

Quasars



Source: `http://www.sdss.org`

Data

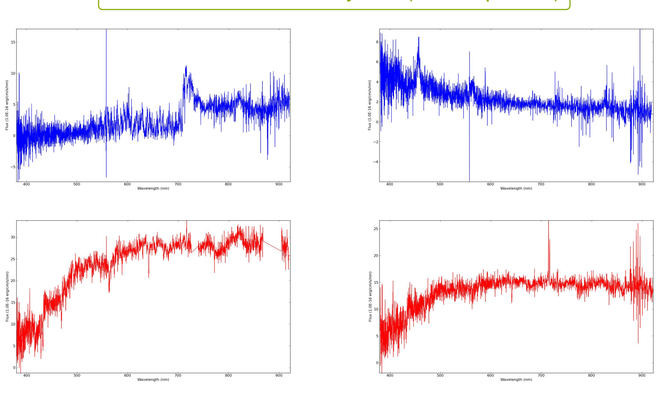
Our data set is obtained from spectroscopic data available in the Sloan Digital Sky Survey (DR6) which is said to be "one of the most ambitious and influential surveys in the history of astronomy" [3]. The data for this survey has been obtained via a 2.5-meter telescope at the Apache Point Observatory which is equipped with two special-purpose instruments: a 120-megapixel camera and a pair of spectrographs.

Apache Point Observatory (New Mexico)

Source: `http://www.sdss.org`

We use a manually labeled data set consisting of $n = 3,351$ raw spectra (319 "quasars" and $3,032$ "other objects"). Each spectrum is represented by a flux value (intensity) for each of the $d = 3,854$ wavelengths.

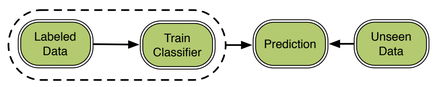
Quasars vs. Other Objects (Raw Spectra)



Classification Task

The result of our manual labeling step is a (training) set T that consists of spectra labeled with "quasar" $(+1)$ or "other object" (-1), i.e., we have $T = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \subset \mathbb{R}^d \times \{-1, +1\}$. The main objective then is to determine a classifier with good classification performance for "unseen" spectra.

Classification Chain



Classification Approaches

k-Nearest Neighbors: The k-Nearest Neighbor classifier uses the k "closest" objects from the given set of classified objects to assign a class to an unclassified object [4]. More precisely, the (binary) classification $\hat{Y}(\mathbf{x})$ for an object $\mathbf{x}$ is

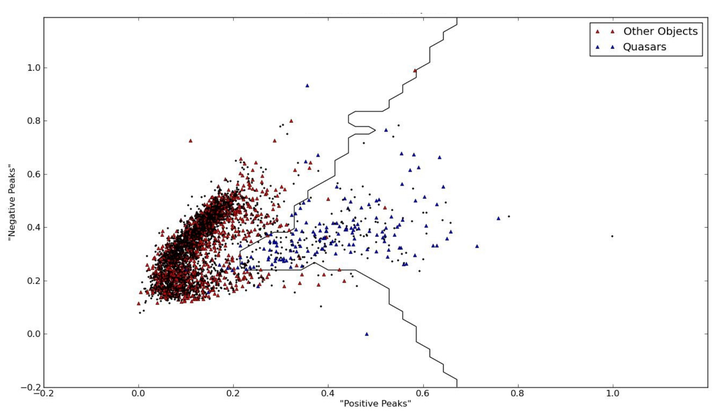
$$\hat{Y}(\mathbf{x}) = \begin{cases} 1 & \text{if } f(\mathbf{x}) > 0 \\ -1 & \text{if } f(\mathbf{x}) \leq 0, \end{cases} \tag{1}$$

where

$$f(\mathbf{x}) = \frac{1}{k} \sum_{\mathbf{x}_i \in N_k(\mathbf{x})} y_i \tag{2}$$

and where $N_k(\mathbf{x})$ denotes the k-nearest neighbors in the training set with respect to $\mathbf{x}$. To define "closeness", arbitrary metrics can be used; a popular choice is the Euclidean metric.

k-Nearest Neighbors



Support Vector Machines: Roughly speaking, the aim of a Support Vector Machine (SVM) is to find a hyperplane in a feature space which maximizes the "margin" between both classes such that only few training patterns lie within the margin. The latter task can be formulated as an quadratic optimization problem, where the first term corresponds to maximizing the margin and the second term to the loss caused by patterns lying within the margin:

$$\begin{aligned} \underset{\mathbf{w} \in \mathcal{H}_0,\, \xi \in \mathbb{R}^n,\, b \in \mathbb{R}}{\text{minimize}} \quad & \frac{1}{2}||\mathbf{w}||^2 + C \sum_{i=1}^{n} \xi_i \\ \text{s.t. } & y_i(\langle \mathbf{w}, \Phi(\mathbf{x}_i) \rangle + b) \geq 1 - \xi_i, \\ & \text{and } \xi_i \geq 0, \end{aligned} \tag{3}$$

where $C > 0$ is a user-defined parameter. The function $\Phi : \mathbb{R}^d \to \mathcal{H}_0$ is induced by a kernel function $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ with $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$. A kernel function can be seen as a "similarity measure" for input patterns. The goal of the learning process is to find the optimal hyperplane $f(\mathbf{x}) = \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b$. Unseen objects can subsequently be classified via Equation (1). A common choice for the kernel function is the linear kernel

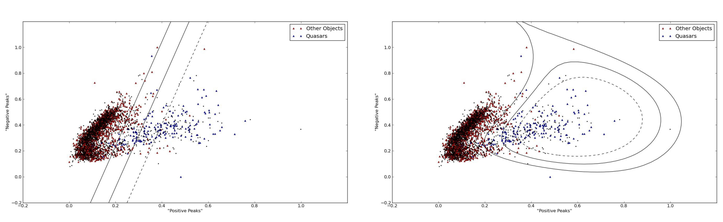
$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle \tag{4}$$

or the RBF kernel

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2} \right) \tag{5}$$

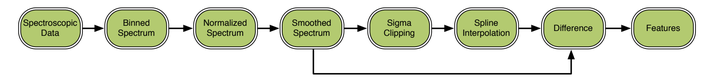
with σ as additional parameter.

Support Vector Machines (Linear and RBF Kernel)



Feature Extraction

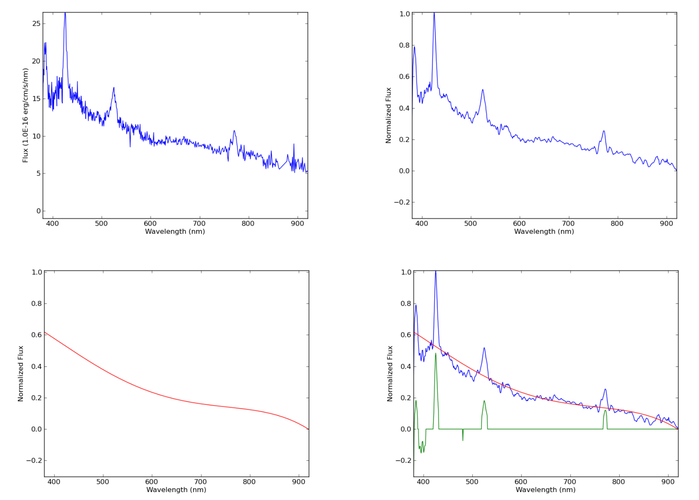
Process Chain



Quasars are distant galaxies with an active galactic nucleus and thus their spectroscopic data exhibits broad emission lines. Before applying our classification approaches, we thus attempt to first extract these meaningful features using the following preprocessing step:

1. Merge consecutive flux values to obtain a "binned" version.
2. Apply a smoothing filter (e.g., the Savitzky-Golay-Filter).
3. Perform a spline interpolation to extract the continuum.
4. Extract peaks based on smoothed spectrum and spline model.

Intermediate Results



The final features can then be created based on the continuum and the extracted peaks (see below).

Experiments

Experimental Setup: To train and evaluate the classification approaches, we use $1,675$ elements of the data as training and the remaining $1,676$ elements as test set. For both approaches, model parameters (k, C, and σ) are determined via a 10-fold cross-validation [4] on the training set. The performances of the resulting models are then evaluated on the test set, where the classification error (i.e., the percentage of misclassified elements) is used as quality measure.

Extracted Features: We evaluate both classification approaches for the following spectrum-to-features reductions:

- **963 Features**: A binned version (factor 4) of the raw spectrum
- **2 Features**: "Integrals" over all positive and over all negative values in final peak curve (Step 4; green curve)
- **6 Features**: "Integrals" over all positive and over all negative values in final peak curve (Step 4; green curve); the difference of these values; shape of the peaks; the first and the last flux value.

Results: The following table shows the classification errors obtained for both the k-Nearest Neighbor and the Support Vector Machine classifier (linear and RBF kernel). In addition to the classification errors, we provide the true positive rate ("quasar hit rate") as well as the false positive rate ("false alarm rate") in brackets (i.e., $(tp/pos, fp/neg)$ where the test set contains $pos = 154$ "quasars" and $neg = 1,522$ "other objects").

Classification Performances

	k-Nearest Neighbors	SVM (Linear)	SVM (RBF)
963 Features	4.5% (71%, 2%)	7.6% (68%, 5%)	5.0% (55%, 1%)
2 Features	3.0% (77%, 1%)	3.3% (74%, 1%)	2.5% (81%, 1%)
6 Features	1.5% (90%, 1%)	1.8% (90%, 1%)	1.5% (90%, 1%)

References

[1] Nicholas M. Ball. *Data Mining and Machine Learning in Astronomy*, 2009, arXiv:0906.2173v1 [astro-ph.IM]

[2] Kirk Borne. *Scientific Data Mining in Astronomy*, 2009, arXiv:0911.0505v1 [astro-ph.IM]

[3] Sloan Digital Sky Survey. `http://www.sdss.org`, June 2010.

[4] Trevor Hastie, Robert Tibshirani und Jerome Friedman. *The Elements of Statistical Learning*, Springer, 2010.

Acknowledgements We thank Anna Amelung for the cartoon.