Volume Title
ASP Conference Series, Vol. **Volume Number**
Author

Photometric Redshift Estimation of Quasars: Local versus Global Regression

Fabian Gieseke[1], Kai Lars Polsterer[2], and Peter-Christian Zinn[2]

[1]*Computational Intelligence, Computer Science Department, University of Oldenburg, Germany*

[2]*Astronomisches Institut, Department of Physics and Astronomy, Ruhr-University of Bochum, Germany*

Abstract. The task of estimating an object's redshift based on photometric data is one of the most important ones in astronomy. This is especially the case for quasars. Common approaches for this regression task are based on nearest neighbor search, template fitting schemes, or combinations of, e.g, clustering and regression techniques. As we show in this work, simple frameworks like k-nearest neighbor regression work extremely well if one considers the overall feature space (containing patterns of all objects with low, middle, and high redshifts). However, such methods naturally fail as soon as only very few or even no training patterns are given in the appropriate region of the feature space. In the literature, a wide range of other regression techniques can be found. Among the most popular ones are regularized regression schemes like ridge regression or support vector regression. In this work, we show that an out-of-the-box application of this type of schemes for the whole feature space is difficult due to the involved computational requirements and the specific properties of the data at hand. However, in contrast to nearest neighbor search schemes, such methods can be employed to extrapolate, i.e, they can be used to predict redshifts for patterns in new, unseen regions of the feature space.

1. Introduction

In this work, we describe the use of standard machine learning regression models in the context of estimating the redshift of *quasi-stellar radio sources (quasars)* based on *photometric data*. The data we use in this work is based on the *Sloan Digital Sky Survey* (SDSS), which is said to be "one of the most ambitious and influential surveys in the history of astronomy'.[1] The corresponding catalog contains photometric data of about one billion objects, whereas spectroscopic data is only available for about two million objects. The key problem of detecting new, unseen quasars is the fact that ground-truth information can only be obtained via time-consuming spectroscopic follow-up observations. Hence, the appropriate candidates for such more detailed observations have to be selected only based on the limited information provided by the photometric data.

Many algorithmic schemes have been proposed in the literature that address this regression task. Among these approaches are, e.g., artificial neural networks (Yèche

[1]`http://www.sdss.org`

et al. 2010), support vector machines Wang et al. (2008), or combinations of standard clustering and regression schemes (Laurino et al. 2011). As we show in this work, simple regression approaches like *k*-nearest neighbor regression (Hastie et al. 2009) yield excellent models for the overall task which involves the analysis of all quasars with low, middle, and high redshifts. Further, taking as many as possible data patterns into account seems to be crucial for this particular task. This renders the direct application of sophisticated schemes like support vector regression difficult due to the involved computational requirements. However, the latter class of schemes can be useful in the context of extrapolation, i.e., in the context of predicting trends for patterns not covered by the training data.

2. Machine Learning

For standard regression problem, one is given a *training set* $T = \{(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)\} \subset \mathbb{R}^d \times \mathbb{R}$ consisting of *patterns* $\vec{x}_i \in \mathbb{R}^d$ with and associated *labels* $y_i \in \mathbb{R}$. The goal of the learning process is to generate a *regression model* that predict reasonable labels to new, unseen patterns that are not contained in the training set (Hastie et al. 2009). We will now briefly sketch two well-known learning schemes in this context.

2.1. *k*-Nearest Neighbor Regression

The *k-nearest neighbor* (kNN) (Hastie et al. 2009) regression model uses the $k \in \mathbb{N}$ closest objects from the given set of objects to assign a label to a new object. More precisely, the regression model $f : \mathbb{R}^d \to \mathbb{R}$ is given by

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

where $N_k(\vec{x})$ denotes the *k*-nearest neighbors of $\vec{x} \in \mathbb{R}^d$ in the training set T. To define closeness, arbitrary metrics can be used. A popular choice is the Euclidean metric.

2.2. Support Vector Regression

Another well-known regression concept are *support vector regression* (SVR) (Hastie et al. 2009; Schölkopf & Smola 2001) models which are of the form

$$\inf_{f \in \mathcal{H}_k,\, b \in \mathbb{R}} \frac{1}{n} \sum_{i=1}^{n} \max \mathcal{L}(y_i, f(\vec{x}_i) + b) + \lambda \| f \|^2_{\mathcal{H}_k}. \tag{2}$$

For SVR, the loss function is given by $\mathcal{L}(y, t) = \max(0, |y - t| - \varepsilon)$ with $\varepsilon \in \mathbb{R}pos$ (the so-called *ε-insensitive loss*). Plugging in other loss functions in the above task leads to various other frameworks; using the square loss $\mathcal{L}(y, t) = (y - t)^2$, for instance, leads to the concept of *ridge regression*. The space $\mathcal{H}_k$ is a *hypothesis space* containing functions of the form

$$f(\cdot) = \sum_{i=1}^{n} \alpha_i k(\cdot, \vec{x}_i) \tag{3}$$

with coefficients $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$ and (positive semidefinite) *kernel function* $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$. The first term of the above objective measures how well the function f can

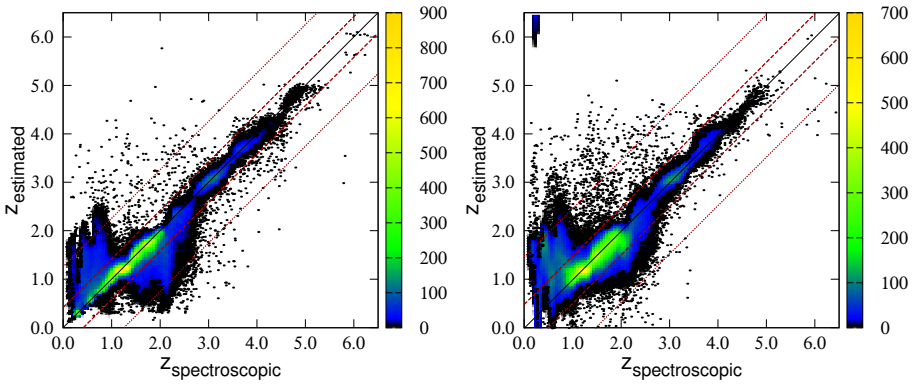

Figure 1. Comparison of the k-nearest neighbor (left) and the support vector regression (right) model. Both models are tested on all quasars given in the data set. Due to computational reasons, the SVR model is only trained on a selected subset of patterns.

predict the (real-valued) labels and the second term measures the complexity of the model. The parameter λ determines the trade-off between both objectives. Common choices for the kernel function are the *linear kernel* $k(\vec{x}_i, \vec{x}_j) = \langle \vec{x}_i, \vec{x}_j \rangle$ or the *RBF kernel* $k(\vec{x}_i, \vec{x}_j) = \exp(-\gamma \|\vec{x}_i - \vec{x}_j\|^2)$ with user-defined parameter $\gamma > 0$.

3. Experimental Evaluation

We will now analyze the performance of both regression models for the task at hand. The labels we use are based on the *SDSS quasar catalog* provided by Schneider et al. (2010). As features, we consider all colors from neighbored bands (`u-g`, `g-r`, `r-i`, `i-z`). This results in a data set consisting of $104,440$ patterns in $\mathbb{R}^4$.

3.1. Standard Redshift Estimation

First, we analysis the performance of both models on the complete data set. Here, the kNN model is based on all patterns, whereas the SVR model (with RBF kernel) is based on a selected subset of the data containing $2,536$ patterns (due to the cubic runtime needed to train a model). The involved parameters are tuned via 5-fold cross validation. Both models are tested on the (remaining) patterns that have not been used for training the models. The result is depicted in Figure 1. It can be seen that the simple kNN model exhibits an excellent overall performance. In contrast, the out-of-the-box application of the SVR model leads to slightly worse results.

3.2. Looking over the Tea Cup's Rim

As shown above, the kNN model is well-suited for densely populated regions of the feature space. However, it cannot predict any *trends* for unseen data. In contrast, SVR has the potential to provide trends. To investigate this issue, we consider only quasars with $z \in [3, 4]$ for training the models (again, a selected subset is used for SVR). The remaining patterns are used for testing. For the SVR mode, we consider a linear kernel. The results are shown in Figure 2. Naturally, the kNN model cannot predict any trends.

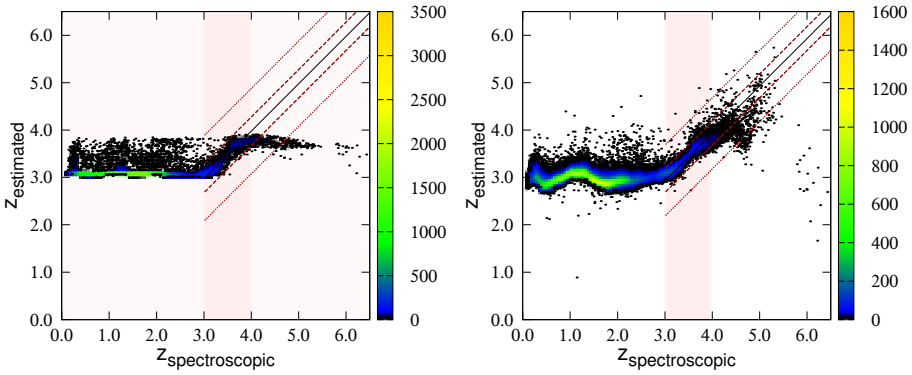

Figure 2. The figures show the performances of the kNN (left) and the SVR (right) models for the extrapolation scenario. Here, the models are only trained on quasars with redshifts $z \in [3, 4]$; the final models are then tested on all quasars. It can be clearly seen that the SVR model is capable of predicting a trend, i.e., it can predict the redshifts of quasars for $z \in [4, 5]$ even though none of these objects is given in the training set.

However, the SVR model seems to provide reasonable guesses for the patterns not covered by the training set.

4. Conclusions and Outlook

The experimental analysis indicates that the *k*-nearest neighbor regression model work well for quasars exhibiting a relatively small redshift. Further, due to computational restrictions, other sophisticated regression schemes like support vector regression cannot be trained on all data patterns, which leads to a worse performance in the context of the overall regression task. However, the latter type of models can successfully be used to detect linear structures in the data what paves the way for detecting new quasars. An interesting future research direction is the question which (combinations of) features lead to the best prediction performance for such settings. We plan to investigate this question in near future.

Acknowledgments. This work is based on data of the Sloan Digital Sky Survey.

References

Hastie, T., Tibshirani, R., & Friedman, J. 2009, The Elements of Statistical Learning (Springer)
Laurino, O., D'Abrusco, R., Longo, G., & Riccio, G. 2011, MNRAS
Schneider, et al. 2010, AJ, 139, 2360
Schölkopf, B., & Smola, A. J. 2001, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond (Cambridge, MA, USA: MIT Press)
Wang, D., Zhang, Y., & Zhao, Y. 2008, in Astronomical Data Analysis Software and Systems XVII, vol. 394, 509
Yèche, C., et al. 2010, Astronomy and Astrophysics, 523, A14