Photometric Redshift Estimation of Distant Quasars

Joris van Vugt[1]
Department of Artificial Intelligence
Radboud University Nijmegen

Supervisor: Dr. F. Gieseke[2]
Internal Supervisor: Dr. L.G. Vuurpijl[3]

July 6, 2016

Bachelor Thesis

[1]Student number: s4279859, correspondence: jorisvan.vugt@student.ru.nl
[2]Institute for Computing and Information Sciences, Radboud University Nijmegen
[3]Department of Artificial Intelligence, Radboud University Nijmegen

Abstract

Light emitted by celestial objects is shifted towards higher wavelengths when it reaches Earth. This is called redshift. Photometric redshift estimation is necessary to process the large amount of data produced by contemporary and future telescopes. However, the filter bands recorded by the Sloan Digital Sky Survey are not sufficient for accurate estimation of the redshift of high-redshift quasars. Filter bands that cover higher wavelengths, such as those in the WISE and UKIDSS catalogues provide more information. Taking these filter bands into account causes the problem of missing data, since they might not be available for every object. This thesis explores three ways of dealing with missing data: (1) discarding all objects with missing data, (2) training multiple models and (3) naively imputing the missing data.

Contents

Chapter 1

Introduction

1.1 Astroinformatics

Astroinformatics is the intersection of astronomy and computer science. Astronomy is a suitable domain for data mining techniques. According to Borne [1], data mining is essential to astronomical research. Catalogues such as the Sloan Digital Sky Survey (SDSS) [2] contain terabytes of data. Upcoming catalogues will produce such data volumes per night, per hour, or even per minute: The European Extremely Large Telescope (EELT) [3] and the Large Synoptic Survey Telescope (LSST) [4] will gather terabytes of data per night. They are expected to be fully operational in the early 2020s and will yield petabytes of data during their lifetimes. The Square Kilometre Array [5] (SKA) will gather petabytes per hour — data volumes that cannot be stored or processed anymore in their full entirety. These enormous amounts of data can not be processed manually. Data mining methods can be used to find what data is interesting enough to be analyzed further. One area of astronomy where automated data analysis is useful is *photometric redshift estimation*.

1.2 Photometric Redshift Estimation

Estimating redshift is amongst one of the most challenging problems in astronomy. The further away a quasar[1] is from Earth, the more its light spectrum is shifted towards the red side (i.e., higher wavelengths). Thus, redshift can be seen as a proxy for the distance of the object. Using a telescope, the flux

[1] quasi-stellar radio source

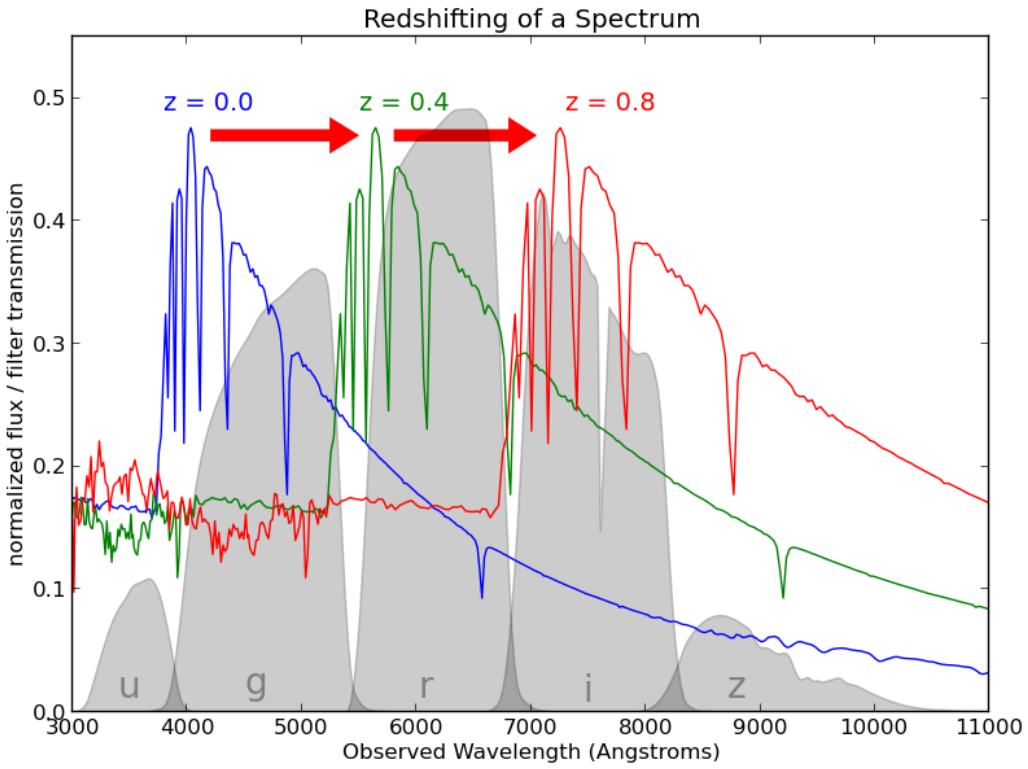

Figure 1.1: A plot of a spectrum showing the different filter bands for the SDSS catalogue and a redshifted spectrum [6]. When z increases, the spectrum is shifted towards higher wavelengths

of a lightsource can be measured for a number of filter bands. Using these to estimate an object's redshift is called photometric redshift estimation. Alternatively, a spectrum could be taken of the lightsource, which will reveal its true redshift. Taking spectra is more expensive than doing photometric analysis. To make sure that the telescope's time is spent well, we would like to only take these spectra of "interesting" (e.g., distant) objects. By estimating the redshift from the filter bands, we can predict the redshift without having to take a spectrum. Then, if the object is possibly interesting, a follow-up spectrum could be taken. Figure 1.1 contains a plot of a spectrum and the range of the different filter bands.

1.2.1 Classification Versus Regression

Using the flux in the filter bands, we can also make a distinction between stars, galaxies and quasars. Because all three of these types of celestial objects are

contained in the SDSS catalogue and I am only interested in the quasars, the quasars first need to be extracted. This task can be described as a multiclass classification task. There are a number of distinct classes – three in this case: stars, galaxies and quasars – and each object needs to be assigned to one class. Conversely, redshift estimation can be formalized as a regression task. For each instance a continuous value – its redshift – has to be predicted.

1.2.2 Algorithms Used for the SDSS Catalogue

Currently, photometric redshift for SDSS is estimated using a nearest neigbour fit[2] [7]. Classification of stars and galaxies is done using a linear threshold[3]. Both of these methods are rather simple and are not completely accurate. Finding better models for these tasks might thus be very useful and not very difficult. The SDSS webpage on classification already mentions that more complex algorithms perform better in some circumstances.

1.3 Machine Learning

Machine learning is the general task of using computers to find patterns in data. Supervised learning in particular is very important in this field. In supervised learning, first a model is fitted on data with known labels. After constructing a model, it is then applied to yet unseen inputs. Generally the data is split into a training and a test set (and sometimes also a validation set). The model is first trained using the training set. Next, we use the test set to find to what extent the model generalizes to data it has not seen before. Only the performance on the test set is relevant, since the model might have overfitted on the training data.

1.4 Datasets

1.4.1 SDSS

The Sloan Digital Sky Survey (SDSS) [2] regularly releases a lot of data. In this thesis I have used all spectroscopically confirmed objects from the SDSS data release 12 (DR12). This survey contains a total of 1,425,280 objects. The distribution of the different classes in this dataset can be found in table

[2] See `http://www.sdss.org/dr12/algorithms/photo-z/`
[3] See `http://www.sdss.org/dr12/algorithms/classify/`

Class	Count
Quasar	164,333
Galaxy	987,729
Star	273,218

Table 1.1: The different classes making up the SDSS data and their number of occurences in the SDSS dataset

1.1. For each object, there is data from five bands (u, g, r, i and z) computed using six different functions (psfMag, modelMag, petroMag, extinction and dered) resulting in a total of thirty features per instance. Any objects that were missing at least one of these features have been removed from the dataset, decreasing the size by about 200,000.

1.4.2 Quasars from SDSS, WISE and UKIDSS

The second dataset I've used contains quasars from Sloan Digital Sky Survey-III: Baryon Oscillation Spectroscopic Survey (SDSS-III/BOSS), Wide-Field Infrared Survey Explorer (WISE) *ALLWISE* data release, UKIRT Infrared Deep Sky Survey (UKIDSS) and several large-area *Spitzer Space Telescope* fields. The composition of this dataset is thoroughly descibed in Richards et al. [8]. In this thesis I have used the "candidate" dataset. Besides the SDSS u, g, r, i and z bands, this dataset also contains Y, J, H and K from UKIDSS, and 3.6 and 4.5 bands from WISE. For all 150453 objects the SDSS and WISE bands are present and for 42366 of these there is also UKIDSS data.

The filter bands from WISE and UKIDSS cover higher wavelengths (see Figure 1.2) than the filter bands from SDSS and are useful for photometric redshift estimation of high-redshift quasars. Chapter 3 discusses this topic in greater detail.

[5] From `http://inspirehep.net/record/1232359/plots`

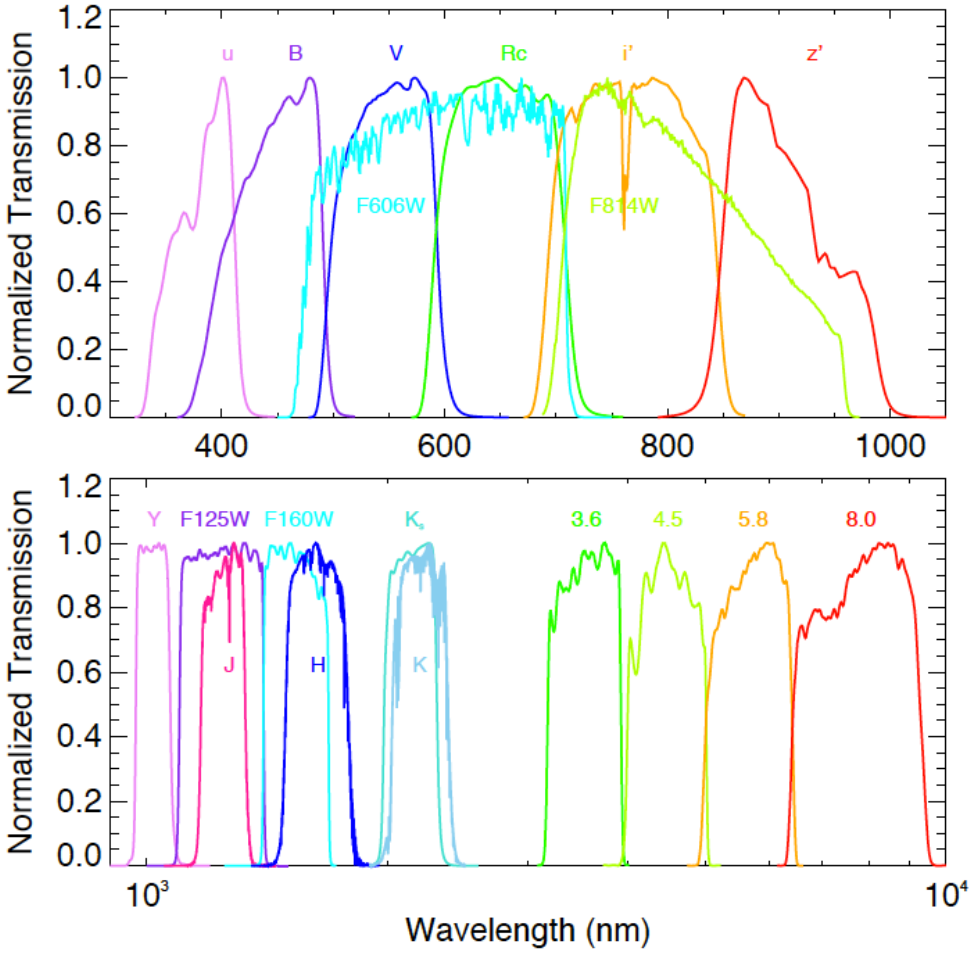

Figure 1.2: This plot shows the wavelength range of the filter bands in the WISE and UKIDSS catalogues including some filter bands from the SDSS catalogue.[5]

Chapter 2

Methods

2.1 Supervised learning

In this thesis, I have used two algorithms: k-nearest neigbours and random forest. Both algorithms were used for classification as well as regression tasks.

k-Nearest neighbours was often used in prior research (e.g., [9–11]), and is also currently being used in the SDSS catalogue [7]. For this reason, this algorithm was chosen to provide a baseline for the performance on the datasets I've used. In all experiments in this thesis, random forest performed better than k-nearest neighbours. Therefore, all results described in this thesis were achieved using random forests. However, most initial exploration of the data was done using k-nearest neighbours.

2.1.1 k-Nearest Neighbours

k-Nearest neighbours is a simple machine learning algorithm. The estimated redshift of a new object is computed by taking the average of the redshift of k points from the training set closest to that object:

$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i \tag{2.1}$$

Where $N_k(x)$ are the k closest neighbours of x. All experiments were done with $k = 12$, which is also used in Polsterer et al. [10]. Higher values of k did not significantly improve performance.

k-Nearest neigbours does not scale well when the dataset is large and has more than a few dimensions. However, using a datastructure like a *k-d tree*, this algorithm can scale to a moderate number of dimensions.

Unfortunately, k-nearest neighbours also has other downsides. If the data is sparse, the prediction might not be accurate. Moreover, due to the nature of the algoritm, it is impossible to extrapolate [13] to higher redshift quasars than were encountered during training. A more detailed explanation of k-nearest neighbours can be found in Chapter 13 of Hastie et al. [12].

2.1.2 Random Forests

A random forest [14] is an ensemble of decision trees. A number of samples are taken from the dataset and a decision tree is trained on each of them. The redshift of a new object is computed by averaging the result of all decision trees:

$$\hat{Y}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{y}_b(x) \tag{2.2}$$

Where B is the number of trees in the forest and $\hat{y}_b(x)$ is the prediction of tree b for x. A more detailed explanation of random forests can be found in Chapter 15 of Hastie et al. [12].

While random forests also somewhat suffer from the problems mentioned with k-nearest neighbours, it also has some benefits. An ensemble of models usually improves the performance a little bit. A random forest reduces the high variance of single decision trees at the cost of a slight increase in bias. Growing the trees in the forest is trivially parallelized, which can greatly reduce the time required to train the forest. Random forests have few hyperparameters and, thus, work well out of the box. Moreover, they scale well to large datasets compared to support vector machines and usually don't suffer in performance.

2.2 Evaluation metrics

2.2.1 Classification

For classification I used the accuracy, precision and recall metrics to evaluate the model's performance. These metrics are calculated using the following formulae:

$$\text{accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{2.3}$$

$$\text{precision} = \frac{TP}{TP + FP} \tag{2.4}$$

$$\text{recall} = \frac{TP}{TP + FN} \tag{2.5}$$

Where TP denotes the amount of true positives, TN true negatives, FP false positives and FN false negatives. These metrics all range between 0 and 1 and a higher score indicates a better model.

Accuracy is the fraction of correctly classified instances. However, this metric might not be very relevant if the dataset is unbalanced, or if the model has more trouble predicting one label than another. Thus, recall and precision scores are also computed per label. Recall can be interpreted as the model's ability to find all instances of a specific class. Precision can be interpreted as the model's ability not to assign the wrong class to an instance.

2.2.2 Regression

For regression tasks, I used the root mean squared error which is defined as follows:

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{n}(\hat{y}_t - y_t)^2}{n}} \tag{2.6}$$

Where $\hat{y}$ is the estimated value and y is the actual value. A lower RMSE means a better model.

Chapter 3

Approach

My approach consists of three steps, which resemble the "real-world" pipeline of estimating the redshift of distant quasars. First we have to filter the quasars from stars and galaxies. Next, we find out which quasars are interesting by making a distinction between quasars that are relatively close by ($z < 4$) and distant quasars ($z \geq 4$). Lastly, we will estimate the redshift of these distant quasars (see Figure 3.1). I will explain each of these steps in detail in this chapter. Steps 1 and 2 are less relevant to my research question, so most efforts have gone towards the last step. All steps were implemented in Python and use scikit-learn [15][1].

3.1 Finding Quasars

Finding quasars can be formalized as a classification task. I trained a random forest with 100 estimators on the SDSS dataset. The classifier achieved an accuracy of 97%. The confusion matrix of the classifier's predictions can be found in Figure 3.2 and recall and precision scores in Table 3.1.

The results are already quite good. Moreover, the precision and recall for

[1] Source code is available at `https://github.com/jvanvugt/redshift-estimation`

Figure 3.1: Pipeline for photometric redshift estimation of distant quasars

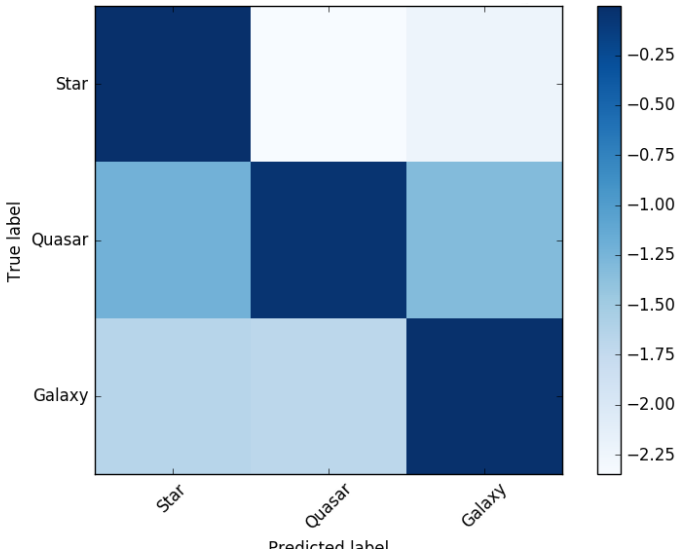

Figure 3.2: Normalized confusion matrix of the classifier's prediction plotted in logscale

Metric	Quasar	Galaxy	Star
Precision	0.93	0.95	0.99
Recall	0.89	0.96	0.99

Table 3.1: Precision and recall scores for classification of quasars, galaxies and stars

quasars can be increased using *cool star rejection* [9]. The light picked up from these cool stars is very similar to that of quasars, which can cause errors in the classification of these types of objects.

3.2 Finding Distant Quasars

Having a model specialized to estimating the redshift of high-redshift quasars might be beneficial. A model suited for this type of quasar might need some other properties, compared to a general model, such as being able to deal with a lot less training data. However, to be able to use a model specialized to high-redshift quasars in practice, a quasar first has to be classified as having a high redshift. In other words, a model trained specifically on high-redshift quasars can't be expected to work well on low-redshift qusars too. Finding out if a quasar has a high redshift can be seen as a classification task by putting a (rather arbitrary) bound at $z = 4$. We now have two classes: quasars with $z < 4$ and quasars with $z \geq 4$.

Training a random forest with 100 estimators for this task with quasars from SDSS achieves an accuracy of 99%. However, quasars with $z \geq 4$ only make up 2% of the dataset, which puts chance level at 98%. Thus, the accuracy doesn't really tell us much about the true performance of the model. Precision and recall, which are much more informative in this case, are listed in table 3.2.

Metric	**z < 4**	**z ≥ 4**
Precision	0.988	0.720
Recall	0.997	0.388

Table 3.2: Precision and recall scores for classification of high vs low redshift quasars

The precision and recall for high-redshift quasars reveal that the model doesn't actually perform very well. The model has difficulties recognizing these quasars, since the recall for this class is only 38.8%. A cause of this problem is that this task is actually a regression task (predicting the redshift), but is cast to a classification task. Quasars with $z \approx 4$ might be classified into the wrong category. In this task, recall is more important than precision. Finding out a quasar's redshift is not very high after taking a spectrum is not as bad as completely missing a high-redshift quasar. Thus, we can sacrifice some precision for an increase in recall. This can be achieved by training the

model on a lower bound (e.g., $z = 3.5$), but evaluating the model with the class boundary at $z = 4$. Using a class boundary of $z = 3.5$ increased the recall to 56%. Naturally, the boundary can be moved to achieve the desired recall score. Alternatively, the decision boundary of the random forest can be moved. For example, for a quasar to be classified as high-redshift, only a third of the trees in the forest need to predict this class, instead of just picking the class predicted by the majority of the trees.

3.3 Redshift Estimation for Distant Quasars

The last stage of the pipeline consists of estimating a quasar's redshift. Contrary to the two prior stages, this stage is a regression task, rather than a classification task. All results in this section were achieved using a random forest regressor with 100 estimators.

Figure 3.3 shows a scatter plot of the (predicted) photometric redshift versus the (true) spectroscopic redshift. Most points are on or near the diagonal, which means that the model was able to accurately predict the redshift. However, almost all quasars with approximately $z \geq 4$ are predicted to have a lower redshift. Figure 3.4 shows that this problem is not caused by the relative low frequency of high-redshift quasars in the dataset. The model used to generate this graph was only trained on quasars with $z \geq 4$. The plot shows that for quasars with approximately $z \geq 4.8$ the model's prediction is not any better than random guessing, as the points are uniformly distributed.

The model's inability to accurately predict the redshift of quasars with $z \geq 4.8$ is caused by the spectrum being redshifted beyond the filter bands in the SDSS catalogue. Because of this, the u, g, r, i and z bands do not contain any useful information for the model to base a prediction on.

3.3.1 Using another dataset

The dataset composed by Richards et al. [8] has filter bands from UKIDSS and WISE which cover higher wavelengths. The flux in these bands might contain the information necessary for photometric redshift estimation of high-redshift quasars. Since the UKIDSS features are only present for a third of the quasars, a way to deal with this missing data problem is necessary. In this thesis, I discuss three options for dealing with this missing data problem:

1. Only use the quasars with complete data

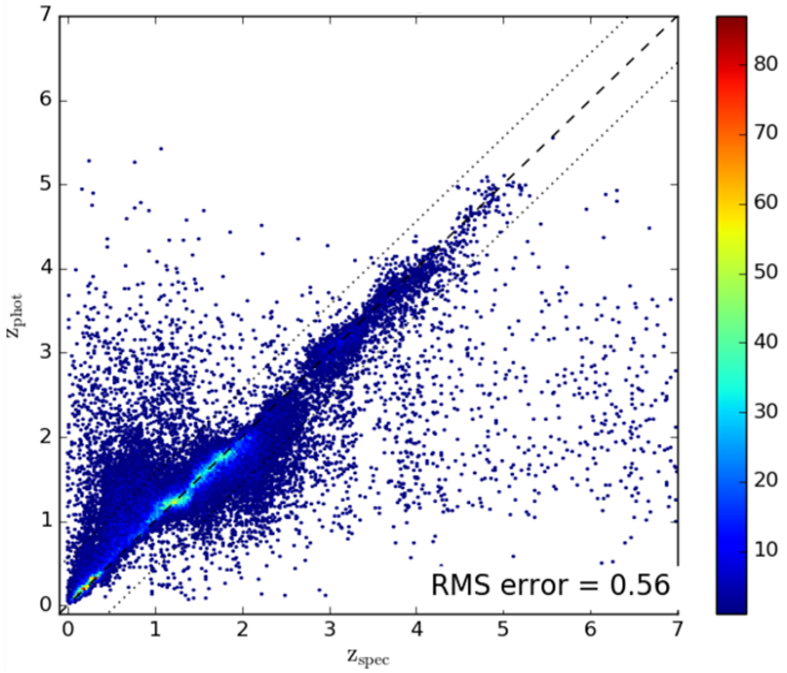

Figure 3.3: Scatter plot of the redshift predicted by the model (vertical axis) versus the true redshift.

2. Train 2 models: one model for complete data (i.e., SDSS, UKIDSS and WISE features) and one model for quasars with partially missing features (i.e., just SDSS and WISE features).

3. Impute the missing UKIDSS data

Figure 3.5 shows the performance of a model trained just on the SDSS features from this dataset. By training only on the SDSS features, this model provides a baseline for the new dataset. The plot shows a performance similar to the performance on the other dataset. The test set consists of approximately 20000 quasars with complete data. This test set will be used for all following experiments to allow for comparisons of the performances. Not having a fixed test set could result in uncomparable results, since quasars with missing UKIDSS data might differ in some way from quasars with full data.

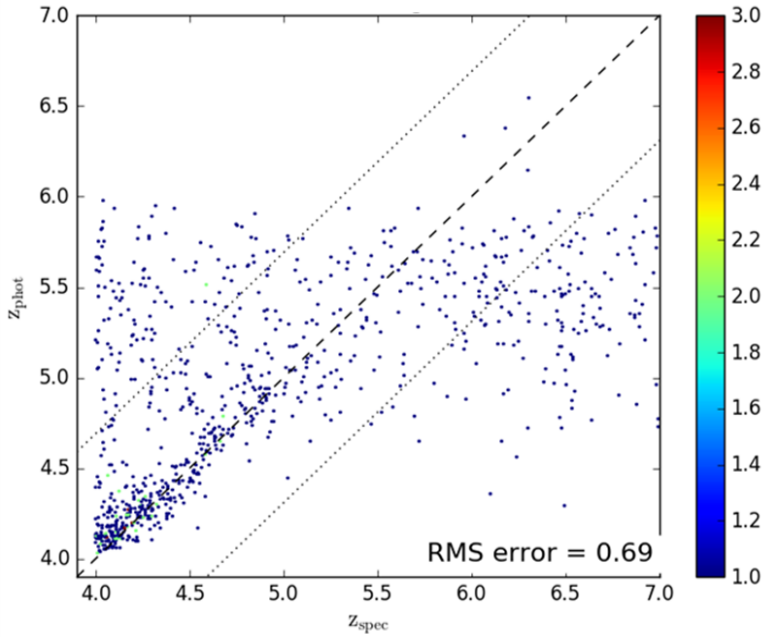

Figure 3.4: Scatter plot of the redshift predicted by the model (vertical axis) versus the true redshift just with quasars with $z \geq 4$.

Quasars with complete data

The first way of dealing with missing data is to simply discard all quasars with missing data. The scatter plot for the performance of this model can be found in Figure 3.6. The performance when using all features ($RMSE = 0.29$) is a lot better when using just SDSS features ($RMSE = 0.52$). Some of the high-redshift quasars' redshifts are predicted accurately, but there are some mispredictions as well.

Figure 3.7 shows all remaining combinations of features from SDSS, WISE and UKIDSS. SDSS and WISE features combined seem to contain almost as much information as SDSS, WISE and UKIDSS features combined. The performance when using just SDSS and WISE features is only slightly worse than using features from all three catalogues.

The major downside of this model is that it can only be used when features from all three catalogues are available. However, using a technique called surrogate splits [16], a random forest can still be used even when some data is missing. In surrogate splits, when a split has to be made on a missing value, the split will be made with correlated objects where that particular value *is* present instead.

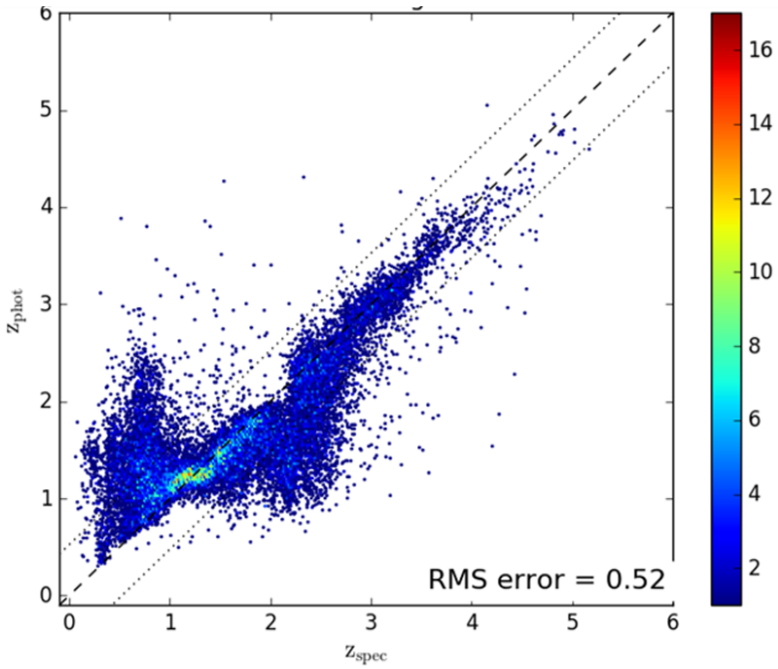

Figure 3.5: Scatter plot of the redshift predicted by the model (vertical axis) versus the true redshift. Trained on the SDSS features in the dataset by Richards et al.

Training two models

The second approach to handling the missing data problem is to train two models. Missing data in this dataset is very regular, because only the entirety of UKIDSS is ever missing. Because of this fact, only 2 different patterns occur: either all data is present or UKIDSS is missing. Therefore it is still feasible to train multiple models.

When estimating the redshift of an unseen quasar, the quasar's redshift is predicted by the model tailored to the features which are present for that quasar. In this case that is either the model with no missing features or the model for quasars with missing UKIDSS features. This technique achieved a root mean squared error of 0.30, which is similar to that of using just SDSS and WISE data.

Imputing missing data

The last technique described here for dealing with missing data is to impute (i.e. fill-in) the missing data. I have tried only a few naive approaches:

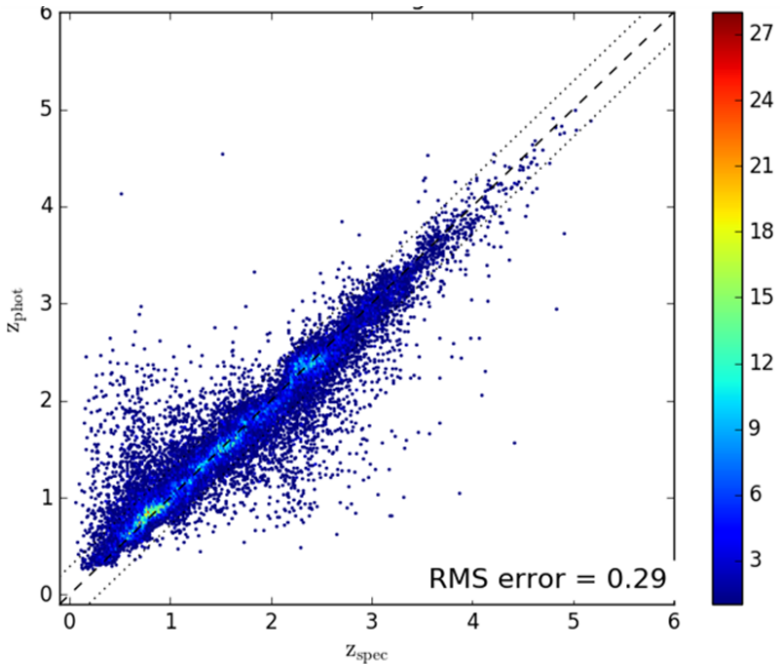

Figure 3.6: Scatter plot of the redshift predicted by the model (vertical axis) versus the true redshift. Trained on quasars without missing data in the dataset by Richards et al.

(1) replace the missing data with the mean of that column and (2) replace the missing data with the median of that column. Both techniques had a similar performance ($RMSE = 0.30$), which is again similar to that of using just SDSS and WISE features and the approach with two models. However, in contrary to the latter technique, this approach only requires training one model.

A more sophisticated approach such as imputing data using nearest neighbours might perform slightly better, although I do no consider this very likely. If useful information can be imputed using data that is already present, the model will already learn that information from that data. The intermediate imputing step shouldn't provide any more information.

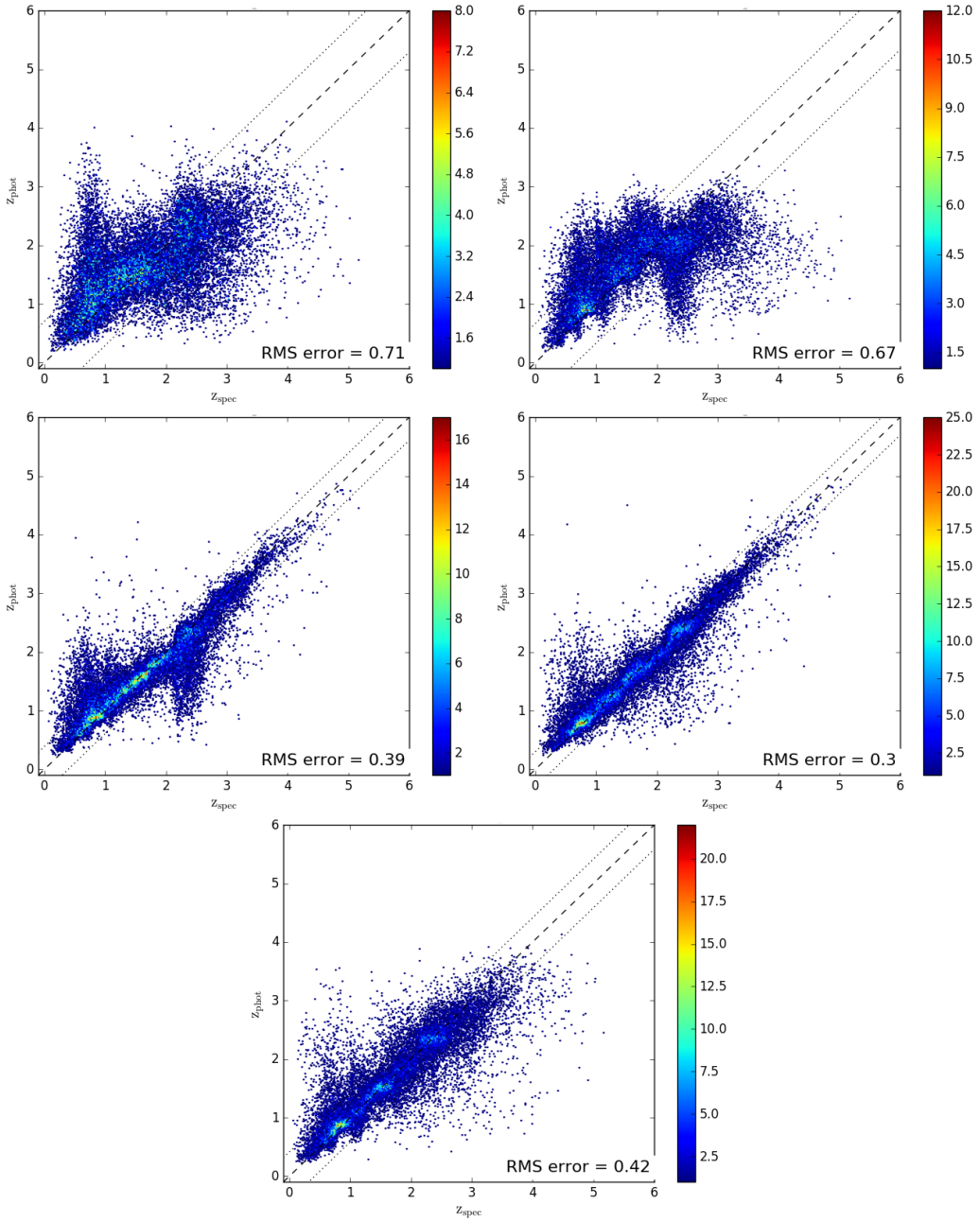

Figure 3.7: Models trained using just WISE (top left), UKIDSS (top right), SDSS and UKIDSS (middle left), SDSS and WISE (middle right) and WISE and UKIDSS (bottom middle)

Chapter 4

Conclusion

Modern telescopes produce so much data that algorithms are necessary to extract interesting data. This is also true for computing an object's redshift. In this thesis I've described a pipeline consisting of three stages for photometric redshift estimation of quasars.

The first step of this pipeline consists of classifying quasars, galaxies and stars with the goal of extracting quasars from other objects in space. A random forest trained on the features available in the SDSS catalogue performed very well on this task.

The second step of the pipeline is classifying low and high-redshift quasars. This step introduced some problems, since it is naturally a regression task – redshift is a continuous value after all. The difficulties in correctly classifying high-redshift quasars can be partially overcome by moving the decision boundary.

The last step of the pipeline is estimating the quasars' redshifts from photometric data. When evaluation the dataset in its entirety, this works fairly well. However, the redshift of high redshift quasars, which only make up a small fraction of the dataset, turned out to be very difficult to predict. Since the data in the SDSS catalogue does not contain useful information for these quasars, another dataset with filter bands from WISE and UKIDSS, besides SDSS, that cover higher wavelengths was considered. Using data from these catalogues improved performance for the dataset in its entirety. However, it is hard to draw any sound conclusions about the performance of the model on high-redshift quasars as there are only few present in the dataset.

The dataset with features from SDSS, WISE and UKIDSS also had some missing data. Three approaches for dealing with this problem were discussed: (1) discarding all missing data, (2) training multiple models to fit the different patterns of missing data and (3) imputing the missing data. Imputing the

missing data with either the mean or the median of the column proved to be the best approach. The advantage of this technique over discarding all missing data is obvious: even objects with missing data can be useful. Furthermore, only one model needs to be trained, in contrast with the approach where multiple models were trained.

Chapter 5

Bibliography

[1] K. D. Borne. Scientific data mining in astronomy. In *Next Generation of Data Mining*, pages 91–114. Chapman and Hall/CRC, 2008.

[2] Donald G York, J Adelman, John E Anderson Jr, Scott F Anderson, James Annis, Neta A Bahcall, JA Bakken, Robert Barkhouser, Steven Bastian, Eileen Berman, et al. The sloan digital sky survey: Technical summary. *The Astronomical Journal*, 120(3):1579, 2000.

[3] Roberto Gilmozzi and Jason Spyromilio. The european extremely large telescope (e-elt). *The Messenger*, 127(11):3, 2007.

[4] P. A. Abell, J. Allison, S. F. Anderson, J. R. Andrew, J. R. P. Angel, L. Armus, D. Arnett, S. J. Asztalos, T. S. Axelrod, S Bailey, et al. Lsst science book, version 2.0. 2009.

[5] Christopher Carilli and Steve Rawlings. Science with the square kilometer array: motivation, key science projects, standards and assumptions. *arXiv preprint astro-ph/0409274*, 2004.

[6] J.T. Vanderplas, A.J. Connolly, Ž. Ivezić, and A. Gray. Introduction to astroml: Machine learning for astrophysics. In *Conference on Intelligent Data Understanding (CIDU)*, pages 47 –54, oct. 2012.

[7] I Csabai, L Dobos, M Trencséni, G Herczegh, P Józsa, N Purger, T Budavári, and AS Szalay. Multidimensional indexing tools for the virtual observatory. *Astronomische Nachrichten*, 328(8):852–857, 2007.

[8] Gordon T Richards, Adam D Myers, Christina M Peters, Coleman M Krawczyk, Greg Chase, Nicholas P Ross, Xiaohui Fan, Linhua Jiang,

Mark Lacy, Ian D McGreer, et al. Bayesian high-redshift quasar classification from optical and mid-ir photometry. *The Astrophysical Journal Supplement Series*, 219(2):39, 2015.

[9] Kai Lars Polsterer, Peter-Christian Zinn, and Fabian Gieseke. Finding new high-redshift quasars by asking the neighbours. *Monthly Notices of the Royal Astronomical Society*, 428(1):226–235, 2013.

[10] Kai Lars Polsterer, Fabian Gieseke, Christian Igel, and Tomotsugu Goto. Improving the performance of photometric regression models via massive parallel feature selection. In *Proceedings of the 23rd Annual Astronomical Data Analysis Software & Systems Conference*, 2013.

[11] Yanxia Zhang, He Ma, Nanbo Peng, Yongheng Zhao, and Xue-bing Wu. Estimating photometric redshifts of quasars via the k-nearest neighbor approach based on large survey databases. *The Astronomical Journal*, 146(2):22, 2013.

[12] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin, 2001.

[13] Fabian Gieseke, Kai Lars Polsterer, and Peter-Christian Zinn. Photometric redshift estimation of quasars: Local versus global regression. *Proceedings of the Astronomical Data Analysis Software and Systems*, 2011.

[14] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[16] Ad Feelders. Handling missing data in trees: surrogate splits or statistical imputation? In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 329–334. Springer, 1999.