ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1.
Available from http://www.i6doc.com/en/.

Uncertain Photometric Redshifts via Combining Deep Convolutional and Mixture Density Networks

A. D'Isanto[1], K. Polsterer[1].

1- Heidelberg Institute for Theoretical Studies (HITS)
Schloss-Wolfsbrunnenweg 35, 69118 Heidelberg - GERMANY

Abstract. The need for accurate photometric redshifts estimation is a major subject in Astronomy. This is due to the necessity of efficiently obtaining redshift information without the need for spectroscopic analysis. We propose a method for determining accurate multi-modal predictive densities for redshift, using Mixture Density Networks and Deep Convolutional Networks. A comparison with the Random Forest is carried out and superior performance of the proposed architecture is demonstrated.

1 Introduction

Determining the distance of an object via redshift is an important task in Astronomy. Redshift is the measure of the shift of galaxies spectral lines due to the expansion of the Universe and it is directly related to their distances. Therefore, it plays a fundamental role in cosmological research. Redshift is measured through spectroscopical analysis. Due to long integration times and costly instrumentation requirements, it is not possible to measure this property for all objects in the Universe. Therefore an alternative way is to estimate the redshifts based on photometric measurements. However, the uncertainty of such a photometric approach is higher than the measurement errors in spectroscopy. For this reason, the astronomical community is interested in quantifying the uncertainty of redshift estimates via predictive distributions instead of merely working with point estimates. We propose two neural network models inspired by Mixture Density Networks (MDN) [1]. The first architecture is a deep MDN designed to take photometric features as inputs and which generates predictive redshift distributions. The second architecture combines a Deep Convolutional Network (DCN) [2] with a MDN, in order to obtain probability densities for redshift, given images as input. In particular the latter approach achieves better predictions due to its use of image data. In contrast to using condensed pre-defined features, this allows to capture more details of the objects. We compare the results obtained with a widely used tool in the related literature, the Random Forest (RF) [3] [4]. Furthermore, in this paper, we use two statistical tools, namely the *continuous rank probability score* (CRPS) and the *probability integral transform* (PIT), in order to properly estimate the quality of the obtained results [5].

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1.
Available from http://www.i6doc.com/en/.

2 Statistical tools: CRPS and PIT

In this section we briefly describe the statistical tools used to evaluate the predictions of the proposed models. As discussed in [6], a predictive distribution explains well an observation if it is well calibrated and sharp; as stated in [6], *calibration expresses the consistency between predictions and observations, while sharpness refers to the concentration of the predictions in the probability distribution.* CRPS quantifies both desired properties, while the PIT provides a visual appreciation of them. The CRPS [7] is meant to compare a distribution with an observation (see Fig.1):

$$CRPS = CRPS(F, x_a) = \int_{-\infty}^{+\infty} [F(x) - F_a(x)]^2 dx \tag{1}$$

where $F(x)$ and $F_a(x)$ represent respectively the cumulative density functions (CDFs) of the probability density function (PDF) and of the observation, namely: $F(x) = \int_{-\infty}^{x} f(t)dt$ and $F_x = H(x - x_a)$, with $H(x)$ being the Heaviside step-function. We use the CRPS as a score function to express the results of the predictions and as a loss function for the proposed neural networks.

The PIT is defined by the value given by the CDF of the predictions F_t at the observation x_t, that is to say: $p_t = F_t(x_t)$. If the predictions are ideal, then

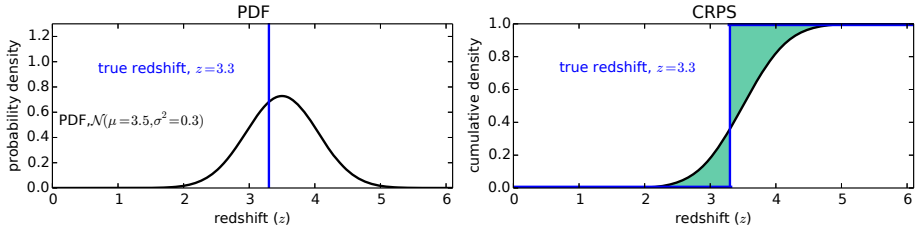

Fig. 1: Meaning of probability density function (*PDF*) and continuous ranked probability score (*CRPS*).

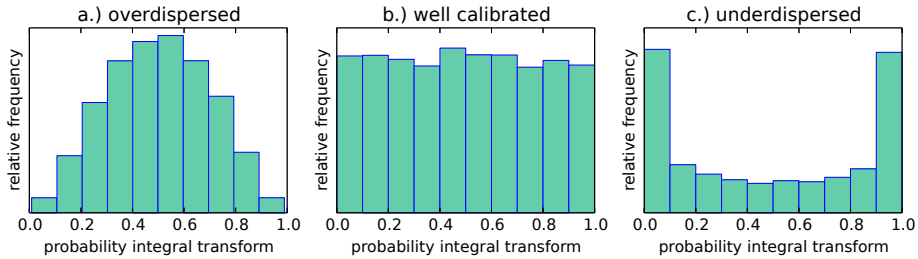

Fig. 2: Three different examples of probability integral transforms (*PIT*s) for overdispersed, well calibrated and underdispersed distributions.

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1. Available from http://www.i6doc.com/en/.

the distribution p_t is uniform. In Fig. 2, this can be verified by plotting the histogram of the distribution: if it shows a uniform shape, than the distribution is well calibrated; if it is U-shaped or center-peaked, it is underdispersed or overdispersed, respectively. From the analysis of the PIT it is possible to infer whether or not the distribution is biased.

3 Deep learning algorithms

In the next two subsections a description of the deep learning algorithms we used for the experiments follows.

3.1 Mixture Density Network

The Mixture Density Network (MDN) [1] is a particular model of Multilayer Perceptron with an output defined via a mixture model. The output distribution is a mixture of Gaussians $p(\theta|x) = \sum_{j=1}^{n} \omega_j \mathcal{N}(\mu_j, \sigma_j)$, with $\mathcal{N}(\mu_j, \sigma_j)$ being a normal distribution. The means, variances and weights, are parametrized by the outputs z of the network:

$$\mu_j = z_j^{\mu} \ , \qquad \sigma_j = \exp(z_j^{\sigma}) \ , \qquad \omega_j = \frac{\exp(z_j^{\omega})}{\sum_{i=1}^{n} \exp(z_i^{\omega})} \ . \tag{2}$$

Commonly the MDN employs negative log-likelihood as a loss function. In this work we use the CRPS as the loss function, because we want the trained MDN to produce predictive distributions that are both well calibrated and sharp as measured by the CRPS.

3.2 Deep Convolutional Network

A Deep Convolutional Network (DCN) [2] is a neural network in which several convolutional and sub-sampling layers are coupled with a fully-connected network, which is particularly adept at learning from raw image data. In our case, we want to estimate redshifts directly from images, without the need to extract photometric features. In fact the DCN, filtering the input images with proper filter weights, is able to automatically extract the *feature maps* that become the input data of the fully-connected part. We combine a modified version of the LeNet-5 [2] architecture with the MDN (see Section 3.1), obtaining what we call a Deep Convolutional Mixture Density Network (DCMDN). In Tab. 1 there are the two different architectures used for the experiments respectively with 28x28 and 16x16 images. Many different architectures had been evaluated, including more compact and less deep convolutional parts. The architectures found to perform best have been chosen for this work. We are aware that cross validation is an appropriate tool to prevent overfitting of the architecture. Due to computational limitations we use a simple hold out strategy, only. The architectures were designed to run on GPU, using a cluster equipped with Nvidia Titan X.

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1. Available from http://www.i6doc.com/en/.

#	Type	Size	Maps	Activ
1	input	28x28	/	/
2	Conv	3x3	256	tanh
3	Pool	2x2	256	tanh
4	Conv	2x2	512	tanh
5	Pool	2x2	512	tanh
6	Conv	3x3	512	ReLu
7	Conv	2x2	1024	ReLu
8	MDN	500	/	tanh
9	MDN	100	/	tanh
10	output	15	/	Eq. 2

#	Type	Size	Maps	Activ
1	input	16x16	/	/
2	Conv	3x3	256	tanh
3	Pool	2x2	256	tanh
4	Conv	2x2	512	tanh
5	Pool	2x2	512	tanh
6	Conv	2x2	1024	ReLu
7	MDN	500	/	tanh
8	MDN	100	/	tanh
9	output	15	/	Eq. 2

Table 1: DCMDN architectures for the two image sizes.

4 Experiments

The data used for the experiments are taken from the Sloan Digital Sky Survey Quasar Catalog V [8], based on the seventh data release of the Sloan Digital Sky Survey (SDSS), consisting in $105,783$ spectroscopically confirmed quasars, in a redshift range between 0.065 and 5.46. We perform the experiments with the proposed architectures using a random subsample of $50,000$ data items. Each data item has a feature and an image representation in five different filter bands (*ugriz*). We compare the performances of MDN and DCMDN with the widely used RF [4]. The RF, in its original design, does not produce predictive distributions. In order to obtain a predictive distribution, we first collect the predictions $z_{t,n}$ of each individual decision tree t in the RF, for every n-th data item. We take $T = 256$ number of trees in the forest and define the predictive distribution for the RF by fitting a mixture of five Gaussian components to the outputs, $p(\theta|x) = \sum_{j=1}^{5} \omega_j \mathcal{N}(\theta|(\mu_j, \sigma_j))$, as described also in Section 3.1 for the MDN. The RF and the MDN are trained on the feature representation of the data items. The original five features are *ugriz* magnitudes extracted from the images. To avoid biases due to object intrinsic parameters like luminosity, all possible pairwise differences (aka. colors) are used additionally. Therefore a 15-dimensional feature vector is used as input. We divide the dataset in a training and a test set, both containing $25,000$ patterns. The DCMDN is trained on the image representation of the data items. The images are obtained using the *Hierarchical Progressive Surveys* (HIPS) [9] protocol and performing a proper cutout on client side, in order to obtain the desired dimensions. Each data item is originally a stack of five images in the *ugriz* filters. Similarly to the features, we additionally form the color images from the *ugriz* images by taking all possible pairwise differences, thus obtaining a stack of 15 images. The images are taken in two sizes: 28x28 pixels^2 and 16x16 pixels^2. Every object is then represented by a tensor of dimension 15x28x28 or 15x16x16. In order to make the network rotation invariant, we also perform data augmentation. We take rotations of each image at 0, 90, 180, 270 degrees, obtaining a training set of $100,000$ images, a validation set of $50,000$ images and a test set of $50,000$ images. Dropout with a ratio of 60% together with early stopping is used to limit overfitting.

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1.
Available from http://www.i6doc.com/en/.

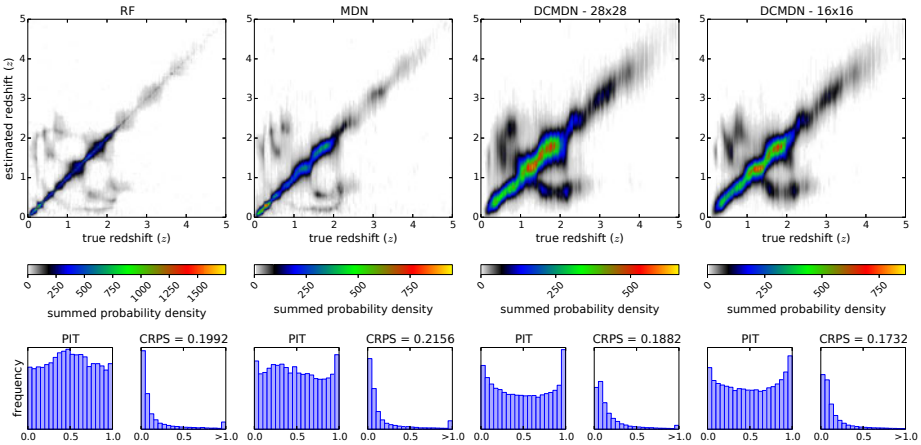

Fig. 3: Results of the prediction obtained with the MDN and the DCMDN (two different input sizes), compared with the RF results. For each experiment, three plots are present. The upper plots compare the spectroscopic redshift with the predictive density produced by the model, with the color indicating the summed probability density of the distributions. In the two lower plots, the histogram of the PIT values and the histogram of the individual CRPS values, are shown. The mean CRPS value is also reported.

5 Analysis of results

The results of the experiments are reported in Fig. 3. In the RF experiment, the model reaches a score of 0.20 and the PIT histogram shows overdispersion. Feeding a RF with the plain pixel information, as done for the DCMDN, results in a CRPS of 0.195 with high overdispersion. The performance of the MDN is a bit worse in terms of the CRPS (score of 0.21) compared to the RF, with a better calibrated PIT. With the DCMDN architecture a significantly better result in terms of the CRPS (0.19 for 28x28, 0.17 for 16x16 images) is achieved. The usual deviation of experiments with other data folds is 0.005 in CRPS. The resulting PIT is acceptable, although it is still underdispersed, which slightly improves for the 16x16 images experiment. The reduced size of the images is more focusing on the central region and ignores neighboring objects and hence improves the result in both, CRPS and PIT. The reason for the better overall performance of the DCMDN is the fact, that the features described in Section 4 use only a fraction of the available information. In fact, the process of extracting historically motivated features is common in Astronomy. In this process a lot of information gets lost. Instead, using images, the DCMDN is able to automatically determine thousands of features, leading to a better prediction of the redshifts.

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1. Available from http://www.i6doc.com/en/.

6 Conclusions

The main purpose of this work was to show how to produce predictive densities for redshifts using deep learning architectures. Using a Gaussian Mixture Model as output, we generate very good probabilistic predictions based on features or images as input. The comparison with a RF based approach shows better performance for our proposed architectures. We show that the proposed DCMDN displays the best performance as it makes use of the entire information given by the images. The use of the PIT allows us to evaluate the produced predictive distributions for underdispersion and overdispersion, indicating that some optimization with respect to calibration can still be done.

We firmly believe that the results obtained with our proposed methods need little improvements before becoming a standard in predicting probabilistic redshift based on photometric data. As regression problems are very common in Astronomy, this approach can easily be applied to other scientific questions.

Acknowledgments

The authors gratefully acknowledge the support of the Klaus Tschira Foundation. This work is based on data provided by the SDSS. SDSS-III is managed by the Astrophysical Research Consortium for the Participating Institutions of the SDSS-III Collaboration. Funding for SDSS-III has been provided by the Alfred P. Sloan Foundation, the Participating Institutions, the National Science Foundation, and the U.S. Department of Energy Office of Science. The SDSS-III web site is http://www.sdss3.org/. We used Theano [10] to run our experiments.

References

[1] Christopher M. Bishop. Mixture density networks. Technical report, 1994.

[2] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998.

[3] Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001.

[4] S. Carliles, T. Budavári, S. Heinis, C. Priebe, and A. S. Szalay. Random Forests for Photometric Redshifts. *ApJ*, 712:511–515, March 2010.

[5] K.L. Polsterer, A. D'Isanto, and F. Gieseke. Uncertain photometric redshifts. 2016.

[6] T. Gneiting, A. E. Raftery, A. H. Westveld, and T. Goldman. Calibrated Probabilistic Forecasting Using Ensemble Model Output Statistics and Minimum CRPS Estimation. *Monthly Weather Review*, 133:1098, 2005.

[7] H. Hersbach. Decomposition of the Continuous Ranked Probability Score for Ensemble Prediction Systems. *Weather and Forecasting*, 15:559–570, October 2000.

[8] Richards G. T. Hall P. B. Schneider, D. P. et al. VizieR Online Data Catalog: The SDSS-DR7 quasar catalog (Schneider+, 2010). *VizieR Online Data Catalog*, 7260, May 2010.

[9] P. Fernique, M. G. Allen, T. Boch, A. Oberto, F.-X. Pineau, D. Durand, C. Bot, L. Cambrésy, S. Derriere, F. Genova, and F. Bonnarel. Hierarchical progressive surveys. Multi-resolution HEALPix data structures for astronomical images, catalogues, and 3-dimensional data cubes. *A&A*, 578:A114, June 2015.

[10] Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May 2016.