Erik Vinicius Rodrigues de Lima

Photometric redshifts for S-PLUS using machine learning techniques

São Paulo

2020

ERIK VINICIUS RODRIGUES DE LIMA

Photometric redshifts for S-PLUS using machine learning techniques

Versão corrigida

Dissertação apresentada ao Instituto de Astronomia, Geofísica e Ciências Atmosféricas da Universidade de São Paulo como requisito parcial para a obtenção do título de Mestre em Ciências.

Área de concentração: Astronomia extragaláctica.

Orientador: Prof. Dr. Laerte Sodré Júnior.

São Paulo
2020

*À meu avô Marcelino e meu tio
Wanderley (in memoriam).*

Agradecimentos

Primeiramente, agradeço à minha mãe, Elenita, que sempre me apoiou, me ensinou a ter caráter e deu a educação que levarei pelo resto da vida. Também me apoiou na decisão de cursar Física e depois perseguir o mestrado em Astronomia, apesar de todas as dificuldades, pois sabe que este é o meu sonho.

À Mylena, por ter ficado ao meu lado durante todos estes anos, tornando-os mais fáceis de lidar com seu carinho e companheirismo.

Também agradeço à minha família por todo o apoio que recebi nesses dois anos de mestrado.

Agradeço em especial às minhas amigas Thayse, Lilianne e Luísa, por dividirem vários cafés e histórias, me fazendo sentir que são pessoas com as quais eu posso contar em diversos momentos. Ao Pablo e Marcelo, também alunos do Laerte, pelas boas conversas, risadas, e discussões. Além disso devo citar a Roberta, pelos eventos e papos sobre aprendizado de máquina. Não menos importante, agradeço também ao Gê, Natália, Ivan e Fábio, pelos momentos de descontração, e outros, que me deram momentos de leveza e risos que são tão necessários no nosso dia a dia.

Agradeço especialmente ao meu orientador, Prof. Dr. Laerte Sodrê Jr. pois, afinal, se ele não tivesse me aceito como aluno de mestrado eu não estaria aqui, tampouco teria conhecido esta incrível área de pesquisa que é o aprendizado de máquina. Também devo citar a Profa. Dra. Cláudia de Oliveira que, por ser minha relatora, apresentou importantes feedbacks no decorrer do meu trabalho, e me incentivou a ter resultados cada vez melhores para serem usados no projeto do qual é coordenadora, o S-PLUS.

Não poderia deixar de fora desta seção os pós-doutorandos Maria Luiza Dantas e Marcus V. C. Duarte. Ela me ajudou dando o pontapé inicial no meu aprendizado em programação, permitindo que eu criasse ou modificasse os programas utilizados neste trabalho. Já o Marcus me ajudou durante todo o tempo em que estive no instituto, suprimindo minhas dúvidas tanto em programação quanto na forma de analisar meus resultados nos meus primeiros meses no IAG.

Também agradeço à equipe do IAG, desde as moças que fazem o café que eu tomo quase todo dia, o pessoal da secretaria e coordenação, sempre muito prestativos quando eu necessitava de sua ajuda, até o pessoal da informática, que permitiu que este trabalho fosse possível de ser realizado em primeiro lugar.

Por último, mas não menos importante, agradeço ao apoio financeiro prestado pelo CNPq, através da bolsa de mestrado, que me permitiu morar em São Paulo durante esses dois anos para realizar esta pesquisa.

Resumo

O foco deste trabalho é a obtenção de redshifts fotométricos de galáxias utilizando os códigos de aprendizado de máquina ANNz2, GPz e modelos de aprendizado profundo feitos com o Keras. Nós aproveitamos a excelente oportunidade que o novo mapeamento multicolor do céu austral, chamado Southern Photometric Local Universe Survey (S-PLUS), oferece ao utilizar um sistema único de filtros, composto por cinco filtros largos e sete filtros estreitos. Além do uso das magnitudes, também é possível utilizar características não fotométricas em métodos de aprendizado de máquina, como o tamanho dos objetos, sua largura à meia altura e seu brilho superficial, de forma a melhorar os resultados.

Para este trabalho, foram usados dados provenientes do Data Release 1 do S-PLUS, unido à outros dois grandes projetos, o Data Release 15 do Sloan Digital Sky Survey (SDSS) e o catálogo unWISE do Wide-field Infrared Survey Explorer (WISE), na região da Stripe-82.

Dentre os três algoritmos comparados neste trabalho, o que apresentou a melhor performance geral foi o baseado em aprendizado profundo. Os redshifts fotométricos obtidos com este método têm precisão (σ_{NMAD}) de 2.49% para galáxias com magnitude r entre 16 e 21, com viés igual a 0.4% e fração de outliers de 0.64%.

Em comparação com o método utilizado atualmente para a estimativa de redshifts fotométricos no S-PLUS, o código de ajuste de *templates* BPZ, foi constatado que os métodos de aprendizado de máquina têm precisão superior, viés inferior e menor fração de outliers.

Uma análise das funções de distribuição de probabilidades é feita, concluindo-se que os algoritmos de aprendizado de máquina apresentam distribuições mais largas quando comparadas às do código BPZ.

Palavras-chave: Redshift fotométrico, Aprendizado de máquina, Galáxias, Levantamentos de galáxias

Abstract

This work focuses on the obtention of photometric redshifts of galaxies using the machine learning codes ANNz2, GPz, and Deep Learning models made with Keras. We take advantage of the great opportunity that the new multiband survey of the austral sky, called Southern Photometric Local Universe Survey (S-PLUS), provides due to the adoption of an unique filter system, composed of five broad-band filters and seven narrow-band filters. Besides the use of magnitudes, it is also possible to use non-photometric features with machine learning methods, such as object sizes, their full width at half maximum and their maximum surface brightness, in order to improve the results.

This work used data from the S-PLUS Data Release 1, together with two other large projects, the Sloan Digital Sky Survey (SDSS) Data Release 15, and the unWISE catalogue of the Wide-field Infrared Survey Explorer (WISE), in the Stripe-82 region.

Amongst the three algorithms compared in this work, the one which performs better overall is the deep-learning based method. The photometric redshifts obtained with this code have precision (σ_{NMAD}) of 2.49% for galaxies with r-band magnitude between 16 and 21, bias of 0.4% and outlier fractions equal to 0.64% .

When compared to the currently used method for photometric redshift determination in S-PLUS, the template fitting code BPZ, it is noticed that machine learning methods have higher accuracy, less bias and lower outlier fraction.

An analysis regarding the probability distribution function is made, concluding that the machine learning algorithms present broader distributions when compared to the BPZ code.

Keywords: Photometric redshift, Machine learning, Galaxies, Galaxy surveys

Contents

List of Figures

List of Tables

List of Abbreviations

1	Introduction	1
1.1	Contemporary Surveys	2
1.2	Distance measurements	4
1.3	Photometric Redshifts	7
1.4	Photometric Redshifts with Template-Fitting	9
1.5	Photometric Redshifts with Machine Learning	10
1.6	Template-fitting or Machine learning?	11
1.7	Objectives	11
2	Methods	13
2.1	Artificial Neural Networks	13
2.1.1	History	13
2.1.2	The Multi Layer Perceptron	14
2.1.3	The training process	17
2.2	Boosted Decision/Regression Trees (BDTs)	18
2.3	Gaussian Processes	19
2.3.1	Weight-space view	19
2.3.2	Function-space view	20
2.4	Softwares used in this work	22
2.4.1	Artificial Neural Networks for photometric redshifts (ANNz2)	22
2.4.2	Gaussian Processes for photometric redshifts (GPz)	24
2.4.3	Keras	25
3	The Data	29
3.1	Datasets	29
3.2	Pre-processing	31
3.3	Training, validation and testing samples	35

4	Results	37
4.1	Metrics	37
4.2	Missing feature analysis	38
4.2.1	Scatter (σ_{NMAD})	39
4.2.2	Bias (μ)	41
4.2.3	Outlier fraction (η)	43
4.3	Input features analysis	44
4.4	Single-point estimates	48
4.5	Loss Analysis	53
4.6	PDF Analysis	54
4.6.1	ANNz2	54
4.6.2	GPz	56
4.6.3	Deep Learning	58
4.6.4	Comparison between methods	61
4.7	Summary	64
5	Conclusions	65
5.1	Future perspectives	66
	Bibliography	69
A	Appendix	79
A.1	SDSS Query	79
A.2	Method Parameters	79
A.2.1	ANNz2 parameters	80
A.2.2	GPz parameters	81
A.2.3	Keras model parameters	83
A.3	Training and Testing sample distribution per feature	84
A.4	Odds values for the samples analysed in section 4.3	86
A.5	Single-point estimate (SPE) results for each method	87
A.5.1	ANNz2	88
A.5.2	GPz	90
A.5.3	Deep learning	92
A.6	Detailed plots for the r-mag and z simultaneous analysis - SPE	94
A.6.1	Number of objects per bin	94
A.6.2	ANNz2	95

A.6.3	GPz	97
A.6.4	Deep Learning	99
A.6.5	BPZ	101
A.7	Results for all missing band scenarios	103
A.8	Additional plots to PDF analysis	104
A.9	PDF comparison between methods	106

List of Figures

1.1	Illustration of the Doppler-Fizeau effect.	2
1.2	Simplified representation of spectra. The black lines represent absorption lines. The top spectrum is in the rest-frame (i.e. a lab), the middle spectrum is blueshifted and the bottom spectrum is redshifted.	3
1.3	M31 spectrum obtained by Slipher in 1913. On the top of the image is a reference spectrum, while the M31 spectrum is on the bottom. It is possible to notice the blueshift of the emission lines.	4
1.4	J-PLUS and S-PLUS transmission curves.	6
1.5	Example of use of the drop-out technique to detect a $z = 3$ galaxy.	8
2.1	Simple neural network model. The blue, white, and red circles represent the input, hidden and output layers. The green circles represent the bias neurons.	15
2.2	Illustration of a single perceptron. A, B and C are inputs, w_A , w_B and w_C are their respective weights, b_0 is the bias neuron. In white is the neuron (or the perceptron itself).	16
2.4	Dependency of each neuron activation on the output of previous neurons. In this situation, $x' = f(x)$ and $x'' = g(f(x))$	17
2.3	Illustration of the training process.	18
2.5	Example of a decision tree.	19
2.6	20 random functions drawn from Equation (2.3)	21
2.7	20 random functions drawn from the conditioned prior of Equation (2.3) on the observed data, represented by red dots.	22
2.8	Keras model structure.	26
2.9	Representation of the ReLU function.	26
3.1	Distribution of r-band magnitude and z_{spec}	31
3.2	Number of objects in the training, validation and testing samples for each nDet_auto value.	33
3.3	Percentage of objects with missing information in each band for nDet_auto ≥ 8	33
3.4	Representation of a 4-fold cross validation.	36
4.1	Missing feature analysis results for the scatter (σ_{NMAD}) for all the ML codes	39
4.2	Missing feature analysis results for the bias (μ) for all ML codes	41
4.3	Missing feature analysis results for the outlier fraction (η) for all methods	43

4.4	Comparison of single-point estimate results for samples with different features . .	47
4.5	Comparison of single-point estimate results for all methods	49
4.6	Photometric redshift density plot of BPZ.	50
4.7	Comparison of all methods in bins of redshift and magnitude.	52
4.8	Loss as a function of iteration for GPz and Deep Learning models created with Keras.	53
4.9	Mean PDFs as a function of r-band magnitude and redshifts for ANNz2.	55
4.10	Distribution of Odds values for ANNz2.	56
4.11	Mean PDFs as a function of r-band magnitude and redshifts for GPz.	57
4.12	Distribution of Odds values for GPz.	57
4.13	Mean PDFs as a function of r-band magnitude and redshifts for Deep Learning. .	59
4.14	Distribution of Odds values for Deep Learning.	60
4.15	Distribution of spectroscopic and photometric redshifts. Each bin is coloured by the Odds value.	62
A.1	Distribution of the training and testing samples	85
A.2	Comparison of Odds distribution for samples with different input features using Deep Learning.	86
A.3	Single-point estimate analysis results for ANNz2.	88
A.4	ANNz2 photometric redshift density plot.	89
A.5	Single-point estimate analysis results for GPz.	90
A.6	GPz photometric redshift density plot.	91
A.7	Single-point estimate analysis results for Deep Learning.	92
A.8	Deep Learning photometric redshift density plot.	93
A.9	Number of objects in each two-dimensional bin.	94
A.10	ANNz2 results for scatter when analysing redshift and magnitude bins simultaneously.	95
A.11	ANNz2 results for bias when analysing redshift and magnitude bins simultaneously. .	96
A.12	ANNz2 results for outlier fraction when analysing redshift and magnitude bins simultaneously.	96
A.13	GPz results for scatter when analysing redshift and magnitude bins simultaneously. .	97
A.14	GPz results for bias when analysing redshift and magnitude bins simultaneously. .	98
A.15	GPz results for outlier fraction when analysing redshift and magnitude bins simultaneously.	98
A.16	Deep Learning results for scatter when analysing redshift and magnitude bins simultaneously.	99

A.17 Deep Learning results for bias when analysing redshift and magnitude bins simultaneously.	100
A.18 Deep Learning results for outlier fraction when analysing redshift and magnitude bins simultaneously.	100
A.19 BPZ results for scatter when analysing redshift and magnitude bins simultaneously.	101
A.20 BPZ results for bias when analysing redshift and magnitude bins simultaneously.	102
A.21 BPZ results for outlier fraction when analysing redshift and magnitude bins simultaneously.	102
A.22 Gaussian fit to ANNz2 average PDFs.	104
A.23 Gaussian fit to GPz average PDFs.	105
A.24 Gaussian fit to Deep Learning average PDFs.	105
A.25 Comparison of the PDFs for a subset of the testing sample. The r-band magnitude is shown on top of each plot.	106

List of Tables

2.1	Redshift and corresponding bin value for a subset of the training sample.	27
3.1	Parameters, respective surveys, values used for restriction and the motivation. . .	32
3.2	Number of objects in the training, validation and testing samples for each nDet_auto value considered.	34
4.1	Metrics, denotations and definitions that are used to analyse the results of this work.	37
4.2	Final results obtained for all metrics studied for each method presented and for all nDet_auto values.	44
4.3	Name of samples, description and the input features considered.	46
4.4	Results for Deep Learning using the samples described in Table 4.3.	46
4.5	Results for each method for nDet_auto ≥ 8	51
4.6	Percentage of objects with z_{spec} within 1, 3 and 5 times Δz of the peak using the nDet_auto ≥ 8 results.	61
4.7	Results of Gaussian fit to averaged PDFs in r-band magnitude bins.	62
4.8	Results of Gaussian fit to averaged PDFs in redshift bins.	63
A.1	Parameters chosen to be used during the training and optimization procedures of ANNz2.	81
A.2	Parameters chosen to be used during the training procedure of GPz.	82
A.3	Parameters chosen to be used during the training procedure of Keras.	84
A.4	Final results obtained for all metrics studied for each method presented and for all nDet_auto values.	103

List of Abbreviations

z_{phot}, photo-z	Photometric redshift	7
SED	Spectral Energy Distribution	7
TF	Template Fitting	9
PDF	Probability Density Function	9
ML	Machine Learning	10
ANN	Artificial Neural Network	14
CNN	Convolutional Neural Network	14
RNN	Recurrent Neural Network.	14
MLP	Multi-Layer Perceptron	14
GP	Gaussian Processes.	19
MLM	Machine Learning Method.	23

1 Introduction

Since the beginning of our history, the human beings were amazed by the night sky and all mysteries it hides. Due to our innate curiosity, the old civilizations developed means to study the universe, making astronomy the oldest natural science (Krupp, 1994), and some of the projects that our scientific ancestors engineered can be considered advanced even to modern day standards, such as the Antikythera mechanism¹ (Freeth, 2002).

The ancient cultures did not have the same vision as we have today about this science, often assigning stars, planets and celestial occurrences to divinities and supernatural elements, in a way to justify the phenomena in the world around them, such as rain, thunder, the seasons and much more (Selin, 2012). This knowledge should not be belittled, since it gave origin to cultural products we use even today, such as the lunar calendar which, although more commonly seen in oriental cultures, had a great impact in the human agriculture and originated the twelve month calendar based on the solar year used nowadays.

Skipping a great part of the development of astronomy in different civilizations we get to the XVII century, which marked the birth of the instrument that is a symbol to astronomy, the telescope. Being first patented by the Dutch eyeglass maker Hans Lippershey in 1608 (King, 1955), this essential instrument, just one year later, in 1609, served as inspiration to Gallileo Galilei, who built his own design and peered into the cosmos. The audacity of Gallileo led him to discover the rings of Saturn, moons of Jupiter and the sunspots, and this knowledge led him to believe that Nicolaus Copernicus' heliocentric model was correct in contrast to the leading geocentric model of that time.

In these four centuries, telescopes developed in an extremely fast manner, reaching sizes unimaginable to its first creators. And through these giant “eyes”, we have an unprecedented view of our universe, leading to discoveries which allowed us to know, although not completely, our place in the cosmos.

Not only the telescopes became technologically advanced. New techniques were created to observe different wavelengths, from radio to gamma-rays. Telescopes also reached space (with the Hubble Space Telescope as a very successful example²), allowing studies in wavelengths that are absorbed by the atmosphere, such as in the ultraviolet and infrared, and becoming immune to its turbulent effects. Also, new procedures were developed to make corrections for these same effects on ground-based equipments (Tarengi and Wilson, 1989).

We learned how to build giant mirrors, reaching more than eight meters, and how to make a combination of them to form an even bigger telescope (Bernstein et al., 2014). And the optical

¹<http://www.antikythera-mechanism.gr/>

²https://www.nasa.gov/mission_pages/hubble/story/index.html

filter systems allow us to probe different aspects of the night sky, be it star-forming regions (James et al., 2004; Hodge et al., 2010) or the distribution of hydrogen gas in space thousands of light years away from the Earth (Laine and Gottesman, 1998; Serra et al., 2019).

1.1 | Contemporary Surveys

Astronomical surveys are essential to deepen our knowledge about the universe. We can cite, for example, the “The revised 3C catalogue of radio sources” created by Bennett (1962), who helped the discussion between Fred Hoyle, Hermann Bondi and Thomas Gold (Hoyle, 1948) and their contemporaneous astrophysicists on whether or not our universe was static or in expansion.

We can also cite the IRAS survey (Neugebauer et al., 1984), the first to probe all the sky in the mid-infrared, the ROSAT survey (Voges, 1993), responsible for the first mapping of the sky in X-rays and the VLA-FIRST survey (Becker et al., 1994), projected to study the sky in 21 cm, among others.

In the 19th century, during the development of ondulatory physics, the mathematician and physicist Christian Doppler noticed that the frequency of sound waves changed according to the relative velocity between the source and the observer. This is the effect responsible for the change in pitch of the siren of an ambulance when it passes by, for example. A few years later, in 1848, the French physicist Hippolyte Fizeau discovered that the same phenomena could happen with light, being responsible for changing the frequency (and consequently the colour) of a moving light source. This effect was named as “Doppler-Fizeau Effect” (illustrated in Figure 1.1).

Figure 1.1: Illustration of the Doppler-Fizeau effect.

This effect is responsible for the shift in the spectra of extragalactic sources, which we can obtain by passing the light received from those sources through a prism, for example. A simplified illustration of spectra is presented in Figure 1.2. In this figure, the black lines represent the

absorption lines which are originated from atomic transitions of the chemical elements forming the light observed and are well known. When a light source is moving towards the observer, these lines will be shifted to higher frequencies (to the blue part of the spectrum) and it is said that the source is blueshifted. If the source is moving away, then these lines are shifted to lower frequencies (to the red part of the spectrum) and it is said that the source is redshifted.

Figure 1.2: Simplified representation of spectra. The black lines represent absorption lines. The top spectrum is in the rest-frame (i.e. a lab), the middle spectrum is blueshifted and the bottom spectrum is redshifted.

Comparing the wavelength of emission or absorption lines in these shifted spectra to a rest-frame spectrum, it is possible to quantify this shift using the simple mathematical relation of Equation (1.1).

$$z = \frac{\lambda_{\text{obs}}}{\lambda_{\text{rest}}} - 1 \quad (1.1)$$

where z is the shift measurement (spectral deviation), λ_{obs} is the observed wavelength of a given line, seen on the spectrum of the object, and λ_{rest} is the wavelength of said line in the rest-frame (i.e. a lab).

In the context of extragalactic astronomy, the first spectral shift measurement was made by [Slipher \(1913\)](#), who discovered that the lines in the spectrum of Andromeda (M31, Figure 1.3) had a shift towards the blue (blueshift, $z < 0$). Subsequently, he began a study to measure the shift in the spectra of different galaxies, discovering that most had lines shifted towards the red (redshift, $z > 0$). Using the Doppler-Fizeau equation he concluded that M31 had one component of its velocity in our direction, while the redshifted galaxies had components of their motion moving away from the Milky Way.

Figure 1.3: M31 spectrum obtained by Slipher in 1913. On the top of the image is a reference spectrum, while the M31 spectrum is on the bottom. It is possible to notice the blueshift of the emission lines.

1.2 | Distance measurements

The measurement of distances in the universe uses different techniques for different scales, from objects in the Solar System to structures beyond galaxy clusters. These techniques compose what is known as the “Cosmic Distance Ladder”. Some of these techniques are:

- Radar measurements: used to determine distances to objects in the Solar System by sending pulses of electromagnetic waves (in radio) to an object and timing how long it takes to echo back to Earth;
- Parallax: used to measure distances to stars in the Solar neighbourhood. This technique is based on geometry, and it is possible to determine distances to these stars by comparing their position to a background object;
- Distance Modulus: Used to find distances beyond 500 light-years (the capability of parallax). It is based on the comparison of the brightness and luminosity of a given object.
- Cepheid Stars: This kind of star pulsates in a regular way, and the relation between the periods of these pulses and the luminosity of the stars are known. With this, one can use the distance modulus to find how far a given object is.
- Tully-Fisher Relation: [Tully et al. \(1975\)](#) discovered a correlation between the luminosity of spiral galaxies and their rotation speed. Once this speed is known, one can find the luminosity of the galaxy and compare it with the apparent magnitude, and thus calculate its distance.

For celestial objects further away than local clusters, one can use the Hubble-Lemaître Law (Hubble, 1929; Lemaître, 1927). This law establishes a relation between the spectral deviation z from Equation (1.1) and the distance to celestial objects. This relation is shown in Equation (1.2) and is valid for the $v_{\text{rad}} \ll c$ regime. If $v_{\text{rad}} \sim c$, then a relativistic correction needs to be applied.

$$z \simeq \frac{v_{\text{rad}}}{c} = \frac{H_0 \times D}{c}, \quad v_{\text{rad}} \ll c \quad (1.2)$$

where H_0 is the Hubble constant and D is the distance to the object.

Given the importance of precise measurements of the spectral shift, which can be used to constrain cosmological parameters, probe the evolution and formation of the galaxies as a function of distance (and consequently, age) and study the large-scale structure of the universe, several projects were developed to obtain this information. Some projects of this kind are the Las Campanas Redshift Survey (Shectman et al., 1996), CfA2 Redshift Survey (Falco et al., 1999), 2-degree Field Galaxy Redshift Survey (Colless et al., 2001), DEEP2 Redshift survey (Davis et al., 2003), VIMOS VLT Deep Survey (Le Fèvre et al., 2005), 2-degree Field SDSS and QSO survey (Cannon et al., 2006), zCOSMOS (Lilly et al., 2007), WiggleZ (Blake et al., 2008), 6-degree Field Galaxy Survey (Jones et al., 2009), Galaxy And Mass Assembly survey (Baldry et al., 2010), the 2MASS Redshift Survey (Huchra et al., 2012), and the Sloan Digital Sky Survey (Strauss et al., 2002; Eisenstein et al., 2001; Baldry et al., 2005). All these surveys, when combined, provided spectroscopic redshifts for a little less than two million objects, making evident our efforts to measure distances in the universe.

Two additional noteworthy spectroscopic survey projects that are planned to begin operations in 2020 are the Prime Focus Spectrograph (PFS, Tamura et al. 2016) and Dark Energy Spectroscopic Instrument (DESI, Levi et al. 2019). The first will use optical fibres to obtain 2400 spectra simultaneously in the blue, visible and near-infrared. The second will measure the position and distance of about 40 million galaxies. Both projects aim to study the nature of dark matter and dark energy.

There is also a second class of surveys, called photometric. This kind of project can estimate parameters of celestial objects using their magnitudes or fluxes in different filters instead of using spectra. This approach has the capability of acquiring information on thousands of objects at the same time and can reach fainter magnitudes than spectroscopic surveys. An example of this class is the recently concluded Dark Energy Survey (Sánchez, 2010) that probed more than 300 million objects among galaxies and supernovae to study patterns in the large-scale structure of the universe with the objective of investigating the nature of dark energy.

With the recent developments in computer technology (both in computational speed, storage capability and data transfer speed), data science and engineering, we were able to build projects more ambitious than the ones before. A great example is the Large Synoptic Survey Telescope (LSST Science Collaboration et al., 2017), which aims to obtain information about nearly 40 billion stars and galaxies with several science goals planned, from small asteroids in the solar system, the formation and evolution of the Milky Way and discoveries of transient phenomena, to insights about the nature of dark matter and dark energy. Another great example is the Square

Kilometre Array Survey, which is pushing today's technology to its limits, both as in storage capacity, data processing speed and data transfer rates (Lazio, 2009).

In addition to these surveys, the Javalambre Physics of the Accelerating Universe Astrophysical Survey (J-PAS, Benítez et al. 2014) will present the most advanced filter system to date, composed of 59 narrow and broad-band filters in the optical range of the electromagnetic spectrum, providing the same precision as a low-resolution spectroscopic survey. To provide the photometric calibration to J-PAS, the Javalambre Photometric Local Universe Survey (J-PLUS, Cenarro et al. 2019) was designed with a twelve filter system in the optical, providing information that can be used to study the large scale structure of the universe, galaxy clusters, stellar populations, quasars and supernovae in the nearby universe.

With the objective of executing a high precision photometric survey in the Southern Hemisphere, the Southern Photometric Local Universe Survey (S-PLUS, Mendes de Oliveira et al. 2019) was developed. This survey will probe 9000 square degrees of the Southern sky, using a photometric system identical to J-PLUS, ranging from 3700 Å to 9000 Å. The project is being run at Cerro Tololo, Chile, using a robotic telescope with a 0.80 meter telescope, equipped with a 1.4 square degrees field-of-view camera composed by a $9.2k \times 9.2k$ CCD with 0.55 arcsec/pixel scale. The photometric system has 5 broad-band filters (u, g, r, i, z) and 7 narrow-band filters (J0378, J0395, J0410, J0430, J0515, J0660, J0861). Each narrow-band filter was designed to explore important characteristic features of the local universe, such as [OII] 3727 Å, Ca H+K, H δ , G-band, Mgb triplet, H α and the Ca triplet, respectively (Gruel et al., 2012; Marín-Franch et al., 2012). The transmission curves of these filters are displayed in Figure 1.4.

The unique photometric system configuration of the J-PAS, J-PLUS and S-PLUS surveys

Figure 1.4: The transmission curves for the filters composing the photometric system of J-PLUS and S-PLUS.

provide a great opportunity to obtain high accuracy photometric redshifts that can be applied to a large range of problems in extragalactic astronomy.

1.3 | Photometric Redshifts

As noticed previously and exemplified with the S-PLUS, J-PLUS, J-PAS and LSST surveys, the size and scope of astronomical surveys are reaching extremely large scales, probing an immense number of objects and making astronomy enter the so called “Big Data” era, in which astronomers have to deal with immense amount of data, with catalogues containing billions of objects, number that will increase exponentially in the next years (Mickaelian, 2016). This characteristic that will allow us to study the universe in unprecedented ways is the same that makes the acquisition of spectroscopic redshifts for tens or hundreds of million objects impossible. Due to observational constraints, this kind of redshift is only possible for shallow and/or narrow field surveys, and becomes increasingly difficult to determine for faint and/or distant galaxies due to the signal to noise required for a robust ascertainment of the properties of these objects (Salvato et al., 2019).

For this reason, and with the increasing interest on using computers to automate processes, another way to determine redshifts was developed based not on the spectra of the objects, but on their photometry, which can be obtained for large fields at a time and have the capability to go to fainter magnitudes. This kind of estimation received the name of photometric redshifts (z_{phot} , photo-zs).

The first determination of non-spectroscopic redshifts was done by Baum (1957), in an attempt to estimate redshifts for galaxies further than $z \sim 0.2$. In his work, he applied an analysis on the spectral energy distribution (SED) of ten galaxies in five clusters, assuming that the SEDs shift towards the red provided a measurement of the galaxies redshift, and was able to measure the distance of a cluster at $z \sim 0.4$, twice of what would be the spectrographic limit at the time. His results indicated that the universe is nearly flat and not consistent with models in which the universe is strongly curved or static. Although Baum did not name these distance measurements as photometric redshifts (he used the term “photoelectric”), this is also the first known suggestion of the use of SED fitting to estimate redshifts, a technique still used today to estimate photo-zs and currently known as template-fitting.

Using the magnitude obtained in each filter for a given object, or its photometry, it is possible to estimate its distance given that there is a relation between magnitudes and/or colours and the redshift of the object (Masters et al., 2015). This is the basic principle behind the determination of photometric redshifts.

This kind of distance measurement is not exclusive to the last years. Steidel et al. (1996, 1998) obtained the photometric redshift for galaxies with redshifts higher than 3 using a technique

named photometric drop-out. In this method, the distance to galaxies is measured by comparing their fluxes in the bluest bands. An example of the use of this technique is illustrated in Figure 1.5.

For galaxies at high redshifts, the continuum flux break that occurs at the $912\text{\AA}$, named the Lyman Break, is shifted to the optical part of the spectrum. Since the break is characterized by a lack of emission in the blue part of the spectrum of the object, it will only be detected in the redder filter (g, in this case) and will not be detected in any filter to the left of the break.

Figure 1.5: Example of use of the drop-out technique to detect a $z = 3$ galaxy. In this situation the galaxy will be detected in the G filter, but not in the U filter. Adapted from [Adelberger et al. \(2004\)](#).

The main advantage of photometric redshifts is the capability of giving distance measurements for a large number of sources, at the cost of lower precision when compared to spectroscopic redshifts. Photo-zs have been used for several applications, such as the evolution of the luminosity function of galaxies ([Ramos et al., 2011](#)), the effects of environment in galaxy evolution ([Cucciati et al., 2016](#)), group/cluster membership determination ([Pelló et al., 2009](#)), search for galaxy clusters ([Almeida et al., 2019](#)), search for high redshift quasars ([Nakazono et al., 2019](#)), weak lensing ([Mandelbaum et al., 2008](#); [Petri et al., 2016](#)) and to trace the large scale filamentary structure of the universe ([Kruuse et al., 2019](#)).

Nowadays there are two main methods of determining photometric redshifts, one based on template-fitting and the other on machine learning.

1.4 | Photometric Redshifts with Template-Fitting

Template Fitting (TF) methods work by comparing the data-points obtained from photometry (one for each filter, representing the SED) of a given object with a template library containing synthetic or empirical spectra, which can be redshifted or not, for galaxies with different stellar populations, such as ellipticals, spirals, star bursting and AGNs, among others.

Using some sort of optimization metric, such as a χ^2 minimization (methodology first used by [Puschell et al. 1982](#)), one can find the most probable spectra and redshift needed to fit the SED of an object, thus determining its photometric redshift. A characteristic of this kind of method is the generation of probability density functions (PDFs) natively, which is originated from the uncertainty of the SED fit to models, and allows us to access the uncertainties related to the determination of the photo-zs themselves by providing a distribution of their possible values.

A number of codes were developed using this approach, such as Photometric Analysis for Redshift Estimate (LePhare, [Arnouts et al. 1999](#); [Ilbert et al. 2006](#)) which uses a direct χ^2 minimization, Bayesian Photometric Redshifts (BPZ, [Benítez 2000](#)) that uses Bayesian Statistics and can use prior information, Hyperz ([Bolzonella et al., 2000](#)) optimized for performing photometric redshift estimation on large catalogues, the Zurich Extragalactic Bayesian Redshift Analyzer (ZEBRA, [Feldmann et al. 2006](#)) capable of automatically correcting the templates to better represent the SED of real galaxies, Easy and Accurate Z (EAZY, [Brammer et al. 2008](#)) which iteratively adapts the SED templates based on the residuals of fits to galaxy spectra, making combinations of different templates, GOODZ ([Dahlen et al., 2010](#)), a code developed to obtain photo-zs for the GOODS survey by using a total of 164034 SED templates and a luminosity function prior in the fitting procedure, and BayEsian Analysis of GaLaxy sEds (BEAGLE, [Chevallard and Charlot 2016](#)), which also models stellar radiation.

All the codes require a library of model templates to function, such as the synthetic spectra generated by [Bruzual A. and Charlot \(1993\)](#), [Bruzual and Charlot \(2003\)](#) or the empirical SEDs compiled by [Coleman et al. \(1980\)](#). Often a combination of spectra from different sources and galaxy types is used to improve the photo-z determination.

1.5 | Photometric Redshifts with Machine Learning

Machine Learning (ML) codes are methods based on the use of a training dataset (real or simulated) which should be, ideally, both complete and representative of the sample where the photo-zs need to be extracted from.

The machine learning codes attempt to determine a robust relation between magnitudes and/or colours (input) and the redshift (output). This relation can then be applied to a given galaxy sample for which photometric redshifts are desired.

Machine learning can be applied to both regression and classification problems. In the first case the output is a continuous variable, such as the redshift of an object or a galaxy cluster mass (Armitage et al., 2019). In the second case the output describes a discrete quantity, like the galaxy morphology, as in the first time artificial neural networks were used in astronomy (Storrie-Lombardi et al., 1992).

The codes that use this approach can follow different paths. Examples include Trees for Photo-Z (TPZ, Carrasco Kind and Brunner 2013), photometric redshifts using Artificial Neural Networks (ANNz, Firth et al. 2003; Collister and Lahav 2004), ANNz 2.0 (Sadeh et al., 2016), deep learning codes made with Keras (Chollet et al., 2015), Gaussian Process for photometric redshift estimation (GPz, Almosallam et al. 2016), generalized linear models with CosmoPhotoZ (Elliott et al., 2015), and others.

In addition to single-point estimates, and similarly to TF algorithms, ML codes can also provide probability distribution functions by using different methods. ANNz2.0 combines different trained models to generate PDFs. GPz considers the PDF as Gaussian distributions centred on z_{phot} with a variance equal to the model uncertainty. PDFs can also be implemented in Keras deep learning models using a particular activation function for the last layer which provides a probability of a certain object having different photo-zs.

Probability distribution functions have received increasing interest in astronomy and cosmology research in the past years. PDFs provide the necessary tools to probe the uncertainties contained in the photo-z determination and within the methods themselves (Wittman et al., 2016).

Other studies in astronomy also take advantage of machine learning methods, such as exoplanet detection (Pearson et al., 2018; Schanche et al., 2019), supernovae classification (Vargas dos Santos et al., 2019) and classification of variable stars (Mahabal et al., 2017), to cite a few.

1.6 | Template-fitting or Machine learning?

Both methods have their own advantages and disadvantages. Template-fitting has the capability of generating PDFs by default, allowing us to verify the uncertainty on the results, and can provide, besides the redshift estimation, information on the galaxy stellar populations by adjusting the SED with a template library of different stellar population models (Molino et al., 2019). Also, given the nature of this method, it is possible to extrapolate templates into other regions of the spectra in order to improve results (Gorecki et al., 2014) or to fainter magnitudes, beyond the limit of the spectroscopic sample.

Machine learning outperforms template-fitting when the training sample available has a reasonable number of objects that are both complete and representative of the target sample (Bilicki et al., 2018). This approach is often faster than template-fitting and optimized to work with extremely large datasets, being able to accommodate the data that are or will be generated from the current and future big surveys. Also, machine learning codes can use non-photometric data, such as surface brightness (Kurtz et al., 2007) and morphological types (Soo et al., 2018).

1.7 | Objectives

The goal of this work is to take advantage of both recent developments in machine learning techniques and the innovative Javalambre filter system used by S-PLUS, optimized for the local universe, to obtain accurate photometric redshifts.

Given the nature of the ML codes, different methods can lead to different results on the same training sample, so a comparison is done between three different ML codes and the one used actually on S-PLUS for the data release 1, the template-fitting based BPZ (Molino et al., 2019).

Although all codes used in this work are machine learning techniques, they work using different methods. GPz (Almosallam et al., 2016) is built upon the theory of Gaussian Processes, ANNz2 (Sadeh et al., 2016) can use both boosted decision/regression trees and artificial neural networks, and the latter is also the focus of the deep-learning library Keras (Chollet et al., 2015).

This dissertation is organized as follows: in chapter 2 an introduction to the three different machine learning approaches (artificial neural networks, boosted decision/regression trees and Gaussian processes) is made. In section 2.4 the three algorithms (ANNz2, GPz and Deep Learning) are presented. In chapter 3, the data used and the pre-processing steps are described. The metrics used in this work to evaluate the quality of the photo-zs are shown in section 4.1. Then, in section 4.2, the results considering samples with different number of missing bands are anal-

ysed. Section 4.3 has a discussion regarding the choice of input parameters on the performance of the Deep Learning algorithm. In section 4.4 the single-point estimate analysis is done and section 4.6 has a discussion regarding the probability distribution functions. The conclusions obtained from this work and the future perspectives are presented in chapter 5.

In the Appendix, section A.1 contains the CasJobs query used to obtain the SDSS catalogue used in this work. Section A.2 has the parameters used for each machine learning method considered in this text. The Odds distribution for the samples analysed in section 4.3 are presented in section A.4. Also, the single-point estimate results specific to each method are shown in A.5 and the analysis considering both r-band magnitude and redshift simultaneously for each method is presented in A.6. The results considering all objects (not bins) for all methods are exhibited in section A.7, additional plots for the PDF analysis are presented in A.8 and the comparison of the PDFs obtained from the machine learning methods for a subset of the testing sample is shown in A.9.

2 Methods

In the following sections, a general introduction will be given for each machine learning approach adopted in this work, with details about how each method works being presented afterwards.

2.1 | Artificial Neural Networks

2.1.1 | History

The start of studies related to the creation of artificial neural networks we use today began in the early 40s. In 1943, the neurophysiologist Warren McCulloch and the mathematician Walter Pitts published a paper with a mathematical model of how neurons should work ([McCulloch and Pitts, 1943](#)). This work began a series of studies about the operations that neurons realize and, in 1958, the first Perceptron, the precursor to the multi-layer perceptrons used today, was created by Frank Rosenblatt ([Rosenblatt, 1958](#)). This perceptron was implemented in hardware instead of software, on a machine called “Mark I Perceptron”, and used for the task of image classification.

After a brief period of great development, investment and hype, the research in this area stagnated as funding decreased due to the lost of interest because of the non-fulfilled promises of what neural networks could achieve¹ and criticism originated from the possibility of AI replacing humans in certain tasks. This period of low investment was called the “First AI Winter”, in a reference to a nuclear winter, and lasted until 1980. In the following years, the hope in AI realisations would begin to increase again, creating another wave of hype among the industry and governments which, after the expectations were again not met, cut funding to this area, giving rise to the “Second AI Winter”. Currently we live in an epoch called “AI Spring”, where scientific research and commercial developments in machine learning made funding increase once more, but as opposed to the previous decades, we now have much more powerful tools to train and analyse the models. In the special case of astronomy, there is the legitimate need of studying

¹See http://www.chilton-computing.org.uk/inf/literature/reports/lighthill_report/p001.htm

machine learning techniques since they can help greatly with analysing the large amount of data that will be made available by future surveys.

As already mentioned, one of the oldest developments in this area were Artificial Neural Networks (ANNs), which were inspired by the functioning of biological brains. In our brain, different areas are responsible for processing different features, such as our motion, behaviour and thought. Each area is composed of billions of nerve cells called neurons, which are interconnected via dendrites, responsible for receiving electric signals from other neurons, causing an activation response leading to some action, like reading, for example.

A computer artificial neural network works in a similar way. First, the features of the training sample (images of objects, for example) are fed to an input layer, then this information is passed to one or more hidden layers responsible for abstracting and processing (finding patterns and edges in the input images for example) and to an output layer, which makes an estimation of the target variable of the study (like the name of the object in the image).

ANNs can be divided in two architectures regarding the type of learning: unsupervised learning and supervised learning. In the former, the training sample used does not contain a target, so all work is done by the model itself. An example of unsupervised learning are the Self-Organizing Maps, which can be used to find patterns and help the visualization of multi-dimensional data, due to its capacity of representing a N-dimensional dataset into a human-readable two-dimensional space, for example. In the latter case, the supervised learning, the network is trained using a sample which contains the “correct answer” for the target variable. Additionally, supervised learning neural networks can be divided in three different architectures:

- Convolutional Neural Networks (CNNs), mostly used to classify or make regression using images as input due to its invariance to translation nature, but can also be used with tabular data;
- Recurrent Neural Networks (RNNs), which is a type of network capable of using previously seen information to predict subsequent events and mostly used for text generation and time-series prediction;
- Multi-Layer Perceptrons (MLPs), a development of the first Perceptron created in 1958, which is generally used to classify or predict values using tabular data as input.

2.1.2 | The Multi Layer Perceptron

Since the network used in this research is a MLP, further details will be given for this specific approach. A multi layer perceptron is also known as a feed-forward neural network.

The general form of a neural network structure is given in Figure 2.1. Each circle represents a neuron (or unit), which are distributed in layers, and each neuron in each layer is connected to

Figure 2.1: Simple neural network model. The blue, white, and red circles represent the input, hidden and output layers. The green circles represent the bias neurons.

every other unit in the previous or following layer (a dense configuration). The strength of these connections (represented as lines) are related to a set of weights (w_n), and each connection has its own value for their weights. This liberty on the choice of weights allows each different input to have a different impact on the prediction values.

The first layer, displayed in blue in Figure 2.1, is the input layer. This layer is different from all others in the network since it is only responsible for introducing the input data for further processing by the model. For this reason, it can be considered “passive”, meaning that they do not process the data in any way, and just propagates it along the network in a process called feed-forward.

After the input layer, there are the hidden layers. A neural network can have one or more hidden layers, whose purpose is similar to the groups of neurons in biological brains, being responsible for taking in information from previous layers, processing this information via an activation function and providing an output to the following layers.

This process can be explained using Figure 2.2. In this figure, the inputs A, B and C are multiplied by their respective weights w_A , w_B , and w_C , added the bias neuron value b_0 and fed to the neuron (in white). The sum of the results are then subjected to an activation function (such as the Rectified Linear Unit, in this example) and the result of this function is used as input to the neurons in the following layer. This process is then repeated for all units in all layers of the network.

The “bias neuron” is connected to all others in the same layer, but not to those in the previous or following layer, and its only goal is to add a constant value to the output of each neuron. This unit is responsible for shifting the activation functions, improving the model fitness to data while

Figure 2.2: Illustration of a single perceptron. A , B and C are inputs, w_A , w_B and w_C are their respective weights, b_0 is the bias neuron. In white is the neuron (or the perceptron itself).

offering additional degrees of freedom to the network. In some cases this unit is not needed, such as when applying a normalization process called “batch normalization” in the network, procedure used in this study. The batch normalization acts on the outputs of each layer by subtracting their mean and dividing by the standard deviation and, for this reason, the addition of a constant value will not change the results, thus the bias neuron is not needed.

In the first iteration, since the network has not began the optimization process, the weights of the network are randomly determined by the “initializer”, which can be defined in different ways. It is possible to initialize these weights using constant values or by sampling distributions.

After the first iteration, the weights are optimized iteratively during the training phase by a process called back-propagation, where the model determines how each weight in the network contributes to the overall error. This is done via partial derivatives of the loss value (a measurement of how far the prediction is from the real values) with respect to the weights. With this procedure, the network can perform the weight optimization process efficiently so the model can establish a reliable relation between the input parameters and the desired output parameters.

Finally, the last layer (output), has the purpose of producing a prediction for the target variable. This is done by taking the neuron values of the last hidden layer, multiplying these values by their respective weights and summing the results, just like any perceptron in the previous hidden layers.

As can be noticed, an important aspect of neural networks is the capability to deal with non-linear problems. This characteristic arises due to the use of non linear activation functions, which describe the response of a neuron to the information it receives. Some examples of non-linear activations include the Rectified Linear Unit (ReLU), sigmoid, hyperbolic tangent, and softmax functions.

Although we want to introduce non-linearity in the models to increase its capability of dealing with complex problems, linear activations are still important in some situations. Usually the input layer neurons have linear activations.

If the last layer has a linear activation, then the prediction is just the result of the sum. Otherwise, if the last layer has a different activation, such as softmax or sigmoid functions, then the activation function will be applied to the sum result and this produces the output.

As described above, several processes occurs during the training of the model. The information enters the network through the input layer, is transformed and processed in the hidden layers using activation functions and then produces an output. The following section describes the training processes in more detail.

2.1.3 | The training process

The training process of a network can be described as a loop, as can be seen in Figure 2.3, and it proceeds via the optimization of the weights w_n , which determines the relation between the inputs and the outputs. This optimization task is done by a process called back-propagation.

Back-propagation was introduced in the 1960s and 70s, and the first application of this method as we use today dates back to a master student thesis written by Linnainmaa (1970).

At the first iteration, the model will make a prediction of the target using the training sample and, since the weights are initially random, this prediction will have a high error. This output is then compared, using a loss function, to the true values of the target variable for the same sample, and this comparison will result in a loss score. After that, an optimizer function will adjust the weights so to minimize this score. As shown in Figure 2.3, this process occurs in a loop, iteratively, until a certain accuracy or other convergence criteria is reached.

This adjustment of weights done at the end of the training epoch comes from the back-propagation algorithm. In this case, the model calculates how each weight w_n affects the loss score using partial derivatives. The reason to use partial derivatives is that, apart from the first hidden layer, the output of all neurons are dependent on the output of the previous neurons, as can be seen in Figure 2.4. In this situation, $x' = f(x)$ and $x'' = g(f(x))$, so the final output can be written as $\text{Output} = h(g(f(x)))$.

Figure 2.4: Dependency of each neuron activation on the output of previous neurons. In this situation, $x' = f(x)$ and $x'' = g(f(x))$.

Figure 2.3: Illustration of the training process.

2.2 | Boosted Decision/Regression Trees (BDTs)

A decision/regression tree is a method which origin can be dated back to the 1930s, but the first known application was by [Belson \(1959\)](#). This approach can predict a given class (or value) target based on tests done on the input sample. These tests are splits that, depending on the algorithm, can be made randomly or in an optimized manner on different input variables and their values. One advantage of this approach is that it creates a decision graph, such as that represented in Figure 2.5, which is human readable and can be easily understood.

In Figure 2.5, generated with data used in this work, the topmost node is called the “root” of the tree. All samples are subject to a given condition (g-W2 colour, for example) and this splits the tree in two “branches”. Each branch has its own further splits and, at the end of the branch, a “leaf” node is created. The creation of leaf nodes is subject to a certain convergence criteria, such as the error value or the number of objects in each split.

The name “boosted” arises due to a methodology of combining several individual, simple trees (called “weak learners”), weighted based on their accuracy, into one “strong classifier” in order to reduce overfitting problems and increase precision. When combined, the ensemble is called a “random forest”.

The regression of values using BDTs is done by submitting a given sample to a trained tree. This object will be subjected to the split conditions in this tree until it falls in a leaf node, which will give an individual estimate of the target variable for this object (the redshift, for example). This process is repeated for all trees, and the final combined output is taken as the mean of the individual output of all trees.

Figure 2.5: Decision tree generated using the Python package *scikit-learn* for illustration purposes.

2.3 | Gaussian Processes

Gaussian Processes (GPs) are a stochastic, non-parametric and Bayesian approach to regression problems.

2.3.1 | Weight-space view

Usually, we can assume that, in a regression problem, the relation between an output y and the inputs x is given by Equation (2.1):

$$y = f(x|w) + \epsilon, \quad (2.1)$$

where $f(x|w)$ is a function of the inputs given the model parameters w and ϵ is a noise contribution. In this situation, the function $f(x|w)$ is known.

In this case, the Bayesian approach is to specify a prior distribution for the parameters w ,

and update this distribution based on observations by using the Bayes' rule:

$$p(w|y, x) = \frac{p(y|x, w) \times p(w)}{p(y|x)}, \quad (2.2)$$

where $p(w|y, x)$ is the updated posterior distribution of w given the observations y and data x , $p(y|x, w)$ is the likelihood, $p(w)$ is the prior, and $p(y|x)$ is the marginal likelihood or evidence.

The prediction of a new data point x_* can be calculated by averaging all predictions made by each function in the distribution at that point, weighted by their posterior distribution.

2.3.2 | Function-space view

If the function $f(x|w)$ is not known we can, instead of estimating the parameters w , sample f from some distribution of functions. This is called the “function-space view” ([Rasmussen and Williams, 2005](#)).

In this situation the objective is to obtain a probability distribution of all possible functions which can map the inputs to the outputs. This is opposed to the previous case where the aim is to obtain a probability distribution of the parameters w of the known function f .

It can be shown that, under very general conditions, f is a random variable which is distributed as a Gaussian Process, described by Equation (2.3):

$$f(x) \sim \mathcal{GP}(\mu(x), \kappa(x, x')), \quad (2.3)$$

where $\mu(x)$ is the GP mean and $\kappa(x, x')$ is the GP covariance (or kernel). The kernel works as a “similarity” measurement of the inputs, and can be described in several ways. A common kernel is the radial basis function given by (2.4), which ensures that the functions f will be smooth:

$$\kappa(x, x') = \sigma_f^2 \times \exp\left(-\frac{|x - x'|^2}{2\lambda^2}\right), \quad (2.4)$$

where σ_f and λ are two hyperparameters controlling the amplitude and the length-scale. If we sample random functions from this prior, assuming that $\mu(x) = 0$, we get examples as shown in Figure 2.6.

One way to find the distribution of functions which are in agreement to the observations is by randomly sampling functions from this prior and reject those that does not fit the data, but this is inefficient. The Bayesian approach offers a much more effective way of selecting these functions based on observations. Analogous to Equation (2.2), we have:

$$p(f|y, x) = \frac{p(y|x, f) \times p(f|x)}{p(y|x)}, \quad (2.5)$$

Figure 2.6: 20 random functions drawn from Equation (2.3)

and it is possible to select only functions that adequately fits the data by conditioning the prior distribution $p(f|x)$ on a number of observations x_* . Thus, a number of functions f_* which are in agreement with the data can be sampled, resulting in Figure 2.7.

The advantages of Gaussian Processes are that, since it is non-parametric and finds the distribution of different functions which are consistent with the observations, it can estimate the outputs y without assuming the form of these functions. It can also provide an estimate of the uncertainty of predictions. One disadvantage, as can be seen in Equations (2.3) and (2.4), is that the size of the GP covariance matrix κ grows quadratically with the number of objects in a dataset. This imposes computational restrictions when using this method for large samples.

Figure 2.7: 20 random functions drawn from the conditioned prior of Equation (2.3) on the observed data, represented by red dots.

2.4 | Softwares used in this work

In this section, a brief review of the three different algorithms we used will be given.

2.4.1 | Artificial Neural Networks for photometric redshifts (ANNz2)

ANNz2 is a code developed by [Sadeh et al. \(2016\)](#) with the purpose of being the successor of the homonymous code ANNz ([Collister and Lahav, 2004](#)). The main difference is that ANNz2 offers PDF solutions natively (without the need of modifications in the code) to any regression or classification problem.

It should be noticed that, although the code has “Artificial Neural Networks” in the name, it works using several different machine learning approaches based on the Toolkit for Multivariate Data Analysis (TMVA) ([Hocker et al., 2007](#)), which is a project integrated with ROOT ([Brun and Rademakers, 1997](#)). The main options are ANNs and BDTs. For this work, only the boosted decision/regression trees were used due to time constraints, since the training of one ANN model (which encompasses 100 neural network models, as described later in the text) demands a high amount of computer time and power.

The code is written in C++ but the script that the user interacts with to run the algorithm is written in Python.

ANNz2 can do both classification and regression. Since this work is focused on the estimate

of photometric redshifts, we use the latter approach. Within this code are different scripts with different functions, such as “single regression” (for point estimates of photo-zs) and “random regression” (for PDF generation). Since it is desired to obtain the PDFs of the photometric redshifts, a random regression was adopted. In this case, the user can choose different parameters that determine how the algorithm will estimate photo-zs.

The parameters chosen are given in section A.1, but the most important ones are:

- nMLMs: number of machine learning methods (MLMs) to train during the learning process. The recommended number is 100 for science cases according to the ANNz2 authors;
- Target: the target of the regression (redshift in our case);
- Input parameters: the columns of the tabular input data that will be used to establish the relation between input and output.
- Each MLM configuration: the user can choose the parameters of each MLM, such as the method (ANN or BDT), the number of neurons or trees, the type of activation for ANNs, the type of boosting for BDTs, among other settings. In our work, we configured the code so it could randomly generate BDT methods, since they are considerably faster to train than ANNs.

The training process of ANNz2 is composed by four different steps. First, during the “gen-InputTrees” step, the method converts the tabular data into the ROOT TTree Class. Then the training process begins, where a number “nMLMs” of methods with different configurations is generated and trained, and this is the most time-demanding step. An interesting differential aspect of this code is the presence of a bias correction procedure, which is done for PDFs and can be done for each MLM if the user chooses to.

For the MLM bias correction, the procedure is as follows. Consider that only one MLM is trained (nominal-MLM) and it outputs a regression estimator named ANNZ_0. At the end of the training of this nominal-MLM, a smaller, less-complex bias MLM is trained (bias-MLM) using the same inputs as the nominal-MLM but, instead of using the z_{spec} as the target, the bias-MLM training target is the difference between the predicted redshift by the nominal-MLM and the spectroscopic redshift, e.g. $(z_{\text{ANNZ}_0} - z_{\text{spec}})$. Then, the code applies a Kolmogorov-Smirnov test to compare the performance of the bias and nominal MLMs and, if the bias-MLM results in a worse performance it is rejected, otherwise it replaces the nominal-MLM estimator.

For the PDF bias correction, the code calculates the correlation between the target value and the position of the bin for all bins in the PDF. This procedure is done for all objects in the training sample, so that a mean redshift-to-bin correlation can be calculated. This correction is then applied to the PDF by re-weighting each PDF bin by the calculated relation between the photo-z and spec-z.

After the training is done, the code starts the “optimization” step, where the different MLMs are ranked according to their performance (bias, scatter and outlier fractions as defined by the code). In this step, after the best performing MLM is chosen as the single-point estimator, the

PDFs of the solutions are generated. To do so, each remaining MLM is combined with its corresponding uncertainty and are then merged in different ways randomly, but taking into account their ranking. This generates a set of candidate PDFs, which are then compared using a parameter $\mathcal{C}$ (Equation (2.6)) and the set of PDFs that better describe the true redshift distribution is selected:

$$\mathcal{C}(z_{\text{spec}}) = \int_{z_0=0}^{z_{\text{spec}}} p(z) \, dz, \quad (2.6)$$

where $\mathcal{C}$ is the cumulative distribution function for PDFs smaller than z_{spec} , z_0 is the lower bound of the PDF and $p(z)$ represents the differential PDF for a given redshift.

The last step is called “evaluation”, where the code uses the trained models to make the photometric redshift prediction in a given sample.

The parameters chosen for ANNz2 are shown in Table A.3.

2.4.2 | Gaussian Processes for photometric redshifts (GPz)

GPz is a code developed in Python and Matlab languages by [Almosallam et al. \(2016\)](#).

As explained in section 2.3, the complexity of this approach increases quadratically with the number of samples in the training data. Therefore the authors use a different, although similar, approach called “Sparse Gaussian Processes” in order to avoid this problem without reducing the accuracy of the model.

In the sparse approach, the Gaussian Processes are described with a set of basis function models, with specific parameters defined via a set of weights. This way one can assume that, instead of dealing with a covariance matrix κ with dimensions $n \times n$, where n is the number of samples in the dataset, one can use a number $m \ll n$ for the same purpose.

Each of these m radial basis functions are optimized to represent the variable densities and patterns of the data while also accounting for its distribution.

Another feature implemented in this code is the concept of “cost-sensitive learning” (CSL), where different samples have different influences in the final model results. In this work, a normalized CSL method is applied, where the weighting scheme is given by:

$$w_{\text{obj}} = \left(\frac{1}{1 + z_{\text{spec,obj}}} \right)^2. \quad (2.7)$$

Using this approach, objects with lower spectroscopic redshift will have greater weight and thus bigger influence in the model training than those with high redshifts, where data is usually scarce.

Lastly, GPz has the capability of dealing with heteroscedastic uncertainties, i.e. noise that is

dependent on the input features. To do this, [Almosallam et al. \(2016\)](#) models the noise variance as a function of the input, with a relation that is learned during the training process.

The most important parameters that can be chosen by the user are:

- Number of iterations;
- Number of basis functions;
- The type of basis functions;
- If the training should consider the noise (heteroscedastic case);
- The cost-sensitive learning method.

The chosen parameters used in this method are in Table [A.2](#).

2.4.3 | Keras

Keras is an open-source application programming interface (API) written in Python with the objective of providing ease of access to low-level machine learning languages, such as TensorFlow, Theano or CNTK. It was developed to facilitate the interaction between the user and the aforementioned ML codes allowing fast experimentation.

It is based on artificial neural networks and focused on deep-learning, where the number of layers and neurons per layer is higher than those of a “shallow” network, such as the one represented in Figure [2.1](#).

All results obtained from this method were acquired while running the code on a Nvidia K80 GPU provided by the Google Colab² platform.

Numerous models were trained and had their performances compared. The following discussion refers only to the best performing model, illustrated in Figure [2.8](#). The network has a “funnel” shape, getting thinner as it approaches the output layer. This architecture was achieved after testing several other configurations with the Talos software ([Kotila et al., 2019](#)).

Since this algorithm is based on deep-learning using artificial neural networks, its training procedure is the same as described in [2.1.3](#).

Different kinds of activation are available in Keras, and each has a different goal. For this work, all hidden layers are configured to use a Rectified Linear Unit (ReLU) activation, which can be described according to Equation [\(2.8\)](#), and is illustrated in Figure [2.9](#). This choice was made due to the fact that this activation is widely used nowadays in different tasks, has a low computational cost (resulting in faster training) and provided good results in different situations.

²<https://colab.research.google.com>

Figure 2.8: Keras model structure. The input layer is represented in blue, the hidden layers are in white and the two output layers are in red. The output layers are represented as two units for simplicity.

Figure 2.9: Representation of the ReLU function.

$$F(x) = \begin{cases} x & \text{if } x \geq 0, \\ 0 & \text{if } x < 0. \end{cases} \quad (2.8)$$

The two output layers in red have two different activations, two different losses, and give two different predictions. One has a “Linear” activation where $F(x) = x$ and it is only used for performance monitoring tasks. The second output is responsible for generating the probability distribution function of the predictions, which are used for all the analysis shown in Chapter 4.

In order to give more focus to the PDF generation in the network, the two loss have different weights. The “Linear” output uses a Mean Absolute Deviation (MAE) loss with a weight of 0.1, while the PDF output layer uses a Categorical Cross-Entropy loss with weight 0.9.

The PDF calculated using the Deep Learning method is a discrete function. Since the redshift is a continuous variable, it is necessary to convert this quantity into discrete values. To do so the redshift space is split into 200 bins, each with width of 0.0067, spanning from $z_{\text{spec}} = 0$ to $z_{\text{spec}} = 1.34$ (the maximum value of the training sample). After that, each object has its redshift assigned to a certain bin using the “One-hot Encoding” method.

Using this method, a one-hot encoded vector t containing 200 elements is assigned to each object. All values of this vector are zero with the exception of one corresponding to the bin containing the redshift of the object. To exemplify, let us consider a subset of objects with their redshift and corresponding bin value given in Table 2.1.

Table 2.1: Redshift and corresponding bin value for a subset of the training sample.

Object	Redshift	Bin
1	0.2602	38
2	0.2167	32
3	0.2151	31
4	0.1526	22
5	0.0979	14

In this situation, the five objects will have a 200-element one-hot vector filled with zeroes but the 38th, 32th, 31th, 22th and 14th element of this vector will be equal to one for the objects 1, 2, 3, 4 and 5, respectively.

To obtain the PDF, the 200 output neurons of this layer (one for each bin) use the “Softmax” activation, which is described by:

$$S(y_i) = \frac{e^{y_i}}{\sum_j^C e^{y_j}}, \quad (2.9)$$

where the sum is over all possible classes y_j (bins of redshift in our case) and y_i is the probability of the i ’th class.

Using this activation, the exponential of an output value is divided by the sum of exponentials of all possible outputs, and the function converts the possible outputs to probabilities which sum to 1.

As can be seen in Figure 2.3, after the model is trained, a comparison between the predicted and true values is made and produces a loss score. This loss score is associated with a loss function, which is a parameter that the user can set. In this work, since we have two outputs, we have one loss function for each. For the single-point estimation, we use the Mean Absolute Error (MAE) function, since it is more robust to outliers than other functions. It can be easily

defined by:

$$\text{MAE} = \overline{|z_{\text{pred}} - z_{\text{true}}|}, \quad (2.10)$$

where z_{pred} is the predicted output, z_{true} is the real value of the target variable.

For the PDF, the Categorical Cross-Entropy loss is used, and is defined by:

$$\text{CCE} = - \sum_i^C t_i \log(S(y_i)), \quad (2.11)$$

where the sum is over all possible classes (bins of redshift), t_i is the non-zero element of the one-hot encoded vector for an object and $S(y_i)$ is the softmax activation, described by Equation (2.9). Since the one-hot encoding creates a vector where only one element is not zero (as explained earlier), this equation reduces to:

$$\text{CCE} = -\log(S(y_p)) = -\log\left(\frac{e^{y_p}}{\sum_j^C e^{y_j}}\right), \quad (2.12)$$

where the condition $t_i = t_p$ was used in Equation (2.11) and y_p is the true (or positive) class.

Since in this work we have a classification problem with 200 different classes (one for each redshift bin), Equation (2.12) will output a single probability for each bin (or class), constructing the probability distribution function.

For the results shown in this work, the photometric redshifts used are obtained directly from the PDF, using a weighted average of the photo-z bins and their probabilities, given by:

$$z_{\text{phot}} = \frac{z_{\text{bin}} \times p_{\text{bin}}}{\sum p_{\text{bin}}}, \quad (2.13)$$

where z_{bin} is the redshift corresponding to each bin, p_{bin} is the probability of the object pertaining to each bin and the sum is over all values of p_{bin} , which is equal to 1.

Another important aspect of this network is the optimizer function which is responsible for minimizing the error function. In this work, in order to take advantage of recent developments in this area, the Nadam optimizer (Dozat, 2016) was chosen. This optimizer incorporates Nesterov Momentum (Nesterov, 1983) into the Adaptive Momentum (Adam, Kingma and Ba 2014) optimizer.

The full list of parameters used for the Keras model is presented in Table A.3.

3 The Data

The data used in this work is originated from a combination of catalogues. More specifically, the S-PLUS Data Release 1, SDSS Data Release 15 and WISE.

The data is in the Stripe-82 region, ranging from 20h to 4h in right ascension and -1.4° to 1.4° in declination. This region offers a unique opportunity to astronomical studies since it has been surveyed by a number of different projects in different regions of the spectra, from radio (VLA-L, [Hodge et al. 2011](#)), infrared (VISTA-CFHT, [Geach et al. 2017](#)), optical (SDSS, [Jiang et al. 2014](#) and CFHT, [Moraes et al. 2014](#)) to ultraviolet (GALEX, [Niemack et al. 2009](#)) and X-rays (Chandra, [LaMassa et al. 2013a](#) and XMM-Newton, [LaMassa et al. 2013b](#)).

3.1 | Datasets

The S-PLUS Data-Release 1 catalogue ([Mendes de Oliveira et al., 2019](#)) contains photometry on the twelve bands presented in section 1.1 using three different apertures. The “_aper” magnitudes are obtained in a 3 arcsecond aperture, similar to SDSS optical fibres. The “_petro” aperture is defined at the distance where the signal-to-noise of the galaxy drops to zero and the “_auto” magnitudes are total restricted magnitudes, containing most of the light from the galaxy while retaining a high signal-to-noise. For this reason, they are ideal for photometric redshift determinations ([Molino et al., 2017](#)).

The S-PLUS photometric redshifts ([Molino et al., 2019](#)) are obtained using the second version of the template-fitting algorithm BPZ ([Benítez, 2000](#)), which is not publicly available. This code is capable of using a luminosity-based prior to weight the likelihood-function obtained from the comparison of data and templates, and also uses optimized SED models for redshift determination for the local universe. The way template-fitting methods work is briefly explained in section 1.4.

The SDSS - Data Release 15 ([Aguado et al., 2019](#)) is the latest data release of the project, and includes all previous releases. The main addition to the SDSS data in this release comes from the MaNGA survey ([Bundy et al., 2015](#)), designed to obtain spectra for nearby galaxies using integral field units. DR15 also had an update regarding the flux calibration schemes (details can be found on the project website¹). The spectroscopic redshifts and classifications used in this

¹<https://www.sdss.org/dr15/manga/manga-data/whatsnew/>

work are obtained using the “spec1d” pipeline (Bolton et al., 2012). The CasJobs² query used to obtain the data in the Stripe-82 region is presented in Appendix section A.1.

The WISE survey (Wright et al., 2010) is a project whose main mission was executed between 2009 and 2011, and which objectives were to study asteroids, stars and luminous galaxies. The project is space-based with an infrared telescope able of imaging the whole sky. The filters used are called W1, W2, W3 and W4, and their effective wavelengths are 3.4, 4.6, 12.1 and 22.2 micrometers, respectively³.

For this work, the photometry in the W1 and W2 filters were obtained from the unWISE (Lang, 2014) catalogue. This is a project whose aim is to unblur AllWISE images, which is a combination of WISE and NEOWISE (Mainzer et al., 2011) data. The AllWISE images are convolved with a model of the point-spread function and this process blurs the images. Although they are ideal for detecting isolated point sources, this blurring can affect photometric measurements. After removing this blur via a process described in the aforementioned paper, the author obtained photometry in the W1, W2, W3 and W4 bands of roughly 469 million sources detected in SDSS using the new and unblurred unWISE data.

The total number of objects counting galaxies, quasars and stars in the Stripe-82 region of S-PLUS DR1 is 3009731, SDSS DR15 is 242527 and is 7239470 for unWISE.

The first step to construct the sample used in this work is to find unWISE objects that have counterparts in the SDSS catalogue. Since the unWISE catalogue objects share the same ID as the SDSS ones, a simple ID matching was done between these two catalogues. The second step is to find the SDSS/unWISE objects corresponding to the S-PLUS DR1. To do this, a coordinate (RA, Dec) matching was done between these catalogues, with a maximum separation of 1 arcsecond, measured in a circle around the object of the first catalogue (i.e. S-PLUS DR1), using the TOPCAT software (Taylor, 2005). This results in a sample containing all S-PLUS objects with SDSS/unWISE counterparts. The total number of objects in the sample is 135960, without any restrictions.

One of the most important aspects of training machine learning algorithms is the selection of the training sample. Since a bad sample will usually lead to bad results, there is the need to do a pre-processing and feature selection of the sample before presenting this data to the algorithms.

The distribution of the r-band magnitudes and redshift for the full data and the sample used in this work can be seen in Figure 3.1. The major difference for the samples at redshifts higher than 0.75 occurs due to the removal of quasars. Overall, the filters that remove most of objects are the constraints on PhotoFlag, PROB_GAL, and class_SDSS, as described in section 3.2.

²<http://skyserver.sdss.org/CasJobs/default.aspx>

³<http://wise2.ipac.caltech.edu/docs/release/allsky/>

Figure 3.1: Distribution of r -band magnitude and redshift for the samples used for $n\text{Det_auto} \geq 8$. The filtered data in orange is the sample used in this work.

3.2 | Pre-processing

Some of the data supplied by the surveys may not have an acceptable photometric quality. Thankfully, the catalogues provide means of selecting the best quality data available. For this work, a series of constraints were applied to the data.

The S-PLUS magnitude in the r -band has been limited between 16 and 21. The lower limit is to avoid objects with bad photometry, mostly due to close neighbours which can bias the magnitudes. The upper limit is defined as the magnitude at which sources with signal to noise larger than ~ 3 are expected to be complete (Mendes de Oliveira et al., 2019).

The second constrained parameter is the S-PLUS $n\text{Det_auto}$, and represents the number of filters in which the object was detected. In this work, the effect of different number of missing features in machine learning algorithms is analysed, comprising objects with complete photometry ($n\text{Det_auto} = 12$) up to objects with four or less missing bands ($n\text{Det_auto} \geq 8$).

Another parameter is the S-PLUS PhotoFlag, given by SExtractor (Bertin and Arnouts, 1996), to determine the quality of the magnitude measurement. All objects selected have PhotoFlag equal to zero.

The objective of this work is to determine photometric redshifts of galaxies. For this reason, only objects with the S-PLUS parameter $\text{PROB_GAL} \geq 0.5$ were chosen. The star-galaxy separation was done using Random Forests, which is also a machine learning algorithm, and the details are presented in Costa-Duarte et al. (2019).

Since there are misclassifications in this star-galaxy separation scheme (as in any other) and since our data has spectroscopic redshifts from SDSS, a filter in $z_{\text{spec}} > 1 \times 10^{-4}$ is applied to remove remaining stars. Also, only objects in SDSS with spectroscopic redshift error $z_{\text{Err}} < 0.4$ are kept.

The SDSS class determination is used to discard quasars in the sample. The redshift determination of these objects will be presented in Nakazono et al. (2019).

The summary of parameters adopted to define the sample, respective limits and motivations are in Table 3.1.

The percentage of objects with missing information (non-detections or non-observations) for each band is presented in Figure 3.3. The filter with the highest percentage of objects with missing information is J0395, which is missing from 8450 objects. It is followed by the u-band filter, with 7971 objects and J0378, which is missing from 7410 objects.

Table 3.1: Parameters, respective surveys, values used for restriction and the motivation.

Parameter	Value	Survey	Motivation
Magnitude (r-band)	[16, 21]	S-PLUS	Select objects with good photometry.
nDet_auto	[8, 12]	S-PLUS	Study the effect of missing bands in z_{phot} determination.
PhotoFlag	= 0	S-PLUS	Avoid bad photometry, such as due to blended objects.
PROB_GAL	≥ 0.5	S-PLUS	Select galaxies only, based in the star-galaxy separation conducted by Costa-Duarte et al. (2019).
z_{spec}	$\geq 1 \times 10^{-4}$	SDSS+WISE	Remove remaining stars misclassified as galaxies after PROB_GAL constrains.
z_{Err}	≤ 0.4	SDSS+WISE	Guarantee spectroscopic redshifts of good quality.
class_SDSS	$\neq \text{QSO}$	SDSS+WISE	Remove Quasars.

Figure 3.2: Number of objects in the training, validation and testing samples for each $nDet_auto$ value.

Figure 3.3: Percentage of objects with missing information in each band for $nDet_auto \geq 8$.

It can be noticed that the selection criteria applied to the sample reduces the number of objects considerably, and a small training sample is not ideal for machine learning tasks. Nonetheless, as we will see, the results obtained can be considered good. The number of objects as a function of the number of filters in which it was detected is presented in Table 3.2.

After selecting the sample, it is now needed to verify which features can be relevant to the task of photometric redshift determination. Also, machine learning codes take advantage of a high number of training samples and/or training features. For this reason, we conducted a test to verify which configuration of input features can provide the best results.

From the S-PLUS DR1 catalogue, the following features were used: the twelve “auto” magnitudes, shown in Figure 1.4, the normalized Full Width at Half Maximum (FWHM_n), the superficial brightness (MUMAX) and the ellipticity, calculated as $(1 - B/A)$, where B is the size along the minor axis and A is along the major axis (in pixels). From the SDSS/unWISE catalogue, the W1 and W2 magnitudes were selected as input.

The target redshifts used in this work were obtained from the SDSS/unWISE catalogue, while the BPZ photometric redshifts and Odds were obtained from the S-PLUS DR1.

As noticed by Cholley (2017), in machine learning problems, deep neural networks have the capability of learning that features with a value of zero does not contribute to the results, as long as this value is not meaningful for the problem being studied. For this reason, any missing bands in the S-PLUS DR1 catalogue have their values set to zero instead of 99, in order to teach the model to ignore this value. With this approach some issues appear, like unrealistic colours, since the colours are calculated by Δmag_{1-2} . To fix this, any colour < -10 is set to -10 and colours > 10 are set to 10.

Table 3.2: Number of objects in the training, validation and testing samples for each $n\text{Det_auto}$ value considered.

$n\text{Det_auto}$ value	Train. Sample	Valid. Sample	Test. Sample	Total
$n\text{Det_auto} = 12$	19679	6612	11268	37559
$n\text{Det_auto} \geq 11$	24705	8294	14143	47142
$n\text{Det_auto} \geq 10$	27770	9310	15892	52972
$n\text{Det_auto} \geq 9$	29442	9814	16824	56080
$n\text{Det_auto} \geq 8$	29963	10047	17148	57158

3.3 | Training, validation and testing samples

After defining the sample, a normalization step is done with Python, using the `StandardScaler` function of the `scikit-learn` package. Using this, each column in the dataset is normalized by its mean and standard deviation. This way all features have a $\mu = 0$ and $\sigma = 1$. The mean and standard deviation are calculated using the training sample only, in order to avoid “information leak” to the testing dataset.

The first step before starting the training process of any method is to split of the entire sample into a dataset for the training process and a testing dataset, containing 70% and 30% of the total number of samples, respectively (process “a” in Figure 3.4). The dataset for the training process is further split into training and validation datasets using K-Fold Cross Validation, and the testing dataset remains untouched until the end of the process in order to probe the precision of the model on unseen data.

A K-Fold Cross Validation approach was adopted in the training processes of all methods. A number $K=4$ was chosen so that, in each fold, 75% of objects are used for training and 25% are used for validation. Graphically, it can be presented as the process “b” in Figure 3.4. The process can be considered “cyclic”, meaning that all subsamples are used at least once as a validation sample.

Also, it should be noticed that all methods have the exact same training, validation and testing samples in order to keep a fair comparison between results.

With this we have four different trained models for each code. At the end of the training process a prediction is made, on the testing dataset, using the four models. The results are then averaged and presented in Chapter 4.

Figure 3.4: Representation of a 4-fold cross validation. The training samples are represented in blue, the validation sample in yellow and the testing sample is in green. The process denoted by “a” is the first split, and the process denoted by “b” represents the 4-fold cross validation split.

4 Results

In this chapter, the results of our photometric redshift analysis on the S-PLUS DR1 data will be presented. The results of the missing feature analysis are evaluated in Section 4.2. The performance of different configurations of input features are compared in Section 4.3. A comparison between the single-point estimate results of each method is done in section 4.4, while the detailed plots for each method separately are presented in Appendix A.5. An analysis regarding the probability density functions for each method is shown in Section 4.6. For all situations, the results are obtained using the testing sample, which contains objects not previously seen by the models during their training phases. This is essential to make a robust estimation of the precision of the algorithms.

The parameters used to configure each method are shown in Appendix A.2.

4.1 | Metrics

It is important to define the metrics that will be used to evaluate the models. The metrics chosen are the same presented in Molino et al. (2019) to provide a fair comparison between the methods used in this work and BPZ:

The scatter (σ_{NMAD}), is used as a direct measurement of the accuracy of the method. The normalized mean absolute deviation is less sensitive to outliers (objects located in the tail of the δz distribution) (Brammer et al., 2008; Molino et al., 2019). The bias (μ) allow us to identify any systematic effects in the photo-z determination. The outlier fraction (η) is used to determine the number of catastrophic results.

Table 4.1: Metrics, denotations and definitions that are used to analyse the results of this work.

Metric	Denoted by	Equation
Delta	δz	$z_{\text{phot}} - z_{\text{spec}}$
Scatter	σ_{NMAD}	$1.48 \times \text{median} \left(\frac{ \delta z - \text{median}(\delta z) }{1 + z_{\text{spec}}} \right)$
Bias	μ	$\overline{\delta z}$
Outlier Fraction	η	$\frac{ \delta z }{1 + z_{\text{spec}}} > 0.15$

Notice that, in the analysis of these metrics as function of r-band magnitude and/or redshift bins, they are measured for all testing sample objects inside the bin only.

For the final results, presented in Table 4.5 and Table A.4, these metrics are determined for the entire testing sample.

4.2 | Missing feature analysis

In this section, the results for all methods, considering different values of `nDet_auto`, will be presented. A total of five situations will be analysed, from objects with complete photometry (`nDet_auto` = 12) to objects with at most four missing bands (`nDet_auto` $\geq$ 8).

4.2.1 Scatter (σ_{NMAD})

Figure 4.1: Missing feature analysis results for the scatter (σ_{NMAD}). From top to bottom: ANNz2, GPz and Deep Learning. The left column shows the statistics as a function of r -band magnitude, and the right column as a function of redshift. The shaded area represent the minimum and maximum values obtained in the 4-Fold CV and the lines represent the mean of the folds.

The panels (a) and (b) in Figure 4.1 show the results for the scatter (σ_{NMAD}) calculated for the predictions made with ANNz2 on the testing sample. It can be noticed that, although the scatter increases with the r-band magnitude, the accuracy of the model increases (σ_{NMAD} decreases) as more objects are considered during the training phase due to the inclusion of objects with missing bands. The scatter is around 1.5% for bright objects and increases to 4% for faint objects when all objects with $\text{nDet_auto} \geq 8$ are considered.

The same behaviour can be verified in Figure (b), where the scatter is approximately 2.5% for both nearby and distant objects, but increases to 4% in the $0.2 \leq z_{\text{spec}} \leq 0.4$ regime. This also shows that this method is not sensitive for the “redshift window of opportunity” between 0.26 and 0.32, where the [OIII] and H α lines enter two different S-PLUS narrow band filters (Molino et al., 2019).

Panel (c) in Figure 4.1 shows that GPz has 1.5% scatter for bright objects and reaches 4.5% for faint objects when $\text{nDet_auto} \geq 8$. Also, the results for objects with complete photometry are generally better up to $\text{r_auto} = 20$. This can be explained by the fact that this code is not optimized to deal with missing features, thus its performance degrades when objects with non-detections are included in the training data. In the panel (d) it is noticed that the inclusion of objects with missing bands only improve the results for $z_{\text{spec}} \geq 0.3$.

Panels (e) and (f) shows the surprisingly low σ_{NMAD} obtained with Deep Learning, mainly for $\text{nDet_auto} \geq 8$. The scatter is at 1% for bright objects and increases to 3.5% for faint objects. When analysing in redshift bins, the scatter is around 2% for nearby objects and 2.5% for distant objects. A noteworthy difference of the results in Panel (f) when compared to the previous methods is that this method is sensible to the redshift “opportunity window” at $0.26 \leq z_{\text{spec}} \leq 0.32$.

With these results, it is clear that the Deep Learning model has the best accuracy between the three machine learning codes, both in terms of magnitude and redshift. Also, ANNz2 and GPz have similar performances, but with a slight advantage of the former, as seen in Panels (a) and (b) for faint ($\text{r_auto} \geq 20$) and/or distant ($z_{\text{spec}} \geq 0.3$) objects.

4.2.2 Bias (μ)

Figure 4.2: Missing feature analysis results for the bias (μ). From top to bottom: ANNz2, GPz and Deep Learning. The left column shows the statistics as a function of r-band magnitude, and the right column as a function of redshift. The shaded area represent the minimum and maximum values obtained in the 4-Fold CV and the lines represent the mean of the folds.

Panels (a) and (b) in Figure 4.2 present the results for the bias, useful to determine if the method has a tendency to over or underestimate photo-zs. As can be seen in (a), the inclusion of objects with missing features decrease the bias, mainly for the faint end, where the bias is lower than 0.3%. Panel (b) shows a systematic effect present in all methods compared in this work, where the photometric redshifts are overestimated for $z_{\text{spec}} \leq 0.15$ and are underestimated for $z_{\text{spec}} \geq 0.4$. As previously mentioned, ANNz2 has a bias-correction feature, described in section 2.4.1, which helps maintaining the bias as a function of magnitude at such low values.

Panels (c) and (d) in Figure 4.2 present the bias for GPz. This code systematically underestimates photo-zs at almost all magnitudes, but the effect is not higher than 0.5% up to magnitude 20. The Panel (d) shows the same systematic effect as ANNz2, although to a lesser extent.

When analysing the bias results for Deep Learning in Panel (e), one can verify that there is a systematic underestimation of the photometric redshift at all magnitudes when `nDet_auto` ≥ 8 . Also, the same behaviour present in the results from ANNz2 and GPz can be seen in Panel (f), although it is less noticeable for the interval $0.1 \leq z_{\text{spec}} \leq 0.5$.

Although ANNz2 has the least biased results when analysing in magnitude bins, the method that performed better overall both in magnitudes and redshift bins is Deep Learning which, even though it has a systematic bias, seen in Panel (e), provided the less biased results seen in Panel (f), mainly when `nDet_auto` ≥ 8 .

4.2.3 Outlier fraction (η)

Figure 4.3: Missing feature analysis results for the outlier fraction (η). From top to bottom: ANNz2, GPz and Deep Learning. The left column shows the statistics as a function of r-band magnitude, and the right column as a function of redshift. The shaded area represent the minimum and maximum values obtained in the 4-Fold CV and the lines represent the mean of the folds.

Plots (a) and (b), in Figure 4.3, represent the outlier fractions calculated for ANNz2, the results show that the outlier fraction decreases when objects with missing information are included in the training sample. For the $n\text{Det_auto} \geq 8$, the outlier fraction is lower than 4% for all magnitude bins and less than 2% for the redshift bins.

The plots (c) and (d) exhibit the outlier fractions for GPz, which remains lower than 2% for all magnitude bins and 1% for almost all redshift bins when $n\text{Det_auto} \geq 8$.

The fraction of outliers of Deep Learning, presented in Figures (e) and (f) are below 3% for both the analysis in magnitude and redshift bins.

For the outlier fraction, the method which has the lower percentage is GPz, followed closely by Deep Learning. Both of these codes presented approximately half of the outlier fractions of ANNz2. The lack of outliers can also be seen in Figures A.6 for GPz and A.8 for Deep Learning. The summary of results is presented in Table 4.2.

Table 4.2: Final results obtained for all metrics studied for each method presented and for all $n\text{Det_auto}$ values.

$n\text{Det_auto} \geq 8$			
Method	σ_{NMAD} (%)	$\bar{\mu}$ (%)	η (%)
ANNz2	3.164	-0.077	1.021
GPz	3.159	-0.465	0.717
Deep Learning	2.330	-0.501	0.636
$n\text{Det_auto} = 12$			
ANNz2	2.846	-0.146	0.719
GPz	2.490	-0.345	0.311
Deep Learning	2.065	-0.661	0.284

It is clear that, for all methods presented, the best results were achieved when using the $n\text{Det_auto} \geq 8$ sample. Even with missing information, the better performance can be explained by the higher number of objects which can be used to train the algorithms.

For this reason, the results shown for the remaining of this text were obtained using this dataset.

4.3 | Input features analysis

In this section, different combinations of input features will be analysed to determine which one can offer the most accurate results. For this test, only the Keras model was used due to the inferior amount of time needed to train the models (20 minutes for the 4 folds, opposed to 2 hours

for GPz and 24+ hours for ANNz2) and higher accuracy when compared to the other ML codes. Also, given the results of section 4.2, only the sample with $nDet_auto \geq 8$ was considered.

The name of the samples used and their respective input feature configurations are presented in Table 4.3.

Figure 4.4 shows the result for the scatter, bias and outlier fraction when the model is trained using different configurations of input features.

Panels (a) and (b) demonstrate that the most accurate results for Deep Learning were obtained when using the full input space containing 30 features. The models trained with this sample provided the lowest σ_{NMAD} (higher accuracy) when compared to the other samples.

In (c) and (d) it can be noticed that the sample A also had the best overall results for bias, when accounting both in terms of magnitude and redshift.

The outlier fraction shown in Figures (e) and (f) shows that all samples provide similar results, with a slight advantage of the M+E sample for faint magnitudes and the A sample for high redshifts.

In summary, the best choice of input parameters is represented by sample A, using all magnitudes, colours (calculated with respect to g_auto) and the non-photometric information (FWHM_n, MUMAX and Ellipticity). As such, this is the configuration used for all machine learning codes presented in this work.

The results obtained using the samples described in this section are outlined in Table 4.4.

Table 4.3: Name of samples, description and the input features considered.

Sample	Input Features
A	uJAVA, J0378, J0395, J0410, J0430, g, J0515, r, J0660, i, J0861, z, W1, W2
	u-g, 378-g, 395-g, 410-g, 430-g, g-515, g-r, g-660, g-i, g-861, g-z, g-W1, g-W2
	FWHM_n, MUMAX, Ellipticity
M	uJAVA, J0378, J0395, J0410, J0430, g, J0515, r, J0660, i, J0861, z, W1, W2
M+E	uJAVA, J0378, J0395, J0410, J0430, g, J0515, r, J0660, i, J0861, z, W1, W2
	FWHM_n, MUMAX, Ellipticity
C	g
	u-g, 378-g, 395-g, 410-g, 430-g, g-515, g-r, g-660, g-i, g-861, g-z, g-W1, g-W2
C+E	g
	u-g, 378-g, 395-g, 410-g, 430-g, g-515, g-r, g-660, g-i, g-861, g-z, g-W1, g-W2
	FWHM_n, MUMAX, Ellipticity

Table 4.4: Results for Deep Learning using the samples described in Table 4.3.

Sample	σ_{NMAD} (%)	$\bar{\mu}$ (%)	$\bar{\mu}_{\text{esc}}^{\text{a}}$ (%)	η (%)
A	2.388	-0.772	-0.488	0.506
M	2.550	-0.973	-0.629	0.723
M+E	2.494	-0.670	-0.376	0.548
C	2.509	-0.241	-0.017	0.717
C+E	2.426	-0.429	-0.220	0.571

^a Defined as $\bar{\mu}/(1 + z_{\text{spec}})$.

Figure 4.4: Comparison of single-point estimate results for Deep Learning using samples with different input features. From top to bottom: scatter, bias and outlier fraction. The left column is the analysis as a function of r-band magnitude, the right column is as a function of redshift.

4.4 | Single-point estimates

In this section, a comparison is made between the three methods presented in this text and BPZ, in order to help the verification of the best method among those compared in this work. These estimates are obtained directly from the PDFs via a weighted average, as described by Equation (2.13).

The results for each method individually are presented in Appendix A.5.

Figure 4.5: Comparison of single-point estimate results for all methods. From top to bottom: scatter, bias and outlier fraction. The left column shows the statistics as a function of r -band magnitude, and the right column as a function of redshift.

Figure 4.5 presents a comparison of the results obtained for all three methods analysed in this work and the photometric redshift (z_b) from BPZ, the template-fitting based method which is currently being used in S-PLUS DR1.

Figures (a) and (b) demonstrate the precision of the results for each method, calculated for the exact same objects, using the σ_{NMAD} equation as defined in Table 4.1. It can be clearly

noticed that Deep Learning has the most accurate results at all magnitudes, with a σ_{NMAD} around 1% lower for faint objects. The advantage of Deep Learning can also be seen in Figure (b), where the scatter is lower than the other machine learning techniques for all redshifts. The exception is for BPZ at the $0.26 \leq z \leq 0.32$ window, where two prominent features enter the narrow-band filters of S-PLUS (namely the [OIII] and H α lines in the J0660 and J0861 filters respectively) and boosts the accuracy of template-fitting methods. Among the ML methods, only Deep Learning is sensible to this fact, showing an increase in precision in this same interval, although not at the same level.

Figures (c) and (d) present the bias calculated for all methods. The results obtained with the ANNz2 code are remarkably unbiased up to magnitude 19.5, with a small increase for objects fainter than this value. When analysing the bias as a function of redshift, the Keras model had the best results both for nearby and distant objects. It should be noticed that ANNz2 used a bias correction procedure, described in section 2.4.1, and a similar approach using Deep Learning is planned for future work. In general, all machine learning methods had results with lower bias when compared to BPZ.

Figures (e) and (f) show the outlier fraction for all methods. Comparatively, Deep Learning has the lowest number of objects outside the ± 0.15 interval, followed by GPz and ANNz2. BPZ suffers from a high number of outlier objects due to overestimation of the photometric redshift at low spectroscopic redshifts (as can be seen in the panel (c) in Figure 4.5). This fact can also be seen in the upper-left portion the density plot shown in Figure 4.6.

Figure 4.6: Distribution of photometric redshifts as a function of spectroscopic redshifts for BPZ.

In order to easily combine information about the metrics as a function of magnitude and

redshift, the following analysis is centred in the determination of the scatter, bias and outlier fractions as a function of both these quantities simultaneously.

Figure 4.7 presents the simultaneous r-band and redshift analysis results for all methods studied and BPZ.

From the leftmost column it is possible to note that all methods have lower precision for fainter objects, independent of their redshift. Also, all methods have a scatter lower than 2% for objects with $r_auto \leq 18$ and $z_{spec} \leq 0.2$. Deep Learning, on the other hand, has σ_{NMAD} lower than 2% for objects with magnitudes up to 19 and redshifts lower than 0.4. Moreover, this method has a sub-percent scatter for bright and nearby objects ($r_auto \leq 17$ and $z_{spec} \leq 0.1$).

The results for the bias are presented in the middle column. Similarly to the scatter results, all methods had higher bias for fainter objects, mainly those with redshift lower than 0.3.

The outlier fractions are shown in the rightmost column, where it is possible to note the remarkably low number of outliers of GPz. The other methods have similar results, with BPZ showing a higher number of outliers in the nearby, faint-end of the galaxies distribution.

The number of objects present in each bin is presented in Figure A.9.

For the single-point estimate analysis, a summary of the different metrics are presented in Table 4.5 for $nDet_auto \geq 8$. The results for samples limited to $nDet_auto$ between 9 and 12 are in the Appendix Table A.4.

Table 4.5: Results for each method for $nDet_auto \geq 8$.

Algorithm	σ_{NMAD} (%)	$\bar{\mu}$ (%)	$\bar{\mu}_{esc}^a$ (%)	η (%)
ANNz2	3.164	-0.077	0.140	1.021
GPz	3.159	-0.465	-0.178	0.717
Deep Learning	2.487	-0.405	-0.182	0.636
BPZ	3.080	0.353	0.580	2.758

^a Defined as $\bar{\mu}/(1 + z_{spec})$.

Figure 4.7: Magnitude and redshift simultaneous analysis for all methods presented in this work, including BPZ. Detailed plots for each method are in appendix.

4.5 | Loss Analysis

One of the problems related to training machine learning models is called overfitting. This issue is related to the model learning the noise in the training sample and is characterized by a increase in accuracy during the training, but decrease at the validation step.

One way to check if a model is overfitting is by the analysis of the loss (or accuracy) at each iteration. The results for GPz and Deep Learning are presented in Figure 4.8. This analysis was not done for ANNz2 because this method does not output this information.

Figure 4.8: Loss as a function of iteration for GPz and Deep Learning models created with Keras.

From Figure 4.8 it is noticed that, for both methods, there is a negligible amount of overfitting, since both losses are similar at the end of the training process.

4.6 | PDF Analysis

For this section, three main quantities were calculated. The Odds, the average of PDFs as a function of redshift, and the averaged PDFs as a function of magnitude in the r-band for all objects. The aim of this analysis is to determine how accurate the PDFs are and their behaviour for different ranges of magnitudes and distances.

The Odds value, which can be used as an estimate of the accuracy of the estimated redshift, is defined as (Benítez, 2000):

$$\text{Odds} = \frac{\int_{z_{\text{peak}} - \Delta z}^{z_{\text{peak}} + \Delta z} p(z) dz}{\int_{z_1}^{z_2} p(z) dz}, \quad (4.1)$$

where z_{peak} is the value corresponding to the highest peak of the PDF, Δz is the average photo-z precision in the nearby universe (see below), z_1 and z_2 are the minimum and maximum values of the redshift and $p(z)$ is the photometric redshift PDF.

As can be noticed from Equation (4.1), a precise (and narrow) PDF will have an Odds value close to one, since a major part of the PDF will be inside the $z_{\text{peak}} \pm \Delta z$ interval. This also represents the confidence of the method for a given estimate, since a less confident result will be characterized by broad PDFs.

For this work, Δz was chosen as 0.02, which is the average expected photo-z precision for a method such as BPZ (Molino et al., 2019). The choice of this particular value, instead of the average precision from the machine learning methods of this work, was done in order to provide a fair comparison with the BPZ code.

4.6.1 | ANNz2

PDFs generated with ANNz2 are based on a combination of different MLMs, such as explained in section 2.4.1.

The averaged PDFs in Figure 4.9 shows that the probability distribution functions of this method become broader as the r-band magnitude gets fainter. This is somewhat expected due to the difficulty in obtaining accurate photometric measurements for faint galaxies.

In the lower portion of the same figure, the averaged PDFs as a function of redshift are presented. It is noticed that their width does not increase significantly between nearby and distant objects, which is a sign that the uncertainty of the method is more dependent on the magnitudes than on the redshift of the objects.

Figure 4.9: Mean PDFs as a function of r -band magnitude and redshifts for ANNz2.

Figure 4.10 shows the underconfidence of the PDFs of this method, which are generally broad, since there are no objects with Odds ≥ 0.5 , with most objects possessing Odds values between 0.1 and 0.3. If the method was confident on its results, the PDF would be narrower, due to the lower uncertainty on the photometric redshift determination.

Figure 4.10: Distribution of Odds values for ANNz2.

4.6.2 | GPz

The PDFs generated by GPz are simple Gaussian distributions with mean equal to z_{phot} and variance given by the sum of model and noise variances, both outputs of the method (Gomes et al., 2017). For this reason, the PDFs of this method can be considered oversimplified when compared to the other methods.

The average PDFs for GPz in Figure 4.11 are notably smooth. This is a result of the PDF generation approach used by this method, since it is a combination of Gaussian functions. Although the probability distribution functions of GPz are smoother, they present similar widths when compared to the other methods.

Similarly to ANNz2, the distribution of Odds values seen in Figure 4.12 shows that the method provides broad PDFs in general, since most objects have Odds between 0.2 and 0.3, and there are no objects with Odds ≥ 0.6 .

Figure 4.11: Mean PDFs as a function of r -band magnitude and redshifts for GPz.

Figure 4.12: Distribution of Odds values for GPz.

4.6.3 | Deep Learning

The deep learning model developed with Keras generates PDFs with 200 bins between $z_{\text{spec}} = 0$ and 1.34, with a bin width of 0.0067. As explained in section 2.4.3, the model uses a Softmax activation, which returns a probability for each bin.

In the upper panel of Figure 4.13, it is noticed that the PDFs gets broader as the average is done on fainter magnitudes. This is a result of the noisier photometry of faint objects, which increase the uncertainty in the photo-z determination.

In the bottom plot is presented the average PDFs by bins of redshift, where one can see that the PDFs do not have broad distributions overall.

A distribution of Odds is presented in Figure 4.14. This shows that Deep Learning has broader PDFs in comparison with BPZ, with a lower amount of objects with Odds > 0.7 when compared with the latter. One possible reason for this difference is a correction applied to the PDFs performed by Molino et al. (2019) based in the highest probability density confidence interval method (HPDCI) described by Wittman et al. (2016). This procedure was not applied to the PDFs of this study and their implementation is planned for future work.

Figure 4.13: Mean PDFs as a function of r -band magnitude and redshifts for Deep Learning.

Figure 4.14: Distribution of Odds values for Deep Learning.

4.6.4 | Comparison between methods

It is possible to compare the Odds for the different methods using a density plot coloured by the Odds value. This plot is shown in Figure 4.15. One can see that, although BPZ provides the highest Odds values, there are a few objects with high Odds with different photo-zs when compared to the true redshift, meaning that the uncertainty in the photometric redshift estimation is low, but its value is not correct.

Moreover, it can be noticed that both BPZ and Deep Learning show a group of objects with higher Odds in the interval $0.26 \leq z_{\text{spec}} \leq 0.32$, which corresponds to the “redshift window of opportunity” as discussed earlier.

Another simple measurement of the precision of the PDFs can be done by calculating the percentage of objects for which the spectroscopic redshift falls within $n \times \Delta z$ (where $\Delta z = 0.02$) of the peak of the distribution. This way it is possible to determine if the PDFs can accurately represent the spectroscopic redshift. The results for all methods using this approach are presented in Table 4.6.

Since a visual comparison between the averaged PDFs of each method can be difficult, a Gaussian fit using the “curve_fit” function of the Scipy package is done to each one, and their standard deviation is calculated. This way it is possible to determine which method has broader or narrower distributions quantitatively. The results are presented in Table 4.7 for magnitude bins and Table 4.8 for redshift bins. The plot of the fit over the averaged PDFs are shown in Figures A.22, A.23 and A.24 for ANNz2, GPz and Deep Learning, respectively.

As can be seen in Tables 4.7 and 4.8, the deep-learning framework Keras, in general, provides the narrowest (lowest standard deviation) PDFs among the three different machine learning approaches used in this work.

Table 4.6: Percentage of objects with z_{spec} within 1, 3 and 5 times Δz of the peak using the $n\text{Det_auto} \geq 8$ results.

Method	$1 \times \Delta z$ (%)	$3 \times \Delta z$ (%)	$5 \times \Delta z$ (%)
ANNz2	57.44	92.13	97.96
GPz	57.77	92.62	98.20
Deep Learning	56.50	90.61	97.18
BPZ	55.34	86.52	95.05

Figure 4.15: Distribution of spectroscopic and photometric redshifts. Each bin is coloured by the Odds value.

Table 4.7: Results of Gaussian fit to averaged PDFs in r -band magnitude bins.

Method	ANNz2	GPz	Deep Learning
Magnitude Bins	STD	STD	STD
$16 \leq r < 17$	0.0412	0.0435	0.0326
$17 \leq r < 18$	0.0711	0.0665	0.0719
$18 \leq r < 19$	0.0936	0.0973	0.1004
$19 \leq r < 20$	0.1733	0.1906	0.2202
$20 \leq r \leq 21$	0.2348	0.1606	0.1156

Table 4.8: Results of Gaussian fit to averaged PDFs in redshift bins.

Method	ANNz2	GPz	Deep Learning
Redshift Bins	STD	STD	STD
$0.0 \leq z < 0.1$	0.0605	0.0572	0.0438
$0.1 \leq z < 0.2$	0.0778	0.0690	0.0658
$0.2 \leq z < 0.3$	0.1123	0.0909	0.0904
$0.3 \leq z < 0.4$	0.1304	0.1104	0.0891
$0.4 \leq z < 0.5$	0.1474	0.1114	0.0932
$0.5 \leq z < 0.6$	0.1329	0.1056	0.0737
$0.6 \leq z \leq 0.7$	0.1495	0.1311	0.0948

4.7 | Summary

Based on the results of Sections 4.2, 4.4 and 4.6, the method which performed overall better on the sample used in this work was Deep Learning. The main advantage of this method is the higher accuracy (lower σ_{NMAD}) seen on panels (a) and (b) of Figure 4.5.

Although the bias of Deep Learning as a function of magnitude is not as low as the ANNz2 one, the bias-correction procedure of the latter can be implemented in the Keras model, possibly reducing this method bias further. When the bias is analysed as a function of the redshift, Deep Learning provided the lowest values for almost all bins, specially for objects further than $z_{\text{spec}} \geq 0.3$.

The Deep Learning model does not have the lowest outlier fraction but is comparable to the two other machine learning methods used in this work, as can be seen on panels (e) and (f), but it is still superior to the template-fitting BPZ in this regard.

When σ_{NMAD} , μ and η are calculated for the entire $\text{nDet_auto} \geq 8$ sample (not binning on magnitudes or redshift), Deep Learning had an scatter of 2.49%, a bias of 0.4% and a 0.64% outlier fraction, as seen in Table 4.5. Also, when the r - z_{spec} simultaneous analysis is presented in Figure 4.7, we can note the better results obtained specially for fainter objects.

As for the PDF analysis, Deep Learning presents narrower PDFs in general, as seen with the Odds distribution of Figure 4.14, when compared to the other two machine learning codes. Moreover, the width of the r -band magnitude averaged probability distribution functions of Deep Learning are slightly narrower than the averaged PDFs of the other methods, as seen in Tables 4.7 and 4.8.

Given the results presented in this section, it is possible to conclude that Deep Learning provided the best results among all methods compared.

5 Conclusions

Distance measurements are fundamental to several science cases in astronomy and cosmology, and the scale of nowadays surveys makes the spectroscopic determination of this quantity either limited to small areas (for deep, narrow surveys) or to relatively shallow magnitudes (for large area surveys).

In the recent years, there was an increasing use of photometric redshifts due to the development of multi-wavelength photometric surveys. The use of this quantity allows the determination of the distance to fainter and/or distant objects, for which a spectroscopic measurement would demand a high amount of observation time.

Although photometric redshifts are not as precise as the spectroscopic counterparts, recent developments in both data acquisition (in particular in multiband surveys) and processing methods allows us to obtain accurate estimates like never before.

Moreover, this technique allows the estimation of distances to a great number of galaxies, billions in the case of LSST. Even with lower precision than spectroscopic redshifts, this knowledge provides an opportunity to study the large scale structure of the universe, which is robust to photometric redshift errors (Zhan et al., 2006), for example.

The objective of this research was to evaluate the accuracy of photo-z measurements using machine learning techniques and was conducted using data from the S-PLUS DR1, SDSS DR15 and unWISE.

We have shown that the combination of narrow-band filter surveys, such as S-PLUS, J-PLUS and J-PAS, with a machine learning approach provides an excellent opportunity to acquire accurate photometric redshifts. This is mainly due to the nature of this kind of method, which takes advantage of a higher number of features to constrain the relation between the characteristics of an object, such as magnitudes and colours, and the desired information (i.e., the redshift).

A prime example of this are the results obtained with Deep Learning, presented in Sections 4.2, 4.6.3 and A.5.3 which, even using a relatively small training dataset for machine learning, with less than 60000 objects, but with 30 input features, can obtain better results when compared to the currently used method for photometric redshift determination used in the S-PLUS survey DR1, the template-fitting based BPZ.

Another method with good results is ANNz2, which achieved a very low bias. Due to the bias correction procedure applied by the method, the photometric redshifts are mostly unbiased, as seen on Figure 4.5 or Figure A.3 even for faint objects. This correction procedure is described in the ANNz2 code repository¹ and it is desirable to apply the same methodology to Deep Learning, in order to unify the low bias of the former code to the high accuracy on photo-z predictions of

¹<https://github.com/IftachSadeh/ANNZ>

the latter method. This is also planned for future work.

Given the importance of redshift determination and the difficulties of acquiring this information spectroscopically for deep, wide surveys, studies have been focusing on improving the precision of photometric redshifts as a faster and cheaper alternative. Accurate photo-zs are key to the success of photometric surveys and different science cases have different requirements. Large Scale Structure Baryonic Acoustic Oscillation measurements require a photo-z scatter lower than 3% and bias lower than 0.3% (Weinberg et al., 2013) for example. From this work it can be verified that these requirements can be met using machine learning techniques with narrow-band surveys which overlap with a spectroscopic survey in order to have a training data. For instance, Deep Learning achieved a scatter $\sigma_{\text{NMAD}} = 2.49\%$ and bias $\mu = 0.4\%$. As already mentioned, this result can be improved using a similar implementation of the bias-correction procedure of ANNz2, which proved effective given that this method achieved a bias $\mu = 0.08\%$.

The PDF results evaluated using the Odds parameter shows that BPZ probability distribution functions are narrower than all machine learning algorithms presented in this work. However, as shown in Section 4.6.4, it is noticed that BPZ has a number of objects with high Odds values but photo-zs different than their spec-zs. Also, when we analyse the simplified metric shown in Table 4.6, BPZ had the worst performance. Nevertheless, further optimization with the aim of improving PDF determinations is planned as future work.

Also, the single-point estimates calculated for Deep Learning using equation 2.13 provides an insight of the accuracy this method can achieve. Even not having many objects with Odds ≥ 0.7 (figure 4.14), these photo-zs calculated directly from the PDFs resulted in the best z_{phot} accuracy among all methods.

Overall, the Deep Learning method provided more accurate results than ANNz2 (Boosted Decision Trees), GPz (Gaussian Processes), and BPZ (Template-fitting). This makes evident the potential of this technique to solve problems in Astronomy.

It is expected that these results continue improving as more surveyed areas overlapping with spectroscopic redshift surveys become available.

5.1 | Future perspectives

Machine learning is a complex field of science, requiring large amounts of dedication and time. As such, at the end of two years of master's degree studies, only the surface of this unique and emerging field has been scratched. Still, this methodology offers big potential for several science cases.

The results shown in this text can still be improved, either by further optimizing the network used or by using entirely different architectures, such as convolutional neural networks. Also, it

is possible to create a neural network which is optimized for blue and red galaxies separately, instead of addressing both with the same model, as in this dissertation.

The size of the sample used, of approximately 60000 objects, may not be sufficient to take advantage of the full potential ANNs have. For this reason different data augmentation approaches can be adopted to generate new galaxies and increment the original training sample. This can be done by random sampling of the magnitudes and redshift of the objects (since these quantities have uncertainties) or with more complex methods, such as using generative adversarial networks, introduced by [Goodfellow et al. \(2014\)](#), which is a type of architecture developed specifically to generate data based on real observations.

Moreover, photometric redshifts and their probability distribution functions can be used in a number of different studies, described briefly in Section 1.3.

The main project for a PhD work would be the improvement of machine learning models, with the aim of acquiring even more precise photometric redshift determinations, for studies regarding galaxy evolution.

Moreover, we plan to apply the ML codes to determine photometric redshifts for the S-PLUS, J-PLUS and J-PAS surveys.

Bibliography

- Adelberger, K. L. et al., “Optical Selection of Star-forming Galaxies at Redshifts $1 < z < 3$ ”, *ApJ*, vol. 607, no. 1, pp. 226–240, 2004. [astro-ph/0401445](#), URL <http://dx.doi.org/10.1086/383221>.
- Aguado, D. S. et al., “The Fifteenth Data Release of the Sloan Digital Sky Surveys: First Release of MaNGA-derived Quantities, Data Visualization Tools, and Stellar Library”, *ApJS*, vol. 240, no. 2, 23, 2019. [1812.02759](#), URL <http://dx.doi.org/10.3847/1538-4365/aaf651>.
- Almeida, S. V. W. et al., “SPACE: S-Plus gAlaxy Cluster cataloguE, DR1 Stripe-82”, , 2019. In preparation.
- Almosallam, I., “Heteroscedastic Gaussian processes for uncertain and incomplete data”, (2017).
- Almosallam, I. A., Jarvis, M. J. and Roberts, S. J., “GPZ: non-stationary sparse Gaussian processes for heteroscedastic uncertainty estimation in photometric redshifts”, *MNRAS*, vol. 462, no. 1, pp. 726–739, 2016. [1604.03593](#), URL <http://dx.doi.org/10.1093/mnras/stw1618>.
- Armitage, T. J., Kay, S. T. and Barnes, D. J., “An application of machine learning techniques to galaxy cluster mass estimation using the MACSIS simulations”, *Monthly Notices of the Royal Astronomical Society*, vol. 484, no. 2, pp. 1526–1537, 2019. [1810.08430](#), URL <http://dx.doi.org/10.1093/mnras/stz039>.
- Arnouts, S. et al., “Measuring and modelling the redshift evolution of clustering: the Hubble Deep Field North”, *MNRAS*, vol. 310, pp. 540–556, 1999. [astro-ph/9902290](#), URL <http://dx.doi.org/10.1046/j.1365-8711.1999.02978.x>.
- Baldry, I. K. et al., “The Sloan Digital Sky Survey u-band Galaxy Survey: luminosity functions and evolution”, *MNRAS*, vol. 358, pp. 441–456, 2005. [astro-ph/0501110](#), URL <http://dx.doi.org/10.1111/j.1365-2966.2005.08799.x>.
- Baldry, I. K. et al., “Galaxy And Mass Assembly (GAMA): the input catalogue and star-galaxy separation”, *MNRAS*, vol. 404, pp. 86–100, 2010. [0910.5120](#), URL <http://dx.doi.org/10.1111/j.1365-2966.2010.16282.x>.
- Baum, W. A., “Photoelectric determinations of redshifts beyond 0.2 c.”, *AJ*, vol. 62, pp. 6–7, 1957. URL <http://dx.doi.org/10.1086/107433>.
- Becker, R. H., White, R. L. and Helfand, D. J., “The VLA’s FIRST Survey”, in: *Astronomical Data Analysis Software and Systems III*, edited by D. R. Crabtree, R. J. Hanisch and J. Barnes, *Astronomical Society of the Pacific Conference Series*, vol. 61, p. 165 (1994).
- Belson, W. A., “Matching and Prediction on the Principle of Biological Classification”, *Applied Statistics*, vol. 8, no. 2, p. 65, 1959. URL <http://dx.doi.org/10.2307/2985543>.

- Benítez, N., “Bayesian Photometric Redshift Estimation”, *ApJ*, vol. 536, pp. 571–583, 2000. [astro-ph/9811189](https://arxiv.org/abs/astro-ph/9811189), URL <http://dx.doi.org/10.1086/308947>.
- Benitez, N. et al., “J-PAS: The Javalambre-Physics of the Accelerated Universe Astrophysical Survey”, *arXiv e-prints*, arXiv:1403.5237, 2014. [1403.5237](https://arxiv.org/abs/1403.5237).
- Bennett, A. S., “The revised 3C catalogue of radio sources.”, *Memoirs of the Royal Astronomical Society*, vol. 68, p. 163, 1962.
- Bernstein, R. A. et al., “Overview and status of the Giant Magellan Telescope project”, in: *Ground-based and Airborne Telescopes V*, Proc. SPIE, vol. 9145, p. 91451C (2014). URL <http://dx.doi.org/10.1117/12.2055282>.
- Bertin, E. and Arnouts, S., “SExtractor: Software for source extraction.”, *A&AS*, vol. 117, pp. 393–404, 1996. URL <http://dx.doi.org/10.1051/aas:1996164>.
- Bilicki, M. et al., “Photometric redshifts for the Kilo-Degree Survey. Machine-learning analysis with artificial neural networks”, *A*, vol. 616, A69, 2018. [1709.04205](https://arxiv.org/abs/1709.04205), URL <http://dx.doi.org/10.1051/0004-6361/201731942>.
- Blake, C. et al., “The WiggleZ Dark Energy Survey”, *Astronomy and Geophysics*, vol. 49, no. 5, pp. 5.19–5.24, 2008. URL <http://dx.doi.org/10.1111/j.1468-4004.2008.49519.x>.
- Bolton, A. S. et al., “Spectral Classification and Redshift Measurement for the SDSS-III Baryon Oscillation Spectroscopic Survey”, *AJ*, vol. 144, 144, 2012. [1207.7326](https://arxiv.org/abs/1207.7326), URL <http://dx.doi.org/10.1088/0004-6256/144/5/144>.
- Bolzonella, M., Miralles, J.-M. and Pelló, R., “Photometric redshifts based on standard SED fitting procedures”, *A&A*, vol. 363, pp. 476–492, 2000. [astro-ph/0003380](https://arxiv.org/abs/astro-ph/0003380).
- Brammer, G. B., van Dokkum, P. G. and Coppi, P., “EAZY: A Fast, Public Photometric Redshift Code”, *ApJ*, vol. 686, pp. 1503–1513, 2008. [0807.1533](https://arxiv.org/abs/0807.1533), URL <http://dx.doi.org/10.1086/591786>.
- Brun, R. and Rademakers, F., “ROOT: An object oriented data analysis framework”, *Nucl. Instrum. Meth.*, vol. A389, pp. 81–86, 1997. URL [http://dx.doi.org/10.1016/S0168-9002\(97\)00048-X](http://dx.doi.org/10.1016/S0168-9002(97)00048-X).
- Bruzual, G. and Charlot, S., “Stellar population synthesis at the resolution of 2003”, *MNRAS*, vol. 344, pp. 1000–1028, 2003. [astro-ph/0309134](https://arxiv.org/abs/astro-ph/0309134), URL <http://dx.doi.org/10.1046/j.1365-8711.2003.06897.x>.
- Bruzual A., G. and Charlot, S., “Spectral evolution of stellar populations using isochrone synthesis”, *ApJ*, vol. 405, pp. 538–553, 1993. URL <http://dx.doi.org/10.1086/172385>.
- Bundy, K. et al., “Overview of the SDSS-IV MaNGA Survey: Mapping nearby Galaxies at Apache Point Observatory”, *ApJ*, vol. 798, 7, 2015. [1412.1482](https://arxiv.org/abs/1412.1482), URL <http://dx.doi.org/10.1088/0004-637X/798/1/7>.

-
- Cannon, R. et al., “The 2dF-SDSS LRG and QSO (2SLAQ) Luminous Red Galaxy Survey”, MNRAS, vol. 372, pp. 425–442, 2006. [astro-ph/0607631](#), URL <http://dx.doi.org/10.1111/j.1365-2966.2006.10875.x>.
- Carrasco Kind, M. and Brunner, R. J., “TPZ: photometric redshift PDFs and ancillary information by using prediction trees and random forests”, MNRAS, vol. 432, no. 2, pp. 1483–1501, 2013. [1303.7269](#), URL <http://dx.doi.org/10.1093/mnras/stt574>.
- Cenarro, A. J. et al., “J-PLUS: The Javalambre Photometric Local Universe Survey”, A&A, vol. 622, p. A176, 2019. URL <http://dx.doi.org/10.1051/0004-6361/201833036>.
- Chevallard, J. and Charlot, S., “Modelling and interpreting spectral energy distributions of galaxies with BEAGLE”, MNRAS, vol. 462, no. 2, pp. 1415–1443, 2016. [1603.03037](#), URL <http://dx.doi.org/10.1093/mnras/stw1756>.
- Chollet, F., *Deep Learning with Python* (Manning Publications Co., Greenwich, CT, USA, 2017), 1st ed., ISBN 1617294438, 9781617294433.
- Chollet, F. et al., “Keras”, <https://keras.io>, 2015.
- Coleman, G. D., Wu, C.-C. and Weedman, D. W., “Colors and magnitudes predicted for high redshift galaxies”, ApJS, vol. 43, pp. 393–416, 1980. URL <http://dx.doi.org/10.1086/190674>.
- Colless, M. et al., “The 2dF Galaxy Redshift Survey: spectra and redshifts”, MNRAS, vol. 328, pp. 1039–1063, 2001. [astro-ph/0106498](#), URL <http://dx.doi.org/10.1046/j.1365-8711.2001.04902.x>.
- Collister, A. A. and Lahav, O., “ANNz: Estimating Photometric Redshifts Using Artificial Neural Networks”, PASP, vol. 116, pp. 345–351, 2004. [astro-ph/0311058](#), URL <http://dx.doi.org/10.1086/383254>.
- Costa-Duarte, M. V. et al., “The S-PLUS: a star/galaxy classification based on a Machine Learning approach”, *arXiv e-prints*, arXiv:1909.08626, 2019. [1909.08626](#).
- Cucciati, O. et al., “Measuring galaxy environment with the synergy of future photometric and spectroscopic surveys”, *Monthly Notices of the Royal Astronomical Society*, vol. 462, no. 2, pp. 1786–1801, 2016. [1604.05323](#), URL <http://dx.doi.org/10.1093/mnras/stw1729>.
- Dahlen, T. et al., “A Detailed Study of Photometric Redshifts for GOODS-South Galaxies”, ApJ, vol. 724, no. 1, pp. 425–447, 2010. [1009.3504](#), URL <http://dx.doi.org/10.1088/0004-637X/724/1/425>.
- Davis, M. et al., “Science Objectives and Early Results of the DEEP2 Redshift Survey”, in: *Discoveries and Research Prospects from 6- to 10-Meter-Class Telescopes II*, edited by P. Guhathakurta, Proc. SPIE, vol. 4834, pp. 161–172 (2003). [astro-ph/0209419](#), URL <http://dx.doi.org/10.1117/12.457897>.
- Dozat, T., “Incorporating Nesterov Momentum into Adam”, in: *ICLR 2016 Workshop Submissions* (Stanford University, 2016).

- Eisenstein, D. J. et al., “Spectroscopic Target Selection for the Sloan Digital Sky Survey: The Luminous Red Galaxy Sample”, *AJ*, vol. 122, pp. 2267–2280, 2001. [astro-ph/0108153](#), URL <http://dx.doi.org/10.1086/323717>.
- Elliott, J. et al., “The overlooked potential of Generalized Linear Models in astronomy-II: Gamma regression and photometric redshifts”, *Astronomy and Computing*, vol. 10, pp. 61–72, 2015. [1409.7699](#), URL <http://dx.doi.org/10.1016/j.ascom.2015.01.002>.
- Falco, E. E. et al., “The Updated Zwicky Catalog (UZC)”, *PASP*, vol. 111, pp. 438–452, 1999. [astro-ph/9904265](#), URL <http://dx.doi.org/10.1086/316343>.
- Feldmann, R. et al., “The Zurich Extragalactic Bayesian Redshift Analyzer and its first application: COSMOS”, *MNRAS*, vol. 372, pp. 565–577, 2006. [astro-ph/0609044](#), URL <http://dx.doi.org/10.1111/j.1365-2966.2006.10930.x>.
- Firth, A. E., Lahav, O. and Somerville, R. S., “Estimating photometric redshifts with artificial neural networks”, *MNRAS*, vol. 339, no. 4, pp. 1195–1202, 2003. [astro-ph/0203250](#), URL <http://dx.doi.org/10.1046/j.1365-8711.2003.06271.x>.
- Freeth, T., “The Antikythera Mechanism: Challenging the Classic Research”, *Mediterranean Archaeology and Archaeometry*, vol. 2, pp. 21–35, 2002.
- Geach, J. E. et al., “VICS82: The VISTA-CFHT Stripe 82 Near-infrared Survey”, *ApJS*, vol. 231, 7, 2017. [1705.05451](#), URL <http://dx.doi.org/10.3847/1538-4365/aa74b6>.
- Gomes, Z., Jarvis, M. J., Almosallam, I. A. and Roberts, S. J., “Improving photometric redshift estimation using GPz: size information, post processing, and improved photometry”, *MNRAS*, vol. 475, no. 1, pp. 331–342, 2017. URL <http://dx.doi.org/10.1093/mnras/stx3187>.
- Goodfellow, I. J. et al., “Generative Adversarial Networks”, *arXiv e-prints*, arXiv:1406.2661, 2014. [1406.2661](#).
- Gorecki, A. et al., “A new method to improve photometric redshift reconstruction. Applications to the Large Synoptic Survey Telescope”, *A*, vol. 561, A128, 2014. [1301.3010](#), URL <http://dx.doi.org/10.1051/0004-6361/201321102>.
- Gruel, N. et al., “Calibration plan for J-PAS and J-PLUS surveys”, in: *Observatory Operations: Strategies, Processes, and Systems IV*, edited by A. B. Peck, R. L. Seaman and F. Comeron, vol. 8448, pp. 547 – 554, International Society for Optics and Photonics (SPIE, 2012). URL <http://dx.doi.org/10.1117/12.925581>.
- He, K., Zhang, X., Ren, S. and Sun, J., “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”, *arXiv e-prints*, arXiv:1502.01852, 2015. [1502.01852](#).
- Hocker, A. et al., “TMVA - Toolkit for Multivariate Data Analysis with ROOT: Users guide”, Tech. Rep. physics/0703039, CERN, Geneva, 2007. TMVA-v4 Users Guide: 135 pages, 19 figures, numerous code examples and references, URL <https://cds.cern.ch/record/1019880>.

- Hodge, J. A., Becker, R. H., White, R. L., Richards, G. T. and Zeimann, G. R., “High-resolution Very Large Array Imaging of Sloan Digital Sky Survey Stripe 82 at 1.4 GHz”, *AJ*, vol. 142, 3, 2011. [1103.5749](#), URL <http://dx.doi.org/10.1088/0004-6256/142/1/3>.
- Hodge, P., Krienke, O. K., Bianchi, L., Massey, P. and Olsen, K., “A Photometric Catalog of 77 Newly Recognized Star Clusters in M31”, *Publications of the Astronomical Society of the Pacific*, vol. 122, no. 893, p. 745, 2010. [1005.2198](#), URL <http://dx.doi.org/10.1086/654810>.
- Hoyle, F., “A New Model for the Expanding Universe”, *MNRAS*, vol. 108, p. 372, 1948. URL <http://dx.doi.org/10.1093/mnras/108.5.372>.
- Hubble, E., “A Relation between Distance and Radial Velocity among Extra-Galactic Nebulae”, *Proceedings of the National Academy of Science*, vol. 15, pp. 168–173, 1929. URL <http://dx.doi.org/10.1073/pnas.15.3.168>.
- Huchra, J. P. et al., “The 2MASS Redshift Survey - Description and Data Release”, *ApJS*, vol. 199, 26, 2012. [1108.0669](#), URL <http://dx.doi.org/10.1088/0067-0049/199/2/26>.
- Ilbert, O. et al., “Accurate photometric redshifts for the CFHT legacy survey calibrated using the VIMOS VLT deep survey”, *A&A*, vol. 457, pp. 841–856, 2006. [astro-ph/0603217](#), URL <http://dx.doi.org/10.1051/0004-6361:20065138>.
- James, P. A. et al., “The H α galaxy survey. I. The galaxy sample, H α narrow-band observations and star formation parameters for 334 galaxies”, *A*, vol. 414, pp. 23–43, 2004. [astro-ph/0311030](#), URL <http://dx.doi.org/10.1051/0004-6361:20031568>.
- Jiang, L. et al., “The Sloan Digital Sky Survey Stripe 82 Imaging Data: Depth-optimized Co-adds over 300 deg² in Five Filters”, *ApJS*, vol. 213, no. 1, 12, 2014. [1405.7382](#), URL <http://dx.doi.org/10.1088/0067-0049/213/1/12>.
- Jones, D. H. et al., “The 6dF Galaxy Survey: final redshift release (DR3) and southern large-scale structures”, *MNRAS*, vol. 399, pp. 683–698, 2009. [0903.5451](#), URL <http://dx.doi.org/10.1111/j.1365-2966.2009.15338.x>.
- King, H., *The History of the Telescope* (Sky Publishing Corporation, 1955). URL <https://books.google.com.br/books?id=CmTmjwEACAAJ>.
- Kingma, D. P. and Ba, J., “Adam: A Method for Stochastic Optimization”, *arXiv e-prints*, arXiv:1412.6980, 2014. [1412.6980](#).
- Kotila, M. et al., “Talos”, <http://github.com/autonomio/talos>, 2019.
- Krupp, E., *Echoes of the Ancient Skies: The Astronomy of Lost Civilizations*, Oxford paperbacks : Mythology, astronomy / Oxford paperbacks (Oxford University Press, 1994), ISBN 9780195088014. URL <https://books.google.com.br/books?id=1IApAQAMA AJ>.
- Kruuse, M., Tempel, E., Kipper, R. and Stoica, R. S., “Photometric redshift galaxies as tracers of the filamentary network”, *A&A*, vol. 625, A130, 2019. [1905.00912](#), URL <http://dx.doi.org/10.1051/0004-6361/201935096>.

- Kurtz, M. J., Geller, M. J., Fabricant, D. G., Wyatt, W. F. and Dell’Antonio, I. P., “ μ -PhotoZ: Photometric Redshifts by Inverting the Tolman Surface Brightness Test”, *The Astronomical Journal*, vol. 134, no. 4, pp. 1360–1367, 2007. [0707.0484](#), URL <http://dx.doi.org/10.1086/521430>.
- Laine, S. and Gottesman, S. T., “Neutral hydrogen gas distribution and kinematics in NGC 7479”, *Monthly Notices of the Royal Astronomical Society*, vol. 297, no. 4, pp. 1041–1051, 1998. URL <http://dx.doi.org/10.1046/j.1365-8711.1998.01513.x>.
- LaMassa, S. M. et al., “Finding rare AGN: X-ray number counts of Chandra sources in Stripe 82”, *MNRAS*, vol. 432, pp. 1351–1360, 2013a. [1210.0550](#), URL <http://dx.doi.org/10.1093/mnras/stt553>.
- LaMassa, S. M. et al., “Finding rare AGN: XMM-Newton and Chandra observations of SDSS Stripe 82”, *MNRAS*, vol. 436, pp. 3581–3601, 2013b. [1309.7048](#), URL <http://dx.doi.org/10.1093/mnras/stt1837>.
- Lang, D., “unWISE: Unblurred Coadds of the WISE Imaging”, *AJ*, vol. 147, no. 5, 108, 2014. [1405.0308](#), URL <http://dx.doi.org/10.1088/0004-6256/147/5/108>.
- Lazio, J., “The Square Kilometre Array”, in: *Panoramic Radio Astronomy: Wide-field 1-2 GHz Research on Galaxy Evolution*, p. 58 (2009). [0910.0632](#).
- Le Fèvre, O. et al., “The VIMOS VLT deep survey. First epoch VVDS-deep survey: 11564 spectra with $17.5 \leq I_{AB} \leq 24$, and the redshift distribution over $0 \leq z \leq 5$ ”, *A&A*, vol. 439, pp. 845–862, 2005. [astro-ph/0409133](#), URL <http://dx.doi.org/10.1051/0004-6361:20041960>.
- Lemaître, G., “Un Univers homogène de masse constante et de rayon croissant rendant compte de la vitesse radiale des nébuleuses extra-galactiques”, *Annales de la Société Scientifique de Bruxelles*, vol. 47, pp. 49–59, 1927.
- Levi, M. E. et al., “The Dark Energy Spectroscopic Instrument (DESI)”, *arXiv e-prints*, arXiv:1907.10688, 2019. [1907.10688](#).
- Lilly, S. J. et al., “zCOSMOS: A Large VLT/VIMOS Redshift Survey Covering $0 < z < 3$ in the COSMOS Field”, *ApJS*, vol. 172, pp. 70–85, 2007. [astro-ph/0612291](#), URL <http://dx.doi.org/10.1086/516589>.
- Linnainmaa, S., *The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors*, Master’s thesis, Univ. Helsinki, Finland, 1970.
- LSST Science Collaboration et al., “Science-Driven Optimization of the LSST Observing Strategy”, *arXiv e-prints*, arXiv:1708.04058, 2017. [1708.04058](#).
- Mahabal, A. et al., “Deep-Learnt Classification of Light Curves”, *arXiv e-prints*, arXiv:1709.06257, 2017. [1709.06257](#).
- Mainzer, A. et al., “Preliminary Results from NEOWISE: An Enhancement to the Wide-field Infrared Survey Explorer for Solar System Science”, *ApJ*, vol. 731, 53, 2011. [1102.1996](#), URL <http://dx.doi.org/10.1088/0004-637X/731/1/53>.

- Mandelbaum, R. et al., “Precision photometric redshift calibration for galaxy-galaxy weak lensing”, *Monthly Notices of the Royal Astronomical Society*, vol. 386, no. 2, pp. 781–806, 2008. 0709.1692, URL <http://dx.doi.org/10.1111/j.1365-2966.2008.12947.x>.
- Marín-Franch, A. et al., “Design of the J-PAS and J-PLUS filter systems”, in: *Modern Technologies in Space- and Ground-based Telescopes and Instrumentation II*, edited by R. Navarro, C. R. Cunningham and E. Prieto, vol. 8450, pp. 1253 – 1258, International Society for Optics and Photonics (SPIE, 2012). URL <http://dx.doi.org/10.1117/12.925430>.
- Masters, D. et al., “Mapping the Galaxy Color-Redshift Relation: Optimal Photometric Redshift Calibration Strategies for Cosmology Surveys”, *The Astrophysical Journal*, vol. 813, no. 1, 53, 2015. 1509.03318, URL <http://dx.doi.org/10.1088/0004-637X/813/1/53>.
- McCulloch, W. S. and Pitts, W., “A logical calculus of the ideas immanent in nervous activity”, *Bulletin of Mathematical Biology*, vol. 5, no. 4, pp. 115–133, 1943. URL <http://dx.doi.org/10.1007/bf02478259>.
- Mendes de Oliveira, C. et al., “The Southern Photometric Local Universe Survey (S-PLUS): improved SEDs, morphologies and redshifts with 12 optical filters”, *MNRAS*, p. 2048, 2019. 1907.01567, URL <http://dx.doi.org/10.1093/mnras/stz1985>.
- Mickaelian, A. M., “Astronomical surveys and big data”, *Baltic Astronomy*, vol. 25, pp. 75–88, 2016. 1511.07322.
- Molino, A. et al., “CLASH: accurate photometric redshifts with 14 HST bands in massive galaxy cluster cores”, *MNRAS*, vol. 470, no. 1, pp. 95–113, 2017. 1705.02265, URL <http://dx.doi.org/10.1093/mnras/stx1243>.
- Molino, A. et al., “Assessing the photometric redshift precision of the S-PLUS survey: the Stripe-82 as a test-case”, *arXiv e-prints*, 2019. 1907.06315.
- Moraes, B. et al., “The CFHT/MegaCam Stripe-82 Survey”, in: *Revista Mexicana de Astronomia y Astrofisica Conference Series*, *Revista Mexicana de Astronomia y Astrofisica Conference Series*, vol. 44, pp. 202–203 (2014).
- Nakazono, L. M. I. et al., “Classification of stars, quasars and galaxies in S-PLUS DR1 (Stripe-82) using machine learning”, , 2019. In preparation.
- Nesterov, Y., “A method for solving the convex programming problem with convergence rate $O(1/k^2)$ ”, (1983).
- Neugebauer, G. et al., “The Infrared Astronomical Satellite (IRAS) mission”, *ApJ*, vol. 278, pp. L1–L6, 1984. URL <http://dx.doi.org/10.1086/184209>.
- Niemack, M. D. et al., “Improving Photometric Redshifts Using Galaxy Evolution Explorer Observations for the Sloan Digital Sky Survey Stripe 82 and the Next Generation of Optical and Sunyaev-Zeldovich Cluster Surveys”, *ApJ*, vol. 690, pp. 89–101, 2009. 0803.3221, URL <http://dx.doi.org/10.1088/0004-637X/690/1/89>.

- Pearson, K. A., Palafox, L. and Griffith, C. A., “Searching for exoplanets using artificial intelligence”, *MNRAS*, vol. 474, no. 1, pp. 478–491, 2018. [1706.04319](#), URL <http://dx.doi.org/10.1093/mnras/stx2761>.
- Pelló, R. et al., “Photometric redshifts and cluster tomography in the ESO Distant Cluster Survey”, *Astronomy and Astrophysics*, vol. 508, no. 3, pp. 1173–1191, 2009. [0910.5418](#), URL <http://dx.doi.org/10.1051/0004-6361/200810644>.
- Petri, A., May, M. and Haiman, Z., “Cosmology with photometric weak lensing surveys: Constraints with redshift tomography of convergence peaks and moments”, *Physical Review D*, vol. 94, no. 6, 063534, 2016. [1605.01100](#), URL <http://dx.doi.org/10.1103/PhysRevD.94.063534>.
- Puschell, J. J., Owen, F. N. and Laing, R. A., “Near-infrared photometry of distant radio galaxies - Spectral flux distributions and redshift estimates”, *ApJ*, vol. 257, pp. L57–L61, 1982. URL <http://dx.doi.org/10.1086/183808>.
- Ramos, B. H. F. et al., “Evolution of Galaxy Luminosity Function Using Photometric Redshifts”, *The Astronomical Journal*, vol. 142, no. 2, 41, 2011. [1105.5668](#), URL <http://dx.doi.org/10.1088/0004-6256/142/2/41>.
- Rasmussen, C. E. and Williams, C. K. I., *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)* (The MIT Press, 2005), ISBN 026218253X.
- Rosenblatt, F., “The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain”, *Psychological Review*, pp. 65–386, 1958.
- Sadeh, I., Abdalla, F. B. and Lahav, O., “ANNz2: Photometric Redshift and Probability Distribution Function Estimation using Machine Learning”, *PASP*, vol. 128, no. 968, p. 104502, 2016. [1507.00490](#), URL <http://dx.doi.org/10.1088/1538-3873/128/968/104502>.
- Salvato, M., Ilbert, O. and Hoyle, B., “The many flavours of photometric redshifts”, *Nature Astronomy*, vol. 3, pp. 212–222, 2019. [1805.12574](#), URL <http://dx.doi.org/10.1038/s41550-018-0478-0>.
- Sánchez, E., “The Dark Energy Survey”, *Journal of Physics: Conference Series*, vol. 259, p. 012080, 2010. URL <http://dx.doi.org/10.1088/1742-6596/259/1/012080>.
- Schanche, N. et al., “Machine-learning approaches to exoplanet transit detection and candidate validation in wide-field ground-based surveys”, *MNRAS*, vol. 483, no. 4, pp. 5534–5547, 2019. [1811.07754](#), URL <http://dx.doi.org/10.1093/mnras/sty3146>.
- Selin, H., *Astronomy Across Cultures: The History of Non-Western Astronomy*, Science Across Cultures: The History of Non-Western Science (Springer Netherlands, 2012), ISBN 9789401141796. URL <https://books.google.com.br/books?id=vRHvCAAQBAJ>.
- Serra, P. et al., “Neutral hydrogen gas within and around NGC 1316”, *Astronomy and Astrophysics*, vol. 628, A122, 2019. [1907.08265](#), URL <http://dx.doi.org/10.1051/0004-6361/201936114>.

- Shectman, S. A. et al., “The Las Campanas Redshift Survey”, *ApJ*, vol. 470, p. 172, 1996. [astro-ph/9604167](https://arxiv.org/abs/astro-ph/9604167), URL <http://dx.doi.org/10.1086/177858>.
- Slipher, V. M., “The radial velocity of the Andromeda Nebula”, *Lowell Observatory Bulletin*, vol. 2, pp. 56–57, 1913.
- Soo, J. Y. H. et al., “Morpho-z: improving photometric redshifts with galaxy morphology”, *MNRAS*, vol. 475, no. 3, pp. 3613–3632, 2018. [1707.03169](https://arxiv.org/abs/1707.03169), URL <http://dx.doi.org/10.1093/mnras/stx3201>.
- Steidel, C. C., Giavalisco, M., Pettini, M., Dickinson, M. and Adelberger, K. L., “Spectroscopic Confirmation of a Population of Normal Star-forming Galaxies at Redshifts $Z \geq 3$ ”, *ApJ*, vol. 462, p. L17, 1996. [astro-ph/9602024](https://arxiv.org/abs/astro-ph/9602024), URL <http://dx.doi.org/10.1086/310029>.
- Steidel, C. C. et al., “A Large Structure of Galaxies at Redshift Z approximately 3 and Its Cosmological Implications”, *ApJ*, vol. 492, pp. 428–438, 1998. [astro-ph/9708125](https://arxiv.org/abs/astro-ph/9708125), URL <http://dx.doi.org/10.1086/305073>.
- Storrie-Lombardi, M. C., Lahav, O., Sodre, J., L. and Storrie-Lombardi, L. J., “Morphological Classification of Galaxies by Artificial Neural Networks”, *Monthly Notices of the Royal Astronomical Society*, vol. 259, p. 8P, 1992. URL <http://dx.doi.org/10.1093/mnras/259.1.8P>.
- Strauss, M. A. et al., “Spectroscopic Target Selection in the Sloan Digital Sky Survey: The Main Galaxy Sample”, *AJ*, vol. 124, pp. 1810–1824, 2002. [astro-ph/0206225](https://arxiv.org/abs/astro-ph/0206225), URL <http://dx.doi.org/10.1086/342343>.
- Tamura, N. et al., “Prime Focus Spectrograph (PFS) for the Subaru telescope: overview, recent progress, and future perspectives”, in: *Proc. SPIE, Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, vol. 9908, p. 99081M (2016). [1608.01075](https://arxiv.org/abs/1608.01075), URL <http://dx.doi.org/10.1117/12.2232103>.
- Tarenghi, M. and Wilson, R. N., “The ESO NTT (New Technology Telescope) - The first active optics telescope”, in: *Active telescope systems*, edited by F. J. Roddier, *Proc. SPIE*, vol. 1114, pp. 302–313 (1989). URL <http://dx.doi.org/10.1117/12.960835>.
- Taylor, M. B., “TOPCAT & STIL: Starlink Table/VOTable Processing Software”, in: *Astronomical Data Analysis Software and Systems XIV*, edited by P. Shopbell, M. Britton and R. Ebert, *Astronomical Society of the Pacific Conference Series*, vol. 347, p. 29 (2005).
- Tully, R. B., de Marseille, O. and Fisher, J. R., “A New Method of Determining Distances to Galaxies”, in: *BAAS*, vol. 7, p. 426 (1975).
- Vargas dos Santos, M., Quartin, M. and Reis, R. R. R., “On the cosmological performance of photometric classified supernovae with machine learning”, *arXiv e-prints*, arXiv:1908.04210, 2019. [1908.04210](https://arxiv.org/abs/1908.04210).
- Voges, W., “The ROSAT all-sky survey”, *Advances in Space Research*, vol. 13, 1993. URL [http://dx.doi.org/10.1016/0273-1177\(93\)90147-4](http://dx.doi.org/10.1016/0273-1177(93)90147-4).

- Weinberg, D. H. et al., “Observational probes of cosmic acceleration”, *Physics Reports*, vol. 530, no. 2, pp. 87–255, 2013. [1201.2434](#), URL <http://dx.doi.org/10.1016/j.physrep.2013.05.001>.
- Wittman, D., Bhaskar, R. and Tobin, R., “Overconfidence in photometric redshift estimation”, *MNRAS*, vol. 457, no. 4, pp. 4005–4011, 2016. [1601.07857](#), URL <http://dx.doi.org/10.1093/mnras/stw261>.
- Wright, E. L. et al., “The Wide-field Infrared Survey Explorer (WISE): Mission Description and Initial On-orbit Performance”, *AJ*, vol. 140, pp. 1868–1881, 2010. [1008.0031](#), URL <http://dx.doi.org/10.1088/0004-6256/140/6/1868>.
- Zhan, H., Knox, L., Tyson, J. A. and Margoniner, V., “Exploring Large-Scale Structure with Billions of Galaxies”, *ApJ*, vol. 640, no. 1, pp. 8–17, 2006. [astro-ph/0508119](#), URL <http://dx.doi.org/10.1086/500077>.

A Appendix

A.1 | SDSS Query

The query used to obtain the SDSS sample used in this work is presented in the Listing [A.1](#).

Listing A.1: CasJobs query for the SDSS sample

```
1 SELECT spec.bestobjid, spec.ra, spec.dec, spec.z, spec.zErr, spec.class,  
   ↪ spec.subclass,  
2   phot.dered_u, phot.dered_g, phot.dered_r, phot.dered_i, phot.dered_z  
3 INTO MyDB.DR15_S82_Spec_subc  
4 FROM SpecObj as spec, PhotoObj as phot  
5 WHERE  
6   spec.BestObjID = phot.ObjID AND (spec.zWarning = 0 OR spec.zWarning = 10) AND  
   ↪ spec.ra BETWEEN 0 and 60 AND spec.dec BETWEEN -1.5 and 1.5 OR  
7   spec.BestObjID = phot.ObjID AND (spec.zWarning = 0 OR spec.zWarning = 10) AND  
   ↪ spec.ra BETWEEN 300 and 360 AND spec.dec BETWEEN -1.5 and 1.5
```

A.2 | Method Parameters

In this section, the parameters used for each method will be presented.

The configuration used for ANNz2 is presented in table [A.1](#) for the training and optimization steps. GPz configuration is presented in table [A.2](#) and the Keras model parameters are shown in table [A.3](#).

A.2.1 | ANNz2 parameters

For ANNz2, the number of machine learning methods to be learned during the training process is set to 100 following the recommendation of [Sadeh et al. \(2016\)](#). The minimum and maximum values for the target variable (the redshift) are set to zero and 1.34 respectively, where 1.34 was chosen because that is the highest redshift in the sample after applying the constraints described in Section 3.2.

The “rndOptTypes” parameter govern which machine learning methodology will be used to generate the 100 nMLMs during the training and, in this case, it was set to “BDT” to randomly generate only boosted decision/regression trees. ANNs were not used due to time constraints, since they take a much longer time to train.

“doBiasCorMLM” is the switch used to indicate that the method should correct all MLMs for the bias, such as described in section 2.4.1.

The “nPDFs” and “nPDFbins” are parameters used to configure the PDF generation. The first is used to determine the type of PDF that will be generated, and the value 1 is the default of the code. The second parameter refers to the number of bins each PDF will have and, to match the PDF generation of Deep Learning, this value was fixed at 200.

For the optimization step, the parameters “max_sigma68_PDF”, “max_bias_PDF” and “max_frac68_PDF” are thresholds for the determination of the MLMs that will be used to generate PDFs. If a given MLM has some parameter higher than those, it is excluded from the PDF generation.

Table A.1: Parameters chosen to be used during the training and optimization procedures of ANNz2.

Training Process		
Parameter	Value	Description
nMLMs	100	Number of ML methods to train.
minValZ	0	Minimum z to generate the PDFs.
maxValZ	1.34	Maximum z to generate the PDFs
rndOptTypes	BDT	Type of ML method that ANNz2 will randomly generate.
doBiasCorMLM	True	To apply a bias-correction procedure to SPEs and PDFs.
nPDFs	1	Type of PDF that will be generated.
nPDFbins	200	To create PDFs with 200 bins between minValZ and maxValZ.
Optimization Process		
Parameter	Value	Description
max_sigma68_PDF	0.044	Maximum value for the scatter, calculated as σ_{68} . ^a
max_bias_PDF	0.010	Maximum value for the bias.
max_frac68_PDF	0.100	Maximum outlier fraction. ^b

^a 68th percentile scatter. ^b Defined as objects with $\Delta z \geq \sigma_{68}$.

A.2.2 | GPz parameters

The “method” parameter refers to the radial basis function method, where the selected value (VL), specifies that each basis function will have a specific length-scale. This method was chosen based on prior testing and performed better than the other options.

The “maxIter” and “m” parameters are the maximum number of iterations and the number of basis functions to be used, respectively. The value of 2000 for maxIter was chosen based in an analysis of the loss value as a function of iterations. At this value, the training loss and the validation loss had approximately the same value, indicating that the method had converged. The “m” parameter was chosen to be 100 since this number was enough to provide good results, and increasing the number of basis also increased the computational time while not providing further improvement. Also, values higher than 200 decreased the stability of this method, causing errors and preventing the completion of training process.

The “joint” flag indicates that the code should learn a linear model and non-linear deviations from it jointly. This is done due to the fact that kernels (basis functions) used in Gaussian processes generally assume a mean of zero, which means that, in regions absent of data, the regressor will sample from a distribution centred in zero and this can lead to potentially wrong results

based on the distribution of data. Instead, GPz can estimate a linear model given the distribution of data in order to avoid this problem. This process is described in detail in [Almosallam \(2017\)](#) (p. 42).

The “heteroscedastic” parameters indicates that the code should model the noise variance as input-dependent. With this parameter set to “False”, the noise variance is assumed to be white Gaussian noise with a fixed precision β . If set to “True”, the this precision β is assumed to be a linear combination of basis functions which are dependent of the input data.

The “cslMethod”, or cost-sensitive method, is a technique used to balance the input data, since there are some regions with less observations (such as at the high- z universe, for example). In this work, the “normalized” scheme (explained in section 2.4.2) was chosen since it provides better results than the “normal” scheme ([Gomes et al., 2017](#)).

Lastly, the “optimMethod” relates to the type of solver for the “minimize” function of the “scipy” Python package, used internally by GPz. The “BFGS” method is the default.

Table A.2: Parameters chosen to be used during the training procedure of GPz.

Parameter	Value	Description
method	VL	Variable length-scales, as explained in section 2.4.2.
maxIter	2000	The number of iterations the method will calculate.
m	100	The number of basis functions to describe the training dataset.
joint	True	To jointly learn a prior linear mean function.
heteroscedastic	True	To model the noise as a function of the input parameters.
csl_method	Normalized	To assign a weight to each object in the training sample, according to its redshift.
optimMethod	BFGS	The optimization method used to find the solution of the optimization process.

A.2.3 | Keras model parameters

The main parameters used in Keras are related to the configuration of the network itself.

The number of input neurons were chosen to be equal to the number of input features in the training dataset. While the number of units in the hidden layers were chosen based on trial and error testing.

The number of output neurons are dependent on the information they should provide. One neuron is responsible for single-point estimation for performance monitoring purposes, while the other 200 neurons are required to generate a PDF with 200 bins.

The activations also need to be related to the neuron function. In all hidden layers, the ReLU activation was chosen since it is well established, provides good results, is computationally cheap and introduces the needed non-linearity required for the photometric redshift estimation problem. In the output layers two different activations were used, the Linear for the single-point estimation and the Softmax for the PDF generation, as described in 2.4.3.

To train the model, the “he_normal” initializer, introduced by He et al. (2015), was chosen. This initializer takes into account the size of the previous layer, providing more efficiency in the search for global or local minimum of the loss function. Mathematically, it is described with (A.1):

$$\sigma = \sqrt{\frac{2}{n_{\text{prev}}}}, \quad (\text{A.1})$$

where n_{prev} is the number of neurons in the previous layer.

Also, a batch size of 256 was used. This means that only 256 objects were presented to the network at a time. These objects are used to estimate the relation between input and output and to calculate the gradient of the optimizer algorithm, which guides the optimization of the neural network weights.

Lastly, the models were trained for 500 epochs. Each epoch is a full pass on the training dataset. This value was chosen by comparing the loss values of the training and validation datasets. A increase in this value generally reflected as a loss of generalization capacity of this model, causing a deterioration of its accuracy.

Table A.3: Parameters chosen to be used during the training procedure of Keras.

Parameter	Value	Description
Input neurons	30	One for each training feature.
Hidden neurons	[35,15]	Each layer has its own number of neurons, as illustrated in figure 2.8.
Output neurons	200+1	1 neuron for SPEs with performance monitoring purposes. 200 neurons to generate the PDF.
Kernel initializer	he_normal	Method used to sample initial weights of the network.
Activations	ReLU, Linear and Softmax	All layers use the ReLU activation, except the output layers which use the Linear (SPEs) and Softmax (PDFs) activations.
Batch size	256	Number of objects presented to the network at each iteration.
Epochs	500	Number of times the entire training dataset is shown to the network.

A.3 | Training and Testing sample distribution per feature

In order to check if the training sample is consistent with the testing sample (with the same distribution), a histogram of each feature is made and is presented in Figure A.1. It can be seen that both samples follow the same distribution and can be considered consistent with each other.

Figure A.1: Distribution of the training and testing samples

A.4 | Odds values for the samples analysed in section 4.3

As can be seen in Figure A.2, different choices of input features have no significant impact on the accuracy of the PDFs calculated for the Deep Learning method.

Analysing this figure, it is possible to notice that the sample A, with all features described in Table 4.3, provided the best results, with less objects with Odds ≤ 0.4 and more objects with Odds ≥ 0.4 . This means that, for this sample, the PDFs are overall narrower when compared to the other configurations.

Figure A.2: Comparison of Odds distribution for samples with different input features using Deep Learning.

A.5 | Single-point estimate (SPE) results for each method

The specific single-point results obtained for each method will be presented in this section. To see the comparison between all methods and BPZ, please refer to [section 4.4](#).

A.5.1 | ANNz2

Figure A.3: Single-point estimate analysis results for ANNz2. From top to bottom: scatter, bias and outlier fraction. The left column is the analysis as a function of r-band magnitude, the right column is as a function of redshift. The shaded represent the minimum and maximum values obtained in the 4-Fold CV and the lines represent the mean of the folds.

From the plots “a” and “b” of figure A.3, it can be noticed that ANNz2 has, in general, a lower precision when compared to BPZ. Although the results are similar as a function of magnitude,

the discrepancy of the scatter as a function of the spectroscopic redshift can be as high as 2%, inside the redshift window of opportunity. Nonetheless, this method is two times more accurate than BPZ for distant objects.

Figures “c” and “d” show the bias for this method, and in both cases it performed better than BPZ. In fact, ANNz2 has the best results regarding the bias as a function of magnitude, with almost zero bias for $r_{\text{auto}} \leq 21$, mainly due to the bias correction applied during the training of the MLMs. However, such bias correction only appears when the bias is analysed as function of magnitude, since figure “d” has the same general negative slope of bias as the other methods presented.

Lastly, figures “e” and “f” indicates that the outlier fraction of this method is, generally, only half of the calculated fraction of outliers for BPZ.

Figure A.4: Distribution of photometric redshifts as a function of spectroscopic redshifts for ANNz2.

The plot in figure A.4 shows that most objects are close to the 1-to-1 line, which would be the ideal case, and most of the outlier objects lie close to the ± 0.15 lines. Moreover, it can be noted the lack of photometric redshifts higher than 0.6, which is caused by the lack of training samples with spectroscopic redshift higher than this value, such as seen in figure 3.1. Since the model was not trained with objects in this region, it cannot make correct predictions for objects with $z_{\text{spec}} \geq 0.6$, and these objects are probably considered outliers.

A.5.2 | GPz

Figure A.5: Single-point estimate analysis results for GPz. From top to bottom: scatter, bias and outlier fraction. The left column is the analysis as a function of r-band magnitude, the right column is as a function of redshift. The shaded represent the minimum and maximum values obtained in the 4-Fold CV and the lines represent the mean of the folds.

The scatter for this method, seen of plots “a” and “b” of figure A.5, is higher than the one computed for BPZ for objects with $n_{\text{Det_auto}} \geq 8$. This issue arises from the fact that GPz

is not optimized to deal with missing bands information in the current version, therefore its accuracy rapidly decreases when objects do not have complete photometry are included in the training dataset. A second version of this code is under development and will be able to deal with missing bands. As a function of redshift, GPz has worse general precision than BPZ, with the major difference in the $0.2 \leq z \leq 0.5$ interval, with contains the window of opportunity that increases the accuracy for template fitting codes such as BPZ.

The GPz bias, as those presented in the plots “c” and “d”, is close to zero at almost all magnitudes, but increases for faint objects, given their noisier photometry. As a function of redshift, the bias has an analogous behaviour to all method presented in this text, being biased towards photo- z overestimation for nearby objects and underestimation for objects with $z \geq 0.4$.

Surprisingly, the outlier fraction of GPz is the lowest of all codes compared in this work. Figure “e” shows that the outlier fraction is at most 2% for faint objects and figure “f” displays a general outlier fraction lower than 1% for all redshift intervals.

Figure A.6: Distribution of photometric redshifts as a function of spectroscopic redshifts for GPz.

The density plot of A.6 confirms the results shown in plots “e” and “f” of figure A.5, with noteworthy lack of objects outside the ± 0.15 interval.

A.5.3 Deep learning

Figure A.7: Single-point estimate analysis results for Deep Learning. From top to bottom: scatter, bias and outlier fraction. The left column is the analysis as a function of r-band magnitude, the right column is as a function of redshift. The shaded represent the minimum and maximum values obtained in the 4-Fold CV and the lines represent the mean of the folds.

Deep Learning has a higher precision in all magnitudes when compared to BPZ, as seen in the plot “a” of figure A.7, with an scatter approximately 1% lower for faint objects. The same

can be seen in figure “b”, where Deep Learning has lower σ_{NMAD} for almost all redshift intervals, with the exception of the $0.26 \leq z \leq 0.32$ interval, where two prominent lines of the nearby universe (O[III] and H α) enter two different narrow band S-PLUS filters. This situation boosts the accuracy of template fitting methods as can be seen from the BPZ results, but also improves the result for Deep Learning although to a lesser extent.

As can be seen in figures “c” and “d”, the Keras model has acceptable results when analysing the bias as a function of magnitude and as a function of z_{spec} . Although a systematic underestimation of the photometric redshift can be seen in figure “c”, it is noted that the absolute bias value is lower than that of BPZ for the same sample, with bigger differences at magnitudes fainter than 19.

The plots “e” and “f” show the results for the outlier fraction. Note that the results obtained with Deep Learning present a lower fraction of catastrophic results at all magnitudes and all redshifts.

Figure A.8: Distribution of photometric redshifts as a function of spectroscopic redshifts for Deep Learning.

Figure A.8 presents a density plot showing the distribution of photo-zs and spec-zs. The dashed line in the middle represent the 1-to-1 relation, which means that, if the prediction was perfect, all objects would lie in this line. The dotted red lines represent the ± 0.15 interval defined by the outlier fraction equation shown in Table 4.1, delimiting the region where objects are considered outliers.

A.6 | Detailed plots for the r-mag and z simultaneous analysis - SPE

A.6.1 | Number of objects per bin

The number of objects per bin of magnitude and redshift is presented in figure A.9. It should be noted that bins with a low amount of objects (< 70), can suffer from high variance due to the low number of objects to calculate the metrics σ_{NMAD} , μ , and η .

Figure A.9: Number of objects in each two-dimensional bin.

A.6.2 | ANNz2

The detailed results for the two-dimensional analysis for ANNz2 are present in this section. The number inside the bins are the metric calculated for the objects in said bin.

Figure A.10: ANNz2 results for scatter when analysing redshift and magnitude bins simultaneously.

Figure A.11: ANNz2 results for bias when analysing redshift and magnitude bins simultaneously.

Figure A.12: ANNz2 results for outlier fraction when analysing redshift and magnitude bins simultaneously.

A.6.3 GPz

The detailed results for GPz are presented in this section. The number inside the bins are the metric calculated for the objects in said bin.

It should be noted that the colour bar has a different scale when compared to other methods.

Figure A.13: GPz results for scatter when analysing redshift and magnitude bins simultaneously.

Figure A.14: GPz results for bias when analysing redshift and magnitude bins simultaneously.

Figure A.15: GPz results for outlier fraction when analysing redshift and magnitude bins simultaneously.

A.6.4 Deep Learning

In this section, the results for Deep Learning are presented. The number inside the bins are the metric calculated for the objects in said bin.

It should be noted that the colour bar has a different scale when compared to other methods.

Figure A.16: Deep Learning results for scatter when analysing redshift and magnitude bins simultaneously.

Figure A.17: Deep Learning results for bias when analysing redshift and magnitude bins simultaneously.

Figure A.18: Deep Learning results for outlier fraction when analysing redshift and magnitude bins simultaneously.

A.6.5 | BPZ

For comparison purposes, the same analysis was done using the photometric redshift calculated by BPZ on the same test sample used by other methods. The number inside the bins are the metric calculated for the objects in said bin.

It should be noted that the colour bar has a different scale when compared to other methods.

Figure A.19: BPZ results for scatter when analysing redshift and magnitude bins simultaneously.

Figure A.20: BPZ results for bias when analysing redshift and magnitude bins simultaneously.

Figure A.21: BPZ results for outlier fraction when analysing redshift and magnitude bins simultaneously.

A.7 | Results for all missing band scenarios

The results for each `nDet_auto` value is presented in the table [A.4](#).

Table A.4: Final results obtained for all metrics studied for each method presented and for all `nDet_auto` values.

nDet_auto ≥ 8				
Method	σ_{NMAD} (%)	$\bar{\mu}$ (%)	$\bar{\mu}_{\text{esc}}$ ^a (%)	η (%)
ANNz2	3.164	-0.077	0.140	1.021
GPz	3.159	-0.465	-0.178	0.717
Deep Learning	2.487	-0.405	-0.182	0.636
BPZ	3.080	0.353	0.580	2.758
nDet_auto ≥ 9				
ANNz2	3.142	-0.100	0.126	1.004
GPz	3.167	-0.421	-0.141	0.725
Deep Learning	2.436	-0.304	-0.091	0.666
BPZ	3.027	0.393	0.618	2.960
nDet_auto ≥ 10				
ANNz2	3.228	-0.174	0.077	1.087
GPz	3.068	-0.413	-0.144	0.717
Deep Learning	2.397	-0.410	0.173	0.585
BPZ	2.987	0.562	0.738	2.932
nDet_auto ≥ 11				
ANNz2	3.057	-0.116	0.103	0.944
GPz	2.932	-0.396	-0.154	0.552
Deep Learning	2.385	-0.361	-0.166	0.509
BPZ	2.928	0.824	0.934	2.892
nDet_auto = 12				
ANNz2	2.846	-0.146	0.056	0.719
GPz	2.490	-0.345	-0.168	0.311
Deep Learning	2.236	-0.385	-0.209	0.284
BPZ	2.887	0.949	0.992	2.583

^a Defined as $\bar{\mu}/(1 + z_{\text{spec}})$.

A.8 | Additional plots to PDF analysis

To calculate the values presented in tables 4.7 and 4.8, a Gaussian fit was done on the averaged PDFs. The best-fit Gaussian function to each PDF of each method are plotted in figures A.22 for ANNz2, A.23 for GPz and A.24 for Deep Learning.

Figure A.22: Gaussian fit to ANNz2 average PDFs.

Figure A.23: Gaussian fit to GPz average PDFs.

Figure A.24: Gaussian fit to Deep Learning average PDFs.

A.9 | PDF comparison between methods

Each method outputs a different PDF for a given object. The PDF from the three machine learning methods for a subset of the testing sample is shown in Figure A.25.

Figure A.25: Comparison of the PDFs for a subset of the testing sample. The r-band magnitude is shown on top of each plot.