Convolutional Neural Networks for Spectroscopic Redshift Estimation on Euclid Data

Radamanthys Stivaktakis, Grigorios Tsagkatakis, Bruno Moraes, Filipe Abdalla, Jean-Luc Starck, and Panagiotis Tsakalides

Abstract—In this paper, we address the problem of spectroscopic redshift estimation in Astronomy. Due to the expansion of the Universe, galaxies recede from each other on average. This movement causes the emitted electromagnetic waves to shift from the blue part of the spectrum to the red part, due to the Doppler effect. Redshift is one of the most important observables in Astronomy, allowing the measurement of galaxy distances. Several sources of noise render the estimation process far from trivial, especially in the low signal-to-noise regime of many astrophysical observations. In recent years, new approaches for a reliable and automated estimation methodology have been sought out, in order to minimize our reliance on currently popular techniques that heavily involve human intervention. The fulfilment of this task has evolved into a grave necessity, in conjunction with the insatiable generation of immense amounts of astronomical data. In our work, we introduce a novel approach based on Deep Convolutional Neural Networks. The proposed methodology is extensively evaluated on a spectroscopic dataset of full spectral energy galaxy distributions, modelled after the upcoming Euclid satellite galaxy survey. Experimental analysis on observations of idealistic and realistic conditions demonstrate the potent capabilities of the proposed scheme.

Index Terms—Astrophysics, cosmology, deep learning, convolutional neural networks, spectroscopic redshift estimation, euclid

1 Introduction

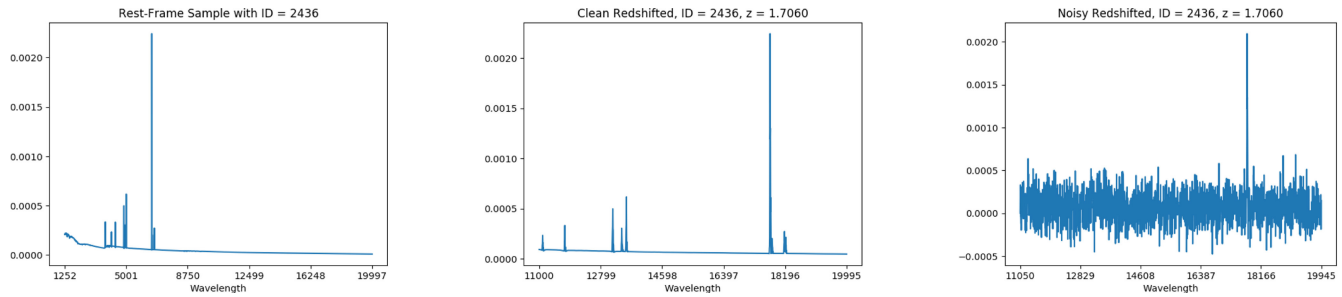
Modern cosmological and astrophysical research seeks answers to questions like "what is the distribution of dark matter and dark energy in the Universe?" [1], [2], or "how can we quantify transient phenomena, like exoplanets orbiting distant stars?" [3]. To answer such questions, a large number of deep space observation platforms have been deployed. Spaceborne instruments, such as the Planck Satellite[1] [4], the Kepler Space Observatory[2] [5] and the upcoming Euclid mission[3][6], seek to address these questions with unprecedented accuracy, since they avoid the deleterious effects of Earth's atmosphere, a strong limiting factor to all their observational strategies. Meanwhile, ground-based telescopes like the LSST[4] [7] will be able to acquire massive amounts of data through high frequency full-sky surveys, providing complementary observations. The number and capabilities of cutting-edge scientific instruments in these and other cases have led to the emergence of the concept of Big Data [8], mandating the need for new approaches on massive data processing and management. The analysis of huge numbers of observations from various sources has opened new horizons in scientific research, and astronomy is an indicative scenario where observations propel the data-driven scientific research [9].

One particular long-standing problem in astrophysics is the ability to derive precise estimates to galaxy redshifts. According to the Big Bang model, due to the expansion of the Universe and its statistical homogeneity and isotropy, galaxies move away from each other and any given observation point. A result of this motion is that light emitted from galaxies is shifted towards larger wavelengths through the Doppler effect, a process termed *redshifting*. Redshift estimation has been an integral part of observational cosmology, since it is the principal way in which we can measure galaxies' radial distances and hence their 3-dimensional position in the Universe. This information is fundamental for several observational probes in cosmology, such as the rate of expansion of the Universe and the gravitational lensing of light by the matter distribution—which is used to infer the total dark matter density—among other methods [10], [11].

1. http://www.esa.int/Our_Activities/Space_Science/Planck
2. http://kepler.nasa.gov/
3. http://sci.esa.int/euclid/
4. https://www.lsst.org

- *R. Stivaktakis and P. Tsakalides are with the Department of Computer Science, University of Crete, Heraklion 700 13, Greece, and also with the Institute of Computer Science, Foundation for Research and Technology (FORTH), Iraklio 700 13, Greece. E-mail: {stivakt, tsakalid}@ics.forth.gr.*
- *G. Tsagkatakis is with the Institute of Computer Science, Foundation for Research and Technology (FORTH), Iraklio 700 13, Greece. E-mail: greg@ics.forth.gr.*
- *B. Moraes is with the Instituto de Fisica, Universidade Federal do Rio de Janeiro, Rio de Janeiro 21941-972, Brazil. E-mail: b.moraes@ucl.ac.uk.*
- *F. Abdalla is with the Department of Physics and Astronomy, University College London, London WC1E 6BT, United Kingdom, and also with the Department of Physics and Electronics, Rhodes University, Grahamstown 6139, South Africa. E-mail: filipe.abdalla@gmail.com.*
- *J.-L. Starck is with Astrophysics Department, CEA Saclay, Paris F-91191, France. E-mail: jean-luc.starck@cea.fr.*

Manuscript received 2 Mar. 2018; revised 22 June 2019; accepted 6 Aug. 2019. Date of publication 14 Aug. 2019; date of current version 29 Aug. 2020. (Corresponding author: Radamanthys Stivaktakis.)
Recommended for acceptance by M. Datcu, J. Le Moigne, P. Soille, P. G. Marchetti, and G.-S. Xia.
Digital Object Identifier no. 10.1109/TBDATA.2019.2934475



(a) Clean Rest-Frame Spectral Profile

(b) Clean (Randomly) Redshifted Equivalent

(c) Noisy Redshifted Equivalent

Fig. 1. Examples of the data used. From the initial rest-frame samples, we produce random redshifted samples in clean and noisy forms. The *y*-axis corresponds to the spectral density flux value, in a normalized form.

The Euclid satellite aims to measure the global properties of the Universe to an unprecedented accuracy, with emphasis on a better understanding of the nature of Dark Energy. It will collect *photometric* data with broadband optical and near-infrared filters and *spectroscopic* data with a near-infrared slitless spectrograph. The latter will be one of the biggest upcoming spectroscopic surveys, and will help us determine the details of cosmic acceleration through measurements of the distribution of matter in cosmic structures. In particular, it will measure the characteristic distance scale imprinted by primordial plasma oscillations in the galaxy distribution. The projected launch date is set for 2022 and throughout its 6-year mission, Euclid will gather of the order of 50 million galaxy spectral profiles, originating from wide and deep sub-surveys. A top-priority issue associated with Euclid is the efficient processing and management of these enormous amounts of data, with scientific specialists from both astrophysical and engineering backgrounds contributing to the ongoing research. To successfully achieve this purpose, we need to ensure that realistically simulated data will be available, strictly modeled after the real observations coming from Euclid in terms of quality, veracity and volume.

Estimation of redshift from spectroscopic observations is far from straightforward. There are several sources of astrophysical and instrumental errors, such as readout noise from CCDs, contaminating light from dust enveloping our own galaxy, Poisson noise from photon counts, and more. Furthermore, due to the need of obtaining large amounts of spectra, astronomers are forced to limit the time of integration for any given galaxy, resulting in low signal-to-noise measurements. As a consequence, not only it becomes difficult to confidently measure specific spectral features for secure redshift estimation, but we also incur the risk of misidentifying features - e.g., confusing a hydrogen line for an oxygen line - which results in so-called catastrophic outliers. Human evaluation mitigates a lot of these problems with current—relatively small - data sets. However, Euclid observations will be particularly challenging, working in very low signal-to-noise regimes and obtaining a massive amount of spectra, which will force us to develop automated methods capable of high accuracy and necessitating minimal human intervention.

Meanwhile, the rise of the "golden age" of Deep Learning [12] has fundamentally changed the way we handle and apprehend raw, unprocessed data. While the existing machine learning models heavily rely on the development of efficient feature extractors, a task non-trivial and very challenging, Deep Learning architectures are able to singlehandedly derive important characteristics from the data by learning intermediate representations and by structuring different levels of abstraction, essentially modelling the way the human brain works. The monumental success of Deep Learning networks in recent years, has been strongly enhanced by their interminable capacity to harness the power of Big Data and fully exploit emerging, cutting-edge hardware technologies, constituting one of the currently most widely used paradigms in numerous applications and in various scientific research fields.

One such a network subsists in Convolutional Neural Networks (CNNs) [13], a sequential model structured with a combination of Convolutional & Non-Linear Layers. The inspiration behind Convolutional Neural Networks resides in the concept of visual receptive fields [14], i.e., the region in the visual sensory periphery where stimuli can modify the response of a neuron. This is the main reason that CNNs initially found application in image classification, by learning to recognize images by experience, in the same perception where a human being can gradually learn to distinguish different image stimuli from one another. Today, CNNs are administered in the use of various types of data, with more or less complicated dimensional structures, with the key property of maintaining their spatial correlations without the need to collapse higher dimensional matrices into flattened vectors.

Our main motivation lies in the use of a state-of-the-art model, such as Convolutional Neural Networks, for an automated and reliable solution of the problem of spectroscopic redshift estimation. Estimating galaxy redshifts is perceived as a regression procedure, still a classification approach can be formulated without the loss of essential information. The robustness of the proposed model will be examined in two different data variations, as depicted in the example of Fig. 1. In the first case (b), we deploy randomly redshifted variations of the original rest-frame spectral profiles of the dataset used, substantially constituting linear translations of the rest-frame, in logarithmic scale. This is considered an idealistic scenario, as it ignores the interference of noise or presumes the existence of a reliable denoising technique. On the other hand, a more realistic case (c) is studied, with the available redshifted observations subjected to noise of realistic conditions.

The main contributions of our work are referenced below:

- We use a Deep Learning architecture for the case of spectroscopic redshift estimation, never used before for the issue at hand. To achieve that we need to convert the problem from a regression task, as engaged in general, to a classification task, as encountered in this novel approach.
- We utilize Big Data and evaluate the impact of a significant increase of the employed observations in the overall performance of the proposed methodology. The dataset used is modelled after one of the biggest upcoming spectroscopic surveys, the Euclid Mission [6].

The outline of this paper is structured as follows. In Section 2, we overview the related work in redshift estimation and Convolutional Neural Networks in general. In Section 3, we describe 1-Dimensional CNNs and we analyse the formulated methodology. In Section 4, we mainly focus on the dataset used and describe its properties. In Section 5, we present the experimental results, with accompanying discussion. Conclusions and future work are engaged in Section 6.

2 Related Work

Photometric observations have been extensively utilized in redshift estimation due to the fact that photometric analysis is substantially less costly and time-consuming, contrary to the spectroscopic case. Popular methods for this kind of estimation include Bayesian estimation with predefined spectral templates [15], or alternatively some widely used machine-learning models, adapted for this kind of problem, like the Multilayer Perceptron [16], [17] and Boosted Decision Trees [17], [18]. However, the limited wavelength resolution of photometry, compared to spectroscopy, introduces a higher level of uncertainty to the given procedures. In spectroscopy, by observing the full Spectral Energy Distribution (SED) of a galaxy, one can easily detect distinctive emission and absorption lines that can lead to a judicious redshift estimation, by measuring the wavelength shift of these spectral characteristics from the rest frame. Due to noisy observations, the main redshift estimation methods involve cross-correlating the SED with predefined spectral templates [19] or PCA decompisitions of a template library. Noisy conditions and potential errors due to the choice of templates are the main reason that most reliable spectroscopic redshift estimation methods heavily depend on human judgment and experience to validate automated results.

The existing Deep Learning models (i.e., Deep Artificial Neural Networks—DANNs) have largely benefited from the dawn of the Big Data era, being able to produce impressive results, that can match, or even exceed, human performance [20]. Despite the fact that training a DANN can be fairly computationally demanding, even more so while we increase its complexity and the data it needs to process, nevertheless, the rapid advancements on computational means and memory storage capacity have rendered feasible such a task. At the same time, and in contrast to the training process, the evaluation phase for a test dataset can be exceptionally fast, with a negligible execution time, regardless of its size. Currently, Deep Learning is considered to be the state-of-the-art in many research fields, such as image classification, natural language processing and robotic control, with models like Convolutional Neural Networks [13], Long-Short Term Memory (LSTM) networks [21], and Recurrent Neural Networks [22], dominating the research field.

The main idea behind Convolutional Neural Networks materialized for the first time with the concept of "Neocognitron", a hierarchical neural network capable of visual pattern recognition [23], and evolved into LeNet-5, by Yann LeCun et al. [13], in the following years. The massive breakthrough of CNNs (and Deep Learning in general) transpired in 2012, in the ImageNet competition [24], where the CNN of Alex Krizhevsky et al. [25], managed to reduce the classification error record by ¯10 percent, an astounding improvement at the time. CNNs have been considered in numerous applications, including image classification [25], [26] & processing [27], video analytics [28], [29], spectral imaging [30] and remote sensing [31], [32], confirming their dominance and ubiquity in contemporary scientific research. In recent years, the practice of CNNs in astrophysical data analysis has led to new breakthroughs, among others, in the study of galaxy morphological measurements and structural profiling through their surface's brightness [33], [34], the classification of radio galaxies [35], astrophysical transients [36] and star-galaxy separation [37], and the statistical analysis of matter distribution for the detection of massive galaxy clusters, known as strong gravitational lenses [38], [39]. The exponential increase of incoming data, for future and ongoing surveys, has led to a compelling need for the deployment of automated methods for large-scale galaxy decomposition and feature extraction, negating the commitment on human visual inspection and hand-made user-defined parameter setup.

3 Proposed Methodology

In this work, we study the problem of accurate redshift estimation from realistic spectroscopic observations, modelled after Euclid. Redshift estimation is considered to be a regression task, given the fact that a galaxy redshift (z) can be measured as a non-negative real valued number (with zero corresponding to the rest-frame). Given the specific characteristics of Euclid, we can focus our study in its redshift range of detectable galaxies. Subsequently, we can restrict the precision of each of our estimations to match the resolution of the spectroscopic instrument, meaning that we can split the chosen redshift range into evenly sized slots equal to Euclid's required resolution. Hence, we can transform the problem at hand from a regression task to a classification task, using a set of ordinal classes, with each class corresponding to a different slot, and accordingly we can utilize a classification model (Convolutional Neural Networks in our case) instead of a regression algorithm.

3.1 Convolutional Neural Networks

A Convolutional Neural Network is a particular type of Artificial Neural Network, which comprises of inputs, outputs and intermediate neurons, along with their respective connections, that encode the learnable weights of the network. One of the key differences between CNNs and other neural architectures, like Multilayer Perceptron [40], is that in typical

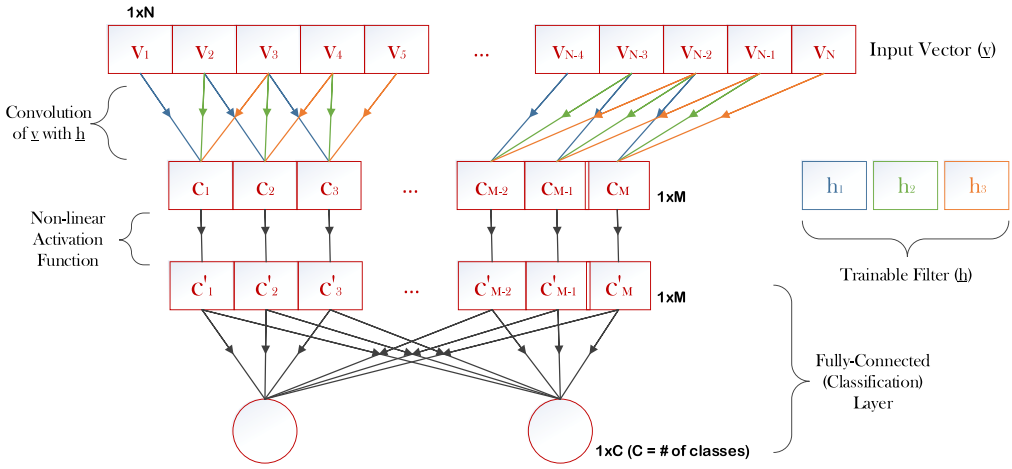


Fig. 2. Simple 1-Dimensional CNN. The input vector v is convolved with a trainable filter h (with a stride equal to 1), resulting in an output vector of size $M = N - 2$. Subsequently, a non-linear transfer function (typically ReLU) is applied element-wise on the output vector preserving its original size. Finally, a fully-connected, supervised layer is used for the task of classification. The number of the output neurons (C) is equal to to the number of the distinct classes of the formulated problem (800 classes in our case).

neural networks, each neuron of any given layer connects with every neuron of its respective previous and following layers (fully-connected layers). On the contrary, CNNs are structured in a locally-connected manner, exhibiting the spatial correlations of the given input, under the assumption that neighboring regions of each observation are more likely to be related than regions that are farther away. By reducing the number of total connections, CNNs can successfully manage to drastically decrease the number of trainable parameters, rendering the network less prone to overfitting.

3.1.1 Typical Architecture of a 1-Dimensional CNN

A typical 1D CNN (Fig. 2) is structured in a sequential manner, layer by layer, using a variety of different layer types. The foundational layer of a CNN is the *Convolutional Layer*. Given an input vector of size $1 \times N$ and a trainable filter $(1 \times K)$, the convolution of the two entities will result in a new output vector with a size $(1 \times M)$, where $M = N - K + 1$. The value of M may vary based on the stride of the operation of convolution, with bigger strides leading to smaller outputs. In the entirety of this paper, we assume the generic case of a stride value equal to 1.

The trainable parameters of the network (incorporated in the filter) are initialized randomly [41] and, therefore, are totally unreliable, but as the training of the network advances, through the process of backpropagation [42], they are essentially optimized and are able to capture interesting features from the given inputs. The parameters (i.e., weights) of a certain filter are considered to be shared [43], in the aspect that the same weights can be used throughout the convolution of the entirety of the input. This, can consequently lead to a drastical decrease in the number of weights, enhancing the ability of the network to generalize and adding to its total robustness against overfitting. To ensure that all different possible features can be captured, more than one filters can be actually utilized.

When addressing challenging problems, the use of shallow CNN architectures is insufficient, given their limited capacity to form deeper and complex representations of the input data. The development of deeper models, able to derive informative and detailed features, can become a necessity. The claim that an effective expansion of the CNN can be achieved by introducing more convolutional layers, one on top of another, is actually invalid. Given the linear property of the convolutional operation, the sequential stacking of all these convolutional layers could actually be accounted for as one merged linear transformation over the input data, thus rendering the formed architecture as shallow. To be able to effectively form deeper, more complex CNN models, a non-linearity needs to be introduced directly after each convolutional layer, enabling the network to act as a universal function approximator [44]. Typical choices for the non-linear function (known as *activation* or *transfer function*) include the logistic (sigmoid) function, the hyperbolic tangent (tanh) and the Rectified Linear Unit (ReLU). The most common choice in CNNs is ReLU ($f(x) = max(0, x)$) and its variations [45]. Compared to the cases of the sigmoid and hyperbolic tangent functions, the rectifier possesses the advantage that it is easier to compute (as well as its gradient) and is resistant to saturation conditions [25], rendering the training process much faster and less likely to suffer from the problem of vanishing gradients [46].

Finally, one or more *Fully-Connected Layers* are typically introduced as the final layers of the CNN, committed to the task of the supervised classification. A Fully-Connected Layer is the typical type of layer met in Multilayer Perceptron and as the name implies, all its neuronal nodes are connected with all the neurons of the previous layer, leading to a dense connectivity. Given the fact that the output neurons of the CNN correspond to the unique classes of the selected problem, each of these neurons must have a complete view of the highest-order features extracted by the deepest Convolutional Layer, meaning that they must be necessarily associated with each of these features.

The final classification step is performed using the multiclass generalization of Logistic Regression known as *Softmax Regression*. Softmax Regression is based on the exploitation of the probabilistic characteristics of the normalized exponential (softmax) function below:

$$h_\theta(x)_j = \frac{e^{\theta_j^T x}}{\sum_{k=1}^{W} e^{\theta_k^T x}}, \tag{1}$$

where x is the input of the Fully-Connected Layer, θ_j are the parameters that correspond to a certain class w_j and W is the total number of the distinct classes related to the problem at hand. It is fairly obvious that the softmax function reflects an estimation of the normalized probability of each class w_j, to be predicted as the correct class. As deduced from the previous equation, each of these probabilities can take values in the range of [0, 1] and, at the same time, they all need to add-up to the value of 1. This probabilistic approach composes a considerable reason for the transformation of the examined problem to a classification task, rendering possible to quantify the level of confidence for each estimation and providing a clearer view on what has been misconstrued in the case of misclassification.

The use of *Pooling Layers* has been excluded from the pipeline, given the fact that pooling is considered, among others, a great method of rendering the network invariant to small changes of the initial input. This is a very important property in image classification, but in our case these translations of the original rest-frame SEDs almost define the different redshifted states. By using pooling, we suppress these transformations, "crippling" the network's ability to identify each different redshift.

3.1.2 Regularizing Techniques

In very complex models, like ANNs, there is always the risk of overfitting the training data, meaning that the network produces over-optimistic predictions throughout the training process, but fails to generalize well on new data, subsequently leading to a decaying performance. The local neuronal connectivity that is employed in Convolutional Neural Networks and the concept of weight sharing, reported in previous paragraphs, cannot suffice in our case, given the fact that the single, final Fully-Connected Layer (which contains the majority of the parameters) will consist of hundreds of neurons. One way to address the problem of the network's high variance exists in the use of Big Data, with a theoretical total negation of the effects of overfitting, when the number of training observations tends to infinity. We will thoroughly examine the impact of the use on Big Data, on clean and noisy observations, in our experimental scenarios.

Dropout [47] and *Batch Normalization* [48] are, also, two very popular techniques in CNNs that can help narrow down the consequences of overfitting. In Dropout, the following simple, yet very powerful trick can be used to temporarily decrease the total parameters of the network at each training iteration. All the neurons in the network are associated with a probability value p (subject to hyper-parameter tuning) and each neuron, independently from the others, can be temporarily dropped from the network (along with all incoming and outgoing connections) with that probability. This is an iterative process, meaning that for each training sample of a training batch, a random portion of the entirety of the original network is dropped, leading to "thinner" and more degenerated variations of its initial structure as the value of the probability p grows bigger. Each layer can be associated with a different p value, meaning that Dropout can be considered as a per-layer operation, with some layers discarding neurons in a higher percentage, while others dropping neurons in a lower rate or not at all. During inference, the entirety of the network is used, meaning that Dropout is not applied at all.

Batch Normalization, on the other hand, can be accounted for more as a normalizer, but previous studies [48] have shown that it can work very effectively as a regularizer as well. Batch Normalization is, in fact, a local (per layer) normalizer that operates on the neuronal activations in a way similar to the initial normalizing technique applied to the input data in the pre-processing step. The primary goal is to enforce a zero mean and a standard deviation of one for all activations of the given layer and for each mini-batch. The main intuition behind Batch Normalization lies in the fact that as the neural network deepens, it becomes more probable that the neuronal activations of intermediate layers might diverge significantly from desirable values and might tend towards saturation. This is known as Internal Covariate Shift [48] and Batch Normalization can play a crucial role on mitigating its effects. Consequently, it can actuate the gradient descent operation to a faster convergence [48], but it can also lead to an overall highest accuracy [48] and, as stated before, render the network stronger and more robust against overfitting.

3.2 System Overview

In this subsection, we analyse the pipeline of our approach. Initially, we operate on clean rest-frame spectral profiles, each consisting of 3750 bins. These wavelength-related bins correspond to the spectral density flux value of each observation, for that certain wavelength range ($\Delta\lambda$ = 5 Å, λ = [1252.5, 20002.5] Å). Our first goal is to create valid redshifted variations using the formula:

$$log(1+z) = log(\lambda_{obs}) - log(\lambda_{emit}) \Leftrightarrow 1+z = \frac{\lambda_{obs}}{\lambda_{emit}}, \qquad (2)$$

where λ_{emit} is the original rest-frame wavelength, z is the redshift we want to apply and λ_{obs} is the wavelength that will ultimately be observed for the given redshift value. This formula is linear on logarithmic scale. For the conduction of our experiments, we work on the redshift range of $z = [1,\ 1.8)$, which is very similar to what Euclid is expected to detect. Also, to avoid redundant operations and to establish a simpler and a faster neural network we use a subset of the wavelength range of each redshifted example (instead of the entirety of the available spectrum), based on Euclid's spectroscopic specifications ($1.1 - 2.0\ \mu m \Leftrightarrow 11000 - 20000$ Å). That means that all the training and testing observations will be of equal size $\frac{20000-11000}{\Delta\lambda} = 1800$ bins.

For the "Regression to Classification" transition our working redshift range of $[1,\ 1.8)$ must be split into 800 non-overlapping, equally-sized slots resulting in a resolution of 0.001, consistent with Euclid expectations. Each slot will correspond to the related ordinal class (from 0 to 799), which in turn must be converted into the 1-Hot Encoding format to match the final predictions procured by the final Softmax Layer of the CNN. A certain real-valued redshift of a given spectral profile will be essentially transformed into the ordinal class that corresponds to the redshift slot it belongs to. Finally, for the predictions, shallower and deeper variations of a Convolutional Neural Network will be trained, with 1, 2 & 3 Convolutional (+ ReLU) Layers, along with a Fully-Connected Layer as the final Classification Layer.

TABLE 1
Comparison of CPU & GPU Training Running Time, in 3 Different Benchmark Experiments

Experiment #	CPU Time (per epoch)	GPU Time (per epoch)
1	75 sec.	11 sec.
2	735 sec.	107 sec.
3	158 sec.	20 sec.

In the 1^{st} and the 2^{nd} experiment, we utilize 40,000 and 400,000 training observations, of the idealistic case, in a CNN with 1 Convolutional Layer. In the 3^{rd} case, we deploy 40,000 realistic training examples for the training of a CNN with 3 Convolutional Layers.

4 A Deeper Perspective on the Data

The simulated dataset used is modeled after the upcoming Euclid satellite galaxy survey [6]. When generating a large, realistic, simulated spectroscopic dataset, we need to ensure that it is representative of the expected quality of the Euclid data. A first requirement is to have a realistic distribution of galaxies in several photometric observational parameters. We want the simulated data to follow representative redshift, color, magnitude and spectral type distributions. These quantities depend on each other in intricate ways, and correctly capturing the correlations is important if we want to have a realistic assessment of the accuracy of our proposed method. To that end, we define a master catalog for the analyses with the COSMOSSNAP simulation pipeline [49], which calibrates property distributions with real data from the COSMOS survey [50]. The generated COSMOS mock Catalog (CMC) is based on the 30-band COSMOS photometric redshift catalogue with magnitudes, colors, shapes and photometric redshifts for 538.000 galaxies on an effective area of $1.24\ deg^2$ in the sky, down to an i-band magnitude of ~ 24.5 [51]. The idea behind the simulation is to convert these real properties into simulated properties. Based on the fluxes of each galaxy, it is possible to select the best-matching SED from a library of predefined spectroscopic templates. With a "true" redshift and an SED associated to each galaxy, any of their observational properties can then be forward-simulated, ensuring that their properties correspond to what is observed in the real Universe.

For the specific purposes of this analysis, we require realistic SEDs and emission line strengths. Euclid will observe approximately 50 million spectra in the wavelength range 11000 - 20000 Å with a mean resolution $R = 250$, where $R = \frac{\lambda}{\Delta\lambda}$. To obtain realistic spectral templates, we start by selecting a 50 percent random subset of the galaxies that are below redshift $z = 1$ with Hα flux above $10^{-16}\ erg\,cm^{-2}\,s^{-1}$, and bring them to rest-frame values ($z = 0$). We then resample and integrate the flux of the best-fit SEDs at a resolution of $\Delta\lambda = 5$Å. This corresponds to $R = \frac{\lambda}{\Delta\lambda} = 250$ at an observed wavelength of 11000 Å, if interpreted in rest-frame wavelength at $z = 2$. For the purpose of our analysis, we will retain this choice, even though it implies higher resolution at larger wavelengths. Lastly, we redshift the SEDs to the expected Euclid range. In the particular case where we wish to vary the number of training samples, we generate more than one copy per rest-frame SED at different random redshifts. We will refer to the resampled, integrated, redshifted SEDs as "clean spectra" for the rest of the analysis.

For each clean spectrum above, we generate a matched noisy SED. The required sensitivity of the observations is defined in terms of the significance of the detection of the $H\alpha$ Balmer transition line: an unresolved (i.e., sub-resolution) $H\alpha$ line of spectral density flux $3 \times 10^{-16} erg\,cm^{-2}s^{-1}$ is to be detected at 3.5σ above the noise in the measurement. We create the noisy dataset by adding white Gaussian noise such that the significance of the faintest detectable $H\alpha$ line, according to the criteria above, is 1σ. This does not include all potential source of noise and contamination in Euclid observations, such as dust emission from the galaxy and line confusion from overlapping objects. We do not include these effects as they depend on sky position and galaxy clustering, which are not relevant to the assessment of the efficiency and accuracy of redshift estimation. Our choice of Gaussian noise models other realistic effects of the observations, including noise from sources such as the detector read-out, photon counts and intrinsic galaxy flux variations.

5 Experimental Analysis and Discussion

We implemented our Deep Learning model with the help of TensorFlow [52] and Keras [53] libraries, in Python code. TensorFlow is an open-source, general-purpose Machine Learning framework for numerical computations, using data flow graphs, developed by Google. Keras is a higher level Deep Learning-specific library, capable of utilizing TensorFlow as a backend engine, with support and frequent updates on most state-of-the-art Deep Learning models and algorithms. Both TensorFlow and Keras have the significant advantage that they can run calculations on the GPU, dramatically decreasing the computational time of the network's training, as depicted in Table 1. For the purpose of our experiments we used NVIDIA's GPU model, GeForce GTX 750 Ti.

As initial pre-experiments have shown, desirable values for the network's different hyperparameters are a kernel size of 8, a number of filters equal to 16 (per convolutional layer) and a stride equal to 1. Additionally, the Adagrad optimizer [54] has been employed for training, a Gradient Descent-based algorithm with an adaptable learning rate capability. The utilization of Adagrad can grant the network a bigger flexibility in the learning process, while at the same time it can become exempt from adjusting to an extra hyperparameter. Finally, in both idealistic and realistic cases, a simple normalization method has been used on all spectral profiles, to ensure their numerical compatibility with the CNN, while preserving their initial structure and integrity. The method is depicted in Equation (3), where X_{max} corresponds to the maximum spectral density flux value encountered in all examples (in absolute terms, given the noisy case), and $X_{original}$ is the initial value for each feature:

$$X_{normalized} = \frac{X_{original}}{2 \times X_{max}}. \tag{3}$$

5.1 Idealistic Observations

5.1.1 Impact of the Network's Depth

Our initial experiments revolve around the depth of the Convolutional Neural Network. We have used a fixed number of 400,000 training examples, 10,000 validation and 10,000 testing examples. Our aim is to examine the impact

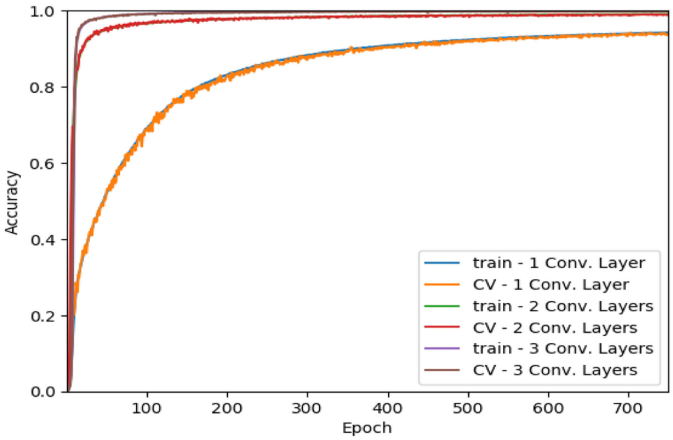


Fig. 3. Accuracy plot for the training & cross-validation sets, for 1,2 & 3 convolutional layers. The *x*-axis corresponds to the number of executed epochs. In all cases we used the same 400,000 Training Examples.

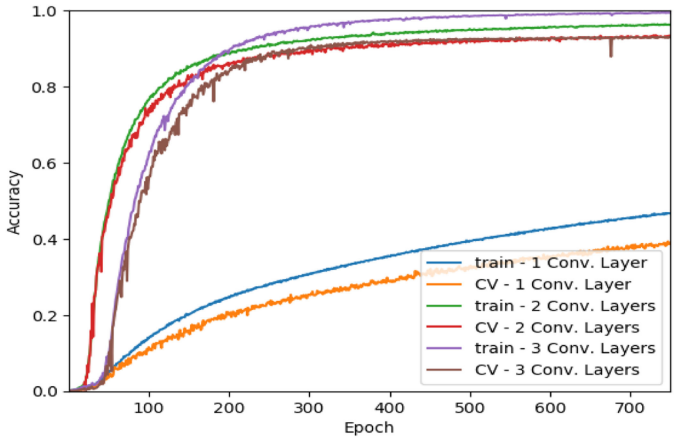


Fig. 5. Training & cross-validation accuracy, for 1,2 & 3 convolutional layers, using a significantly decreased amount of training observations (40,000). Overfitting is introduced, to various extents, based on each case.

of increasing the depth of the model, on the final predictive outcome. Specifically, we have trained and evaluated CNNs with 1,2 & 3 Convolutional Layers. In all cases, a final Fully-Connected Layer with 800 output neurons has been used for classification.

The metric of accuracy (correctly classified samples over the total number of samples) is the conventional way to measure the performance of a trained classifier during and after the training process. As the training goes by, we expect that the parameters of the network will start to adapt to the problem at hand, thus decreasing the total loss defined by the cost function with a consequent improvement on the accuracy percentage. In Fig. 3, we support this presumption by demonstrating the accuracy's rate of change over the number of training epochs. It can be easily derived that as a CNN becomes deeper, it is clearly more capable to form a reliable solution. Both 2 and 3-layered networks converge very fast and very close to the optimal case, with the latter narrowly resulting in the best accuracy. On the other hand, the shallowest network is very slow and significantly underperforms compared to the deeper architectures.

More information can be deduced in Fig. 4, where we compare for the shallowest and for the deepest case, and per testing example, the predicted redshift value outputed by the trained classifier versus the state-of-nature. Ideally, we want all the green dots depicted in each plot to fall upon the diagonal red line that splits the plane in half, meaning that all predicted outcomes coincide with the true values. As the green dots move farther away from the diagonal, the impact of the faulty predictions become more significant leading to the so called catastrophic outliers. A good estimator is characterized, not only by its ability to procure the best accuracy, but also by its capacity to diminish such extreme irregularities.

5.1.2 *Data-Driven Analysis*

In this setting, we will explore the significance of broad data availability in the overall performance of the proposed model. As mentioned before, Big Data have revolutionized the way Artificial Neural Networks perform [20], serving as the main fuel for their conspicuous achievements. Fig. 5 illustrates the behavior of the same network variations as in previous experiments (1, 2 and 3 Convolutional Layers),

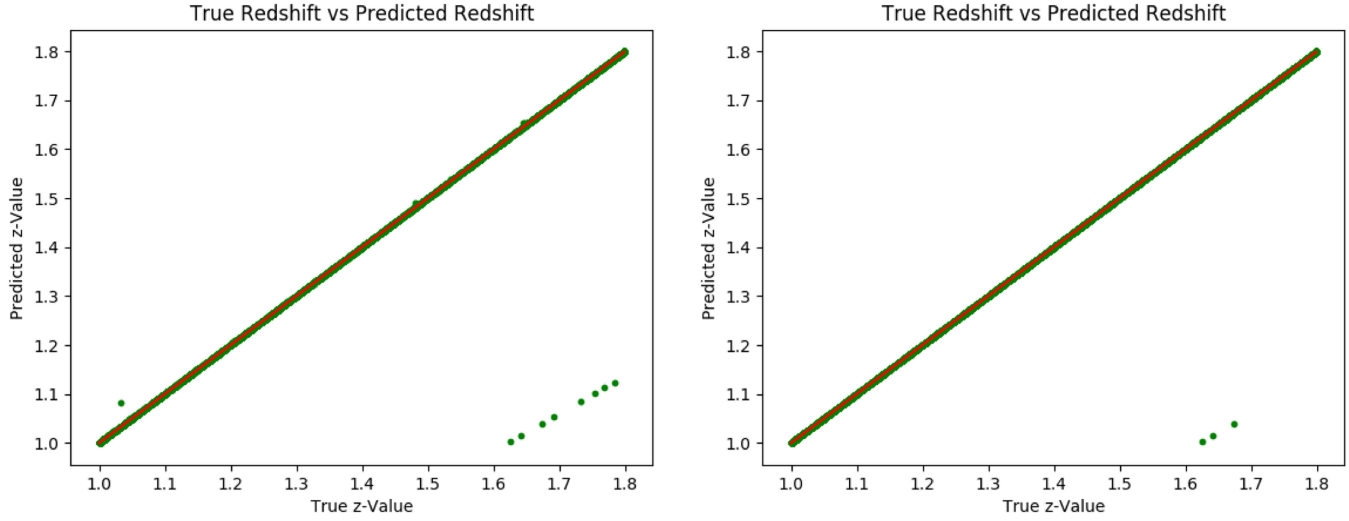


Fig. 4. Classification accuracy achieved by a CNN with one (left) and three (right) convolutional layers. The given scatter plots illustrate points in 2D space that correspond to the true class for each testing observation versus the predicted outcome of the corresponding classifier for that observation.

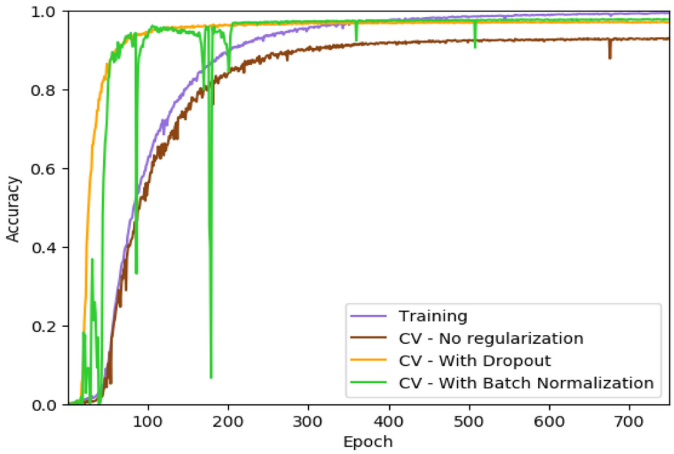


Fig. 6. The effects of regularization in a case where overfitting is introduced. For a dataset with 40,000 training observations and a CNN with 3 Convolutional Layers, we illustrate a best-case training accuracy (purple), that can eventually reach a 100 percent, versus the Cross-Validation performance with or without the utilization of a regularizer.

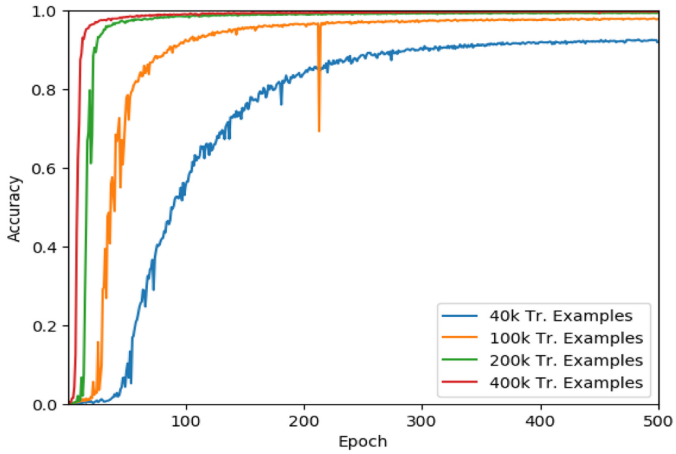


Fig. 7. Validation performance of a 3-layered network, using larger and more limited in size datasets. In all cases the training accuracy (not depicted here) can asymptotically reach 100 percent accuracy, after enough epochs.

using this time a notably more constrained, in size, training set of observations compared to the previous case. Specifically, we have lowered the number of training examples from 400,000 to 40,000, namely to one-tenth. Compared to the results we have previously examined in Fig. 3, we can evidently identify a huge gap between the performance of identical architectures trained with copious versus more limited amounts of data. It is adequately obvious that in all three cases overfitting is introduced, to various extents, leading to overoptimistic models that perform well in the training set, but with a decaying performance on the validation and the testing examples.

Fig. 6 demonstrates a characteristic scenario where the impact of the utilized regularizers is evaluated. Given that the deepest architecture with the 3 Convolutional Layers suffers the most from the effects of overfitting, we choose to present a best-case training accuracy for the 3-layered case, and for 40,000 training examples, versus the validation accuracy of the same architecture with and without the adoption of a regularizer. Even though both Batch Normalization and Dropout significantly improve the performance of the model, converging at the same time on proximal solutions, still, they fail to achieve a flawless performance compared to the training best-case. Regarding the Dropout parameter p, extensive experiments have shown that an approximate value of 0.5 resulted in the best performance. The contribution of lower values of p is less drastic, given that the decreased percentage of dropped neuronal units entails a smaller impact on the decrease of the network's total number of parameters. On the other hand, higher values of p (> 0.7) have the exact opposite effect, leading to an excessive elimination of units and parameters, rendering the network less effective in the end.

As a final step, we want to preserve the network's structural and hyperparametric characteristics immutable, whereas altering the amount of training observations utilized in each experimental recurrence. We have deployed a scaling number of training examples beginning from 40,000 observations, then to 100,000 and finally to 200,000 and 400,000 observations and we have used them to train a 3-layered CNN (3 Convolutional + 1 Fully-Connected Layers) without regularization, in all cases. As shown in Fig. 7, while we increase the exploited amount of data, the curve of the validation accuracy also increases in a smoother and steeper pace until convergence. On the contrary, when we use less data, the line becomes more unstable with a delayed convergence and a poorer final performance. It is very important to state, that despite the fact that the training accuracy can asymptotically reach, in all cases, 100 percent accuracy after enough epochs, the same doesn't apply for the validation accuracy (and respectively for the testing accuracy) with the phenomenon of overfitting taking its toll, mostly in the cases where the volume of the training data is not enough to handle the complexity of the network, failing to generalize in the long term.

5.1.3 Tolerance on Extreme Cases

Before advancing to noise-afflicted spectral profiles it is worthsome to investigate some extreme cases, concerning two astrophysical-related aspects of the data. As previously presented, one of our main novelties lies in the realization of the redshift estimation task as a classification task, guided by the specific redshift resolution that Euclid can achieve, and leading to the categorization of all possible detectable redshifts into 1 of 800 possible classes. As a first approach, we want to extend our working resolution to a double precision (from 0.001 to 0.0005), meaning that the existing redshift range of [1, 1.8) will be split into 1600 classes instead of 800.

As observed in Fig. 8, doubling the total number of possible classes has a non-critical impact in the predictive capabilities of our approach, given the fact that at convergence, the model results in a similar outcome for the two cases. Despite the fact that doubling the classes leads to a slower convergence, a behavior that can be attributed to the drastical increase of the parameters of the fully-connected layer, the network is still adequate enough to estimate successfully, in the long term, the redshift of new observations. Furthermore, as depicted in the scatter plot of the same figure, we can deduce that increasing the predictive resolution of the CNN, can lead to an increase in the total robustness of the model against catastrophic outliers, given the fact that none of the misclassified observations in the testing set, exists far from the diagonal red line, namely the optimal error-free case.

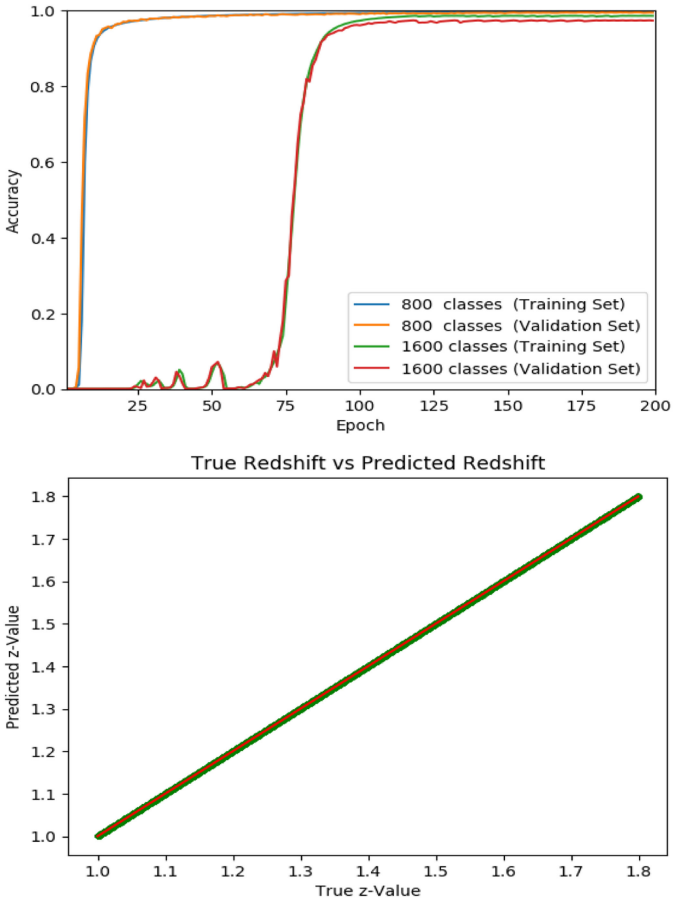


Fig. 8. Performance of a 3-Layered network trained with 400,000 training examples. In the first plot, we compare the cases where the redshift estimation problem is transformed into a classification task, with the use of 800 versus 1600 classes. In the second plot, we present the scatter plot of the predicted result versus the state-of-nature of the testing samples, only for the case of 1600 total classes.

In our second approach, we want to challenge the network's predictive capabilities, when presented with lower-dimensional data, and to essentially define which is the turning point where the abstraction of information becomes more of a strain, rather than a benefit. Having to deal with data that exist in high-dimensional spaces (like in the case of Euclid) can become more of a burden, rather than a blessing, as described by Richard Bellman [55] with the introduction of the very well-known term of the "curse of dimensionality". In our case, data dimensionality can be derived by splitting the operating wavelength of the deployed instrument into bins, where each bin corresponds to the spectral density flux value of the wavelength range it describes. Euclid operates in the range of $1.1 - 2.0\,\mu m$ with a bin size of $\Delta\lambda = 5\mathring{A}$, which implies 1800 different bins per observation. To reduce that number, we need to increase the wavelength range per bin by merging it with neighboring cells, namely by adding together their corresponding spectral density flux values. Essentially, we can assert that by lowering the dimensionality of data in this way, we can accomplish to concentrate existing information in cells of compressed knowledge, rather than discarding redundant information.

Fig. 9, actually supports our claim, leading to the conclusion, that when dealing with clean data, cutting down the number of total wavelength bins into more manageable

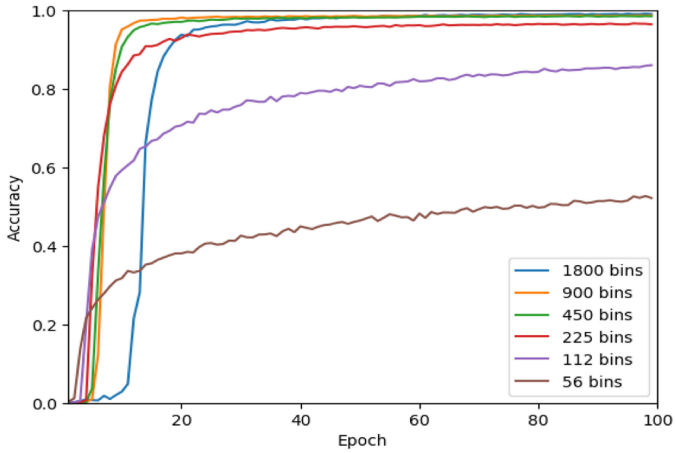


Fig. 9. Validation performance of a 3-Layered network trained with 400,000 training examples. We want to examine the behavior of the model, when trained with data of reduced dimensionalities.

numbers can result not only in an congruent performance with the initial model, but also into a faster convergence. On the other hand, oversimplifying the model can be deemed inefficacious, if we take into account the decline of the achieved accuracy in the three lower-dimensional cases. A moderate decline in the performance becomes visible in the case of 225 bins, with a more aggressive degeneration of the model in the rest of the cases.

5.2 Realistic Observations

Having to deal with idealistic data presumes the ambitious scenario of a reliable denoising technique for the spectra, prior the estimation phase. Although successful methods have been developed in the past [56], [57], our main aim is to integrate implicitly the denoising operation in the training of the CNN, meaning that the network should learn to distinguish the noise from the relevant information by itself, without depending on a third party. This way, an autonomous system can be established with a considerable robustness against noise, a strong feature extractor and, essentially, a reliable predictive competence. To that end, we have directly used noisy observations (described in Section 4) as the training input of the deployed CNNs.

A comparison between the idealistic and the realistic scenarios constitutes the first step that will lead to an initial realization of the difficulty of our newly set objective. In the illustrated Fig. 10, we observe that training a noise-based model with a number of observations that has proven to be sufficient in the clean-based case, leads to an exaggerated performance during the training process, that doesn't apply to newly observed spectra, thus leading to overfitting. Clean data are notably simpler in structure than their noisy counterparts, which in their turn are excessively diverge, meaning that generalization in the latter case is seemingly more difficult. The main intuition to battle this phenomenon lies in drastically increasing the spectral observations used in training. Feeding the network with bigger volumes of data can mitigate the effects of overfitting, given the fact that, despite it creates a specialized solution fitted for the set of observed spectra, this set tends to become so large that it befits the general case. This intuition is strongly supported by Fig. 11, where we compare the performance of identical

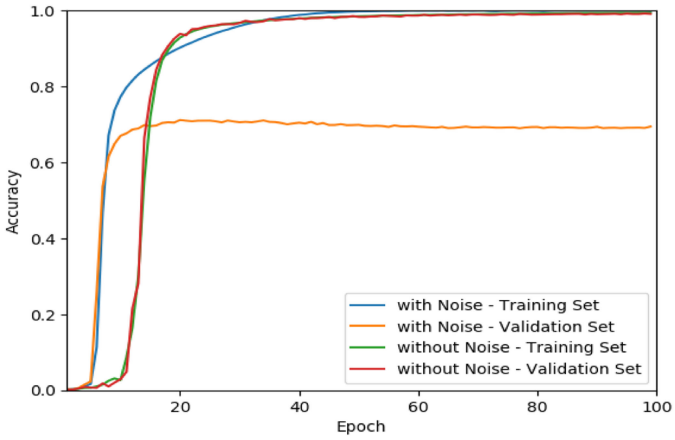


Fig. 10. Comparison of the model's performance, trained with clean and with noisy data (400,000 in both cases). The 3-layered neural network utilizes the same hyperparameters in both cases, without any form of regularization.

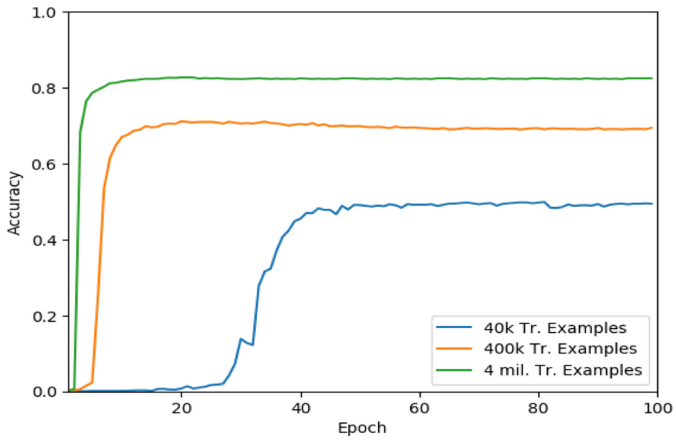


Fig. 11. Accuracy on the validation set for different sizes of the training set (realistic case). No regularization has been used.

models when trained with different-sized sets. Preserving constant hyperparameters and not utilizing any form of regularization, we can derive that just by increasing in bulk the total amount of data, the network's generalization capabilities also increase in a scalable way. Finally, the new difficulties established by the noisy scenario, also become highly apparent while observing the results of Fig. 12. The drastical increase in the number of misclassified samples is more than obvious, subsequently leading to an abrupt rise in the amount and variety of the different catastrophic outliers. Nevertheless, the faulty predictions that lie approximate to the corresponding ground truths constitute the majority of the mispredictions, as verified by the highly populated green mass around the diagonal red line in the scatter plots, and the highest bar column bordering the zero value in the case of the histograms.

5.2.1 *Impact of Regularization*

The effects of regularization are illustrated in Fig. 13, in two different settings, one with a Training Set of 400,000 examples and another with a Training Set of 4,000,000 examples. For Batch Normalization, we inserted an extra Batch-Normalization Layer after each Convolutional Layer (and after ReLU). Although in literature [48], the use of Batch Normalization is proposed before the non-linearity, in our case extensive experimental results suggested otherwise. The operation of Dropout was utilized only in the Fully-Connected Layer. It is important to note that it can be also included in the Convolutional Layers, but our experimental efforts did not demonstrate a mentionable change in the final performance of the network when Dropout was introduced in all of its layers. Given that the vast majority of the network's trainable parameters is concentrated in the Fully-Connected Layer, it is expected that the employment of Dropout in this layer will be of a significantly greater impact compared to its adoption in the Convolutional Layers. Moreover, regarding the Dropout parameter p, an intermediate value of 0.5 appeared to yield the optimal solution compared to lower or larger values, in accordance with the elaborated intuition presented in Section 5.1.2 for the idealistic case.

As depicted in both examined plots of Fig. 13, Dropout ($p = 0.5$) can moderately help enhance the network's performance in the validation set, leading to an increase in the accuracy by ~1.5 percent in the best case and ~0.5 percent in the worst case, compared to the scenario where no regularization is used. On the other hand, Batch Normalization appears to have a bigger regularizing effect in improving the accuracy of the employed model, resulting in a notable increase by almost 10 percent in the validation set when trained with 400,000 training examples, but also in a significantly lower gain of ~2 percent when the training observations are 4,000,000. In the latter case, even though Batch Normalization still leads to the best performance, its difference compared to Dropout has clearly diminished. Finally, we can observe that as the number of initial training examples increases, the impact of the utilization of a regularizing technique is mitigated. Even though, most of the times, a carefully chosen regularizer can meaningfully help battle the phenomenon of overfitting, an abundant increase in the number of training examples is a necessity if we want the testing performance to approach the ideal 100 percent accuracy.

5.3 Comparison with Other Classifiers

In this subsection, we present a comparison between the best-case performance of the proposed CNN and the performance of other popular classifiers, namely k Nearest Neighbours (kNN) [58], Random Forests [59] and Support Vector Machines (SVM) [60]. The bar plots in Fig. 14 corroborate the claim that Convolutional Neural Networks reign supreme as the most effective algorithm for the issue at hand, in all the cases that have been examined.

Starting with kNN, a best-case choice for the hyperparameter of k materialized in the case of $k = 1$. Greater values for k (≥ 3) resulted in a noteworthy decrease in the performance, for both idealistic and realistic cases, all leading to similar accuracies. Furthermore, it is important to state that the utilization of typical distance functions such as the Euclidean and the Manhattan, resulted in an inferior performance compared to the optimal and reported choice of the Chebyshev distance ($\max_i(|x_i - y_i|)$, where x, y are different galaxy spectral profiles). This behavior can be clearly attributed to the fact that for the distinction between two different redshift states, we exclusively care about the relative bin position of the characteristic emission and absorption lines

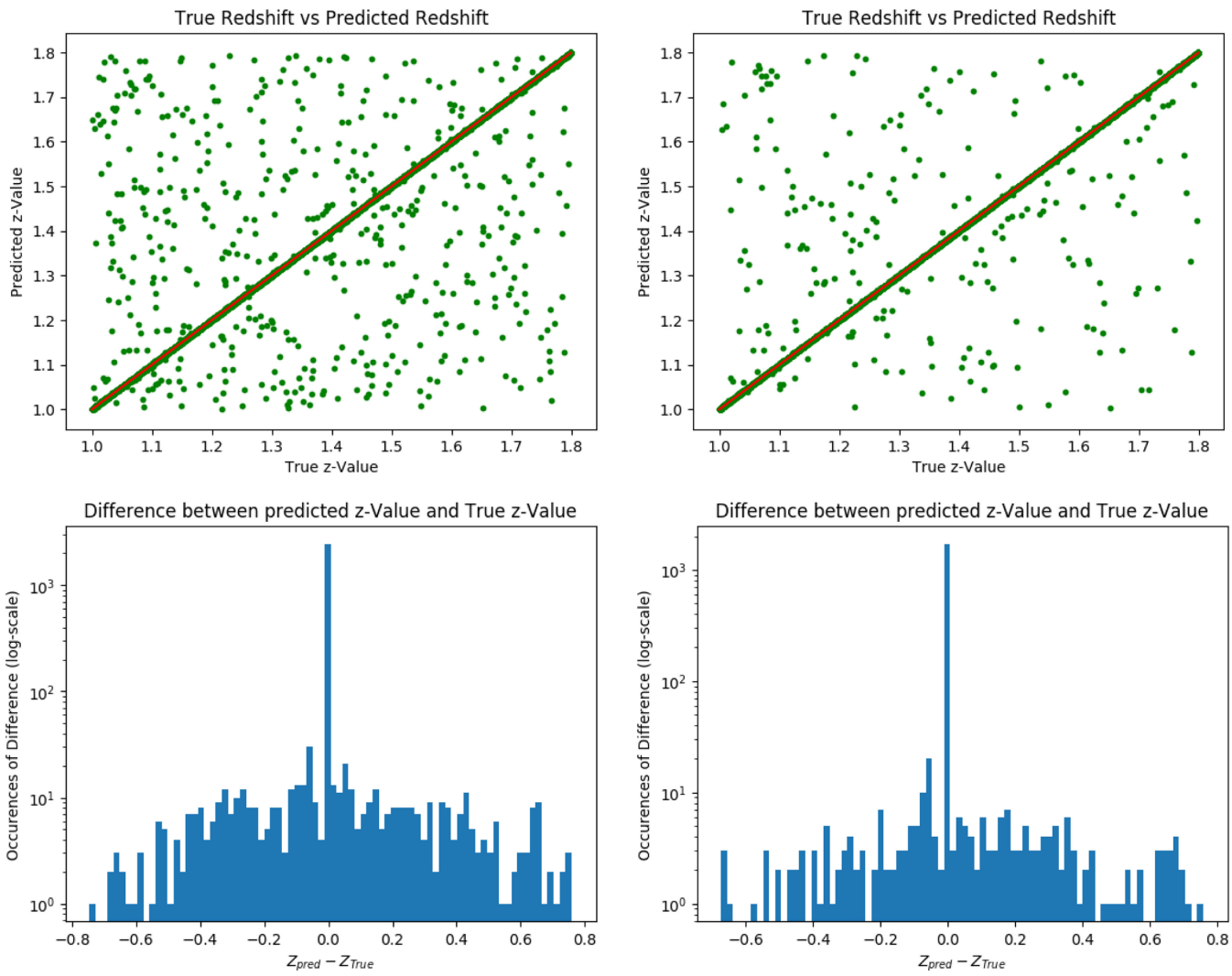


Fig. 12. Classification scatter plots and histograms for the realistic case, for 3-layered networks trained with 400,000 Training Examples (column a) and 4,000,000 Training Examples (column b). The depicted histograms represent the actual difference in distance (positive or negative) between misclassified estimated redshifts and their corresponding ground truth value versus the frequency of occurrence, in logarithmic scale, for each case.

(most importantly the $H\alpha$ line) of two spectra, potentially ignoring the flux values of bins that do not encode peaks for neither spectra. Given that in Chebyshev we preserve only the maximum of distances between corresponding bin values of pairs of observations, we expect that observations of equivalent redshifts will consist of peaks in concurring wavelength bins leading, essentially, to small maxima. On the other hand, for spectral profiles with different redshifts, the existence of a peak for one of the spectra implies the absence of a peak in the same bin for the other, resulting in a high-valued distance associated with the bin and, ultimately, in a larger maximum. Finally, the abrupt decrease in the performance of kNN in the case of realistic examples can be potentially justified by taking into account the diversity of the noise-afflicted spectra. The concealment of the significant emission and absorption lines due to the noise, in conjunction with the lack of a feature extractor, can consequently lead to severe under-performance issues.

In the case of Random Forests, our experimental findings unveiled an approximate value of 100 to be the optimal choice for the number of trained estimators. Larger values resulted in a significant increase in the computational time-complexity of the model without an additional increase in its accuracy. On the other hand, considerably smaller values were proved to be insufficient leading to a fairly poorer performance. Another significant observation, that can be deduced from the results in Fig. 14, lies in the surprisingly underachieving overall performance of Random Forests in both realistic and idealistic scenarios. By default, the algorithm of Decision Trees [61], and by extension that of the Random Forests, constructs its decision choices based on the values of the different attributes. This is an undesirable property in the case of redshift estimation, given that the categorization of each galaxy profile in its respective redshift class is not based on the sheer amplitude of the different attribute values (i.e., the values of each bin) per se, but on the relative values between neighboring wavelength bins, in such a manner, that will conclude to the presence of peaks. The existence of such a peak, in a wavelength bin with a certain flux value for a spectral observation might not necessarily imply the existence of another peak, in the same bin for a different observation with an equivalent flux value.

Lastly, the main competitor of Convolutional Neural Networks in both idealistic and realistic scenarios stands in the

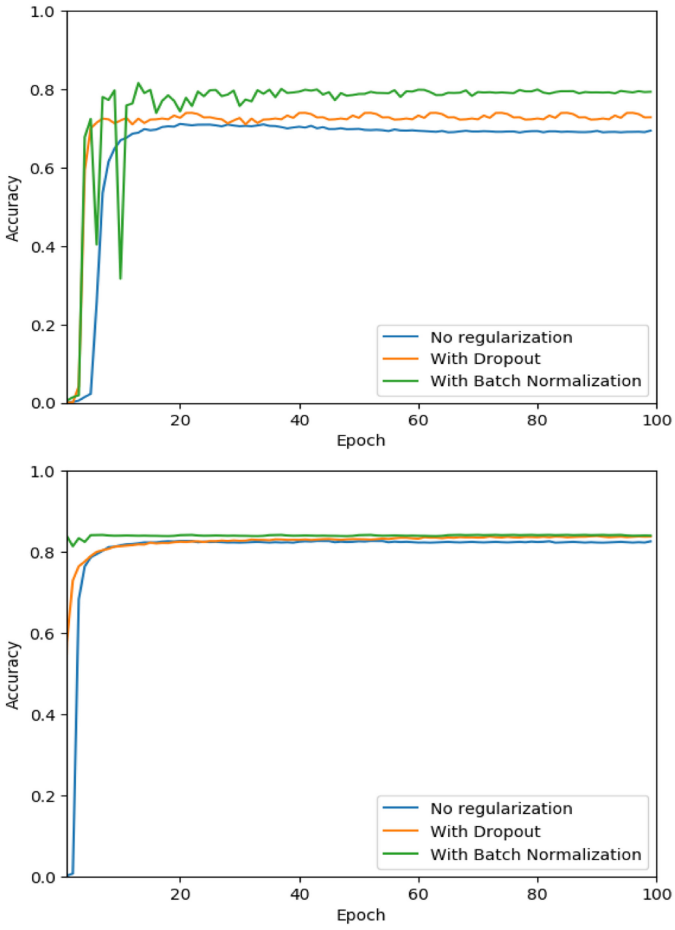


Fig. 13. Impact of regularization in the validation performance, in regard with the size of the employed training set. In the upper plot, a network trained with 400,000 observations is illustrated, while in the lower plot 4,000,000 training examples have been utilized. In all examined cases, the training accuracy (not depicted here) can asymptotically reach 100 percent accuracy in less than 50 epochs.

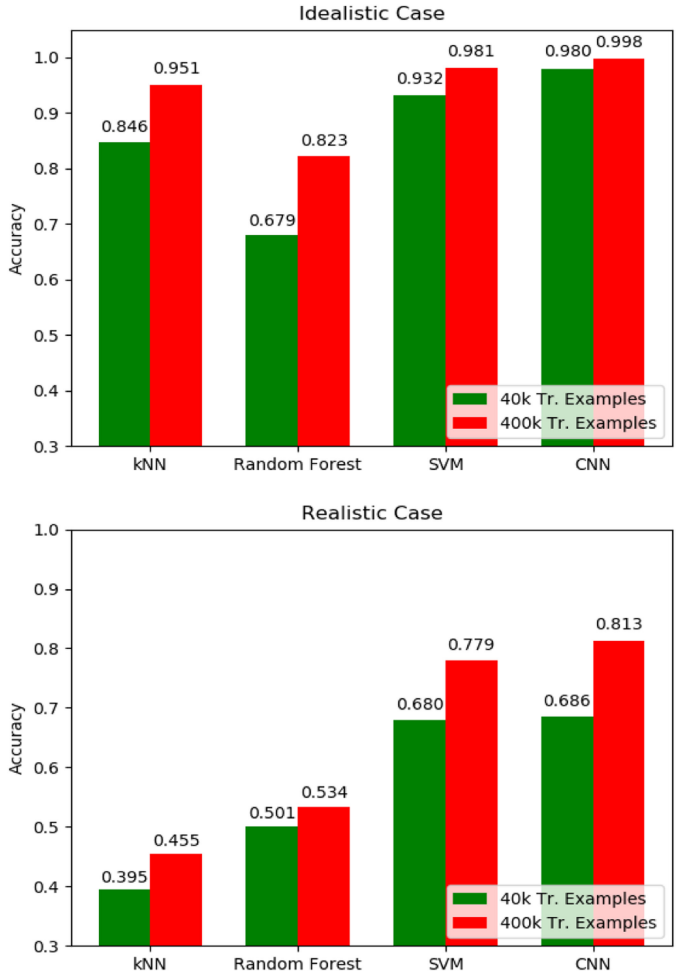


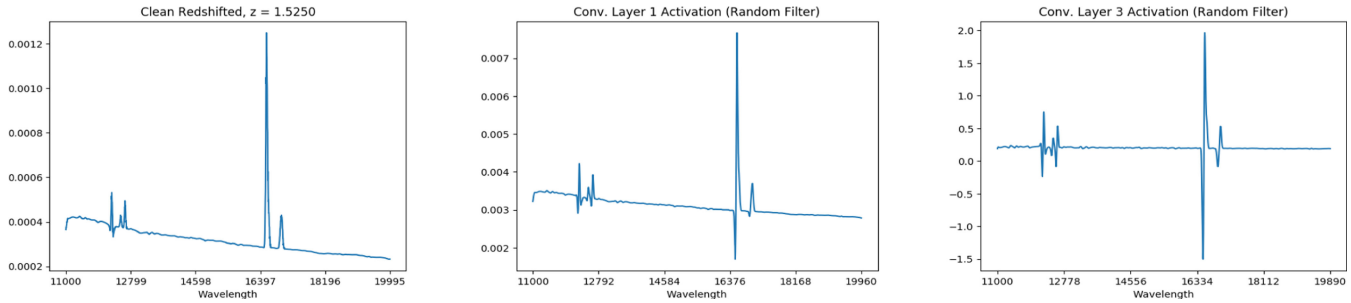
Fig. 14. Comparison bar plots for the k nearest neighbours, random forests, support vector machines and convolutional neural networks algorithms. For each classifier, we present the best-case performance on the test set, in both idealistic and realistic cases, when trained with a limited and with an increased amount of training data.

case of the Support Vector Machines. SVMs are considered to be highly performing and computationally efficient classifiers, which however, are mostly suited for binary classification problems or in cases where the total amount of distinct classes is limited. Regarding the problem of spectroscopic redshift estimation, even though the adopted best-case SVM can conclude to rather satisfying results, the proposed utilization of 800 different redshift classes renders its use as computationally inexpedient. With such an overwhelming amount of possible classes to predict, either techniques of "one-vs-rest" [62] and "one-vs-one" multi-class classification require the need of training 800 and $(800 \times 799 / 2) = 319{,}600$ individual classifiers respectively, leading to a computational time-complexity that is considerably greater compared to that of the proposed CNN (which, at the same time, performs better than the SVM). As a final note, exhaustive tuning for the different hyperparameters of SVM suggested the utilization of a Gaussian Radial Basis Function (RBF) kernel, in combination with a penalty parameter C of 10^8 and a kernel coefficient γ of 10^{-1}.

5.4 Intermediate Representations

In this section, we will briefly examine the undergoing transformation of the input data, as they flow deeper into the network. As discussed in previous sections, Convolutional Neural Networks are excellent feature extractors, able to distill significant knowledge from raw input data even when partially afflicted with different levels of noise. The hierarchical intermediate representations that a well-trained CNN can construct are able to capture complex and important spatial correlations of the given input. The information extracted by the convolutional operations can, then, be furtherly processed and examined in the Fully-Connected Layer and in association with the anticipated target class, which in the case of training is the state-of-nature.

In essence, a galaxy spectral profile can be inherently modeled as the combination of three fundamental components, emission and absorption lines (peaks), the spectral continuum and noise. The line information can be further decomposed taking into account their unique characteristics, such as shapes, heights and grouping. Excluding the noise component, which is later introduced in the case of the realistic scenario, we can clearly observe in the illustrated Fig. 15 that the salient effect of randomly chosen filters of the adopted network's initial and latter Convolutional Layers is the gradual removal of the spectral continuum. The elimination of the continuum is a key step in most existing spectroscopic redshift analyses [19], [56], [63], [64], given that distinguishing



(a) Clean Redshifted Spectral Profile (b) Activation of 1^{st} Conv. Layer (c) Activation of 3^{rd} Conv. Layer

Fig. 15. A random testing example (clean case) and the corresponding activations of the 1st and the 3rd convolutional layers.

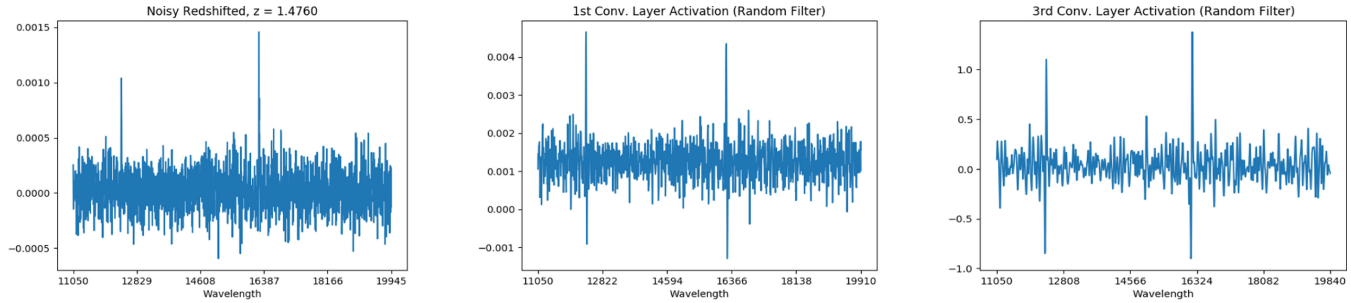
the various emission and absorption lines, their magnitudes and their relative positions, constitutes a stronger constraint for the task at hand, compared to identifying the shape of the continuum. By preserving the aforementioned characteristics, the convolutional feature extractor of the CNN can pave the way for the Fully-Connected Layer to process and exploit only the significant information encapsulated in the initial input, minus the distraction caused by irrelevant features. The introduction of mirror amplitudes in the depicted Fig. 15, that usually extend to the negative half-plane in the case of the final Convolutional Layer, cannot be deemed of significant importance given their immediate attenuation, or even complete nullification, by the succeeding ReLU.

With the inclusion of the noise scenario, we are engaged with new intricate challenges, given that the majority of the characteristic spectral lines of a galaxy profile can be obscured by the added noise or, in a worse case, false noise lines can be introduced. Thus, the emergence of high levels of noise can prevent the detection of existing true features or, on the other hand, it can lead to an identification of noise features, mistaken for true spectral lines. In an attempt to address this phenomena and try to separate the wheat from the chaff, we can observe in the depicted Fig. 16, that even though the outright removal of the interfering noise may not be easily achievable, given the low signal-to-noise ratio of the observed spectrum, nevertheless, the adopted CNN is able to perform a partial denoising of the examined spectral profile, gradually isolating the desired true peaks from the faulty discontinuities. As the number of the identified peaks improves and the mitigation of noise leads to a progressive increase in the network's certainty that they are not a fabrication of the contamination of noise, all the more we can ascertain that its predicted estimations will eventually become more credible.

5.5 Levels of Confidence

The transformation of the redshift estimation problem from a regression to a classification procedure, apart from enabling the utilization of a potent deep learning classifier, it also provides the benefit of associating each estimation with a level of confidence of the network's certainty that the predicted outcome corresponds to the true redshift value. In an ideal case, not only the network would deduce all the correct predictions during inference, but it would do so with a 100 percent probability output for the state-of-nature of each testing observation, versus a 0 percent for the remainder of the classes. In a more tangible scenario, we would expect that the network would suffer from an inconsequential amount of faulty predictions and, at the same time, it would associate strong and confident probabilities with the correct, congruent to the ground-truth, predictions and negligible probabilities with the remaining classes.

A characteristic example of the derived levels of confidence for the best-case CNNs in the idealistic and realistic scenarios is illustrated in Fig. 17. It is important to note that the reported results correspond to the predicted dominant class for each testing observation, with the term dominant referring to the class with the highest probability output, not necessarily being the ground-truth class. In the idealistic case, we can observe that the employed CNN is highly confident about the validity of its predictions leading to a very steep cumulative curve in the transition from the 90 percent probability to 100 percent. As also verified by the corresponding histogram, most predictions are associated with a very high probability that lies in the range of [0.9, 1], with a significantly decreased frequency of occurrences as the levels of confidence drop. This is a very desirable property, given the fact that we want the network to be certain about its designated choice, leading to concrete estimations that are not subject to dispute.



(a) Noisy Redshifted Spectral Profile (b) Activation of 1^{st} Conv. Layer (c) Activation of 3^{rd} Conv. Layer

Fig. 16. A random testing example (noisy case) and the corresponding activations of the 1st and the 3rd convolutional layers.

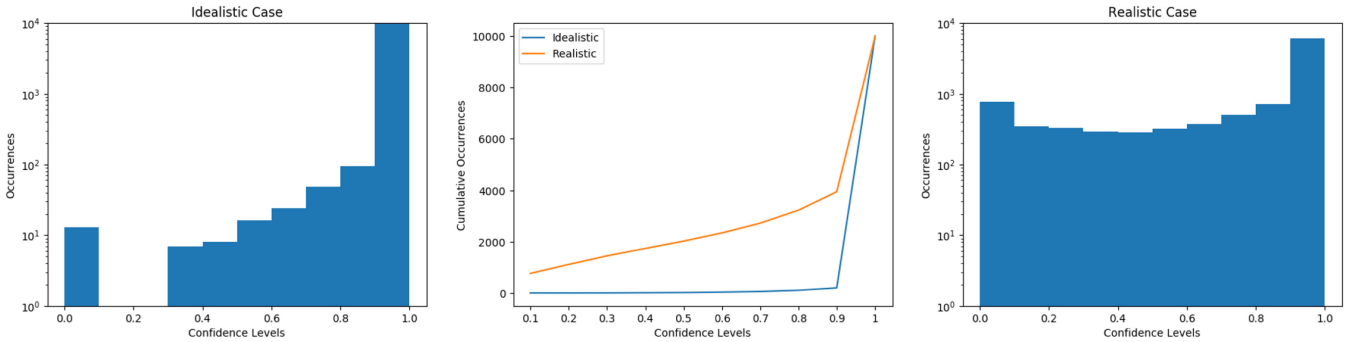


Fig. 17. The different levels of confidence associated with each prediction (and derived by softmax) on the Testing Set. The middle plot depicts the cumulative occurrences per level of confidence, for the idealistic and the realistic case. For example, the *y*-axis value that corresponds to the x-value of 0.4 represents the number of testing observations that obtain a predictive output for the dominant class, which is less than or equal to a probability of 0.4. The dominant class refers to the class with the highest probability output and might not necessarily be the ground-truth class. Additionally, the left (idealistic case) and right (realistic case) histograms exhibit a similar scenario, but not in a cumulative form and in a logarithmic scale for the number of occurrences.

Considering that the best-case CNN can achieve an accuracy percentage of 99.8 percent in the Testing Set, which in turn consists of 10,000 samples, we can infer that in the low-confidence domain of $[0, 0.5)$ mostly lie the cases where the model misclassifies (as also verified by manual inspection).

The addition of noise in the realistic scenario can undoubtedly complicate the efforts for reliable and confident predictions, given the transition from clear and easily distinguishable features, to obscured physical lines and falsely introduced peaks. Fig. 17 validates the assumption of an expected significant decrease in the confidence of the adopted network, but nevertheless, we can ascertain that the predicted probabilities that lie in the high confidence domain of $[0.5, 1]$ still remain dominant in quantity compared to the less confident examples. Given the test accuracy percentage of 81.3 percent of the best-case CNN, we can safely infer that the majority of tentative predictions mainly corresponds to misclassified low-SNR observations where a reliable distinction between true physical lines and their spurious noisy counterparts is fairly improbable, despite the attempts of the network to extract relevant representations. On the other hand, predictions that lie in the intermediate or the highest confidence domains account for adequately denoised spectral observations where the mitigation of noise has resulted in a, more or less, reliable discrimination between significant and counterfeit features, in accordance to what has already been discussed in Section 5.4.

6 Conclusion and Future Work

In this paper, we proposed an alternative solution for the task of spectroscopic redshift estimation with its transformation from a regression to a classification problem. We adopted a variation of the typical architecture of an Artificial Neural Network, commonly known as a Convolutional Neural Network, and we thoroughly examined its estimating capabilities for the issue at hand in various settings. We did so, using big volumes of training observations that fall into the category of the so called Big Data. Experimental results unveiled the great potential of this radically new approach in the field of spectroscopic redshift analysis and triggered the need for a deeper study concerning Euclid and other spectroscopic surveys.

To that end, our future work will be concentrated in the following research efforts. In the case of Euclid observations, some new noise patterns can be introduced that will complement the existing realistic noise-scenario. Additionally, a multimodal architecture can be established [65], able to derive aggregated knowledge from both photometric (images) and spectroscopic (spectral profiles) modalities of the different Euclid instruments. Moreover, newly developed and advanced deep learning architectures can be adopted (e.g., DenseNet [66]), customized for the 1-Dimensional case. Our end and foremost goal is the development of a pre-trained robust predictive model, built with a vast amount of simulated realistic observations. Subsequently, the extracted knowledge from this massive dataset can be reused and enhanced, by exploiting a significantly smaller dataset of real Euclid-collected labeled observations for the re-training and fine-tuning of the pre-trained model. Eventually, this optimized model can be employed as an automated means to effectively address the task of redshift estimation on future incoming and unlabeled Euclid observations. As a final note, another potential avenue of applications involves other spectroscopic surveys. The Dark Energy Spectroscopic Instrument (DESI) [67] is one of the major upcoming cosmological surveys currently under construction and installation in Kitt Peak, Arizona. It will operate in different wavelengths and under different observational and instrumental conditions compared to Euclid, and consequently will be able to detect galaxies with different redshift properties.

Acknowledgments

This work was partially funded by the DEDALE project, contract no. 665044, within the H2020 Framework Program of the European Commission.

References

[1] G. Bertone, *Particle Dark Matter: Observations, Models and Searches*. Cambridge, U.K.: Cambridge Univ. Press, 2010.
[2] E. J. Copeland, M. Sami, and S. Tsujikawa, "Dynamics of Dark Energy," *Int. J. Modern Phys. D*, vol. 15, pp. 1753–1935, 2006.
[3] G. Marcy, R. P. Butler, D. Fischer, S. Vogt, J. T. Wright, C. G. Tinney, and H. R. Jones, "Observed properties of exoplanets: masses, orbits, and metallicities," *Progress Theoretical Phys. Suppl.*, vol. 158, pp. 24–42, 2005.
[4] Planck Collaboration, P. A. R. Ade, N. Aghanim, M. Arnaud, M. Ashdown, J. Aumont, C. Baccigalupi, A. J. Banday, R. B. Barreiro, J. G. Bartlett, et al., "Planck 2015 results. XIII. Cosmological parameters," *A&A*, vol. 594, Sep. 2016, Art. no. A13.

[5] W. J. Borucki, D. Koch, G. Basri, N. Batalha, T. Brown, D. Caldwell, J. Caldwell, J. Christensen-Dalsgaard, W. D. Cochran, E. DeVore, E. W. Dunham, A. K. Dupree, T. N. Gautier, J. C. Geary, R. Gilliland, A. Gould, S. B. Howell, J. M. Jenkins, Y. Kondo, D. W. Latham, G. W. Marcy, S. Meibom, H. Kjeldsen, J. J. Lissauer, D. G. Monet, D. Morrison, D. Sasselov, J. Tarter, A. Boss, D. Brownlee, T. Owen, D. Buzasi, D. Charbonneau, L. Doyle, J. Fortney, E. B. Ford, M. J. Holman, S. Seager, J. H. Steffen, W. F. Welsh, J. Rowe, H. Anderson, L. Buchhave, D. Ciardi, L. Walkowicz, W. Sherry, E. Horch, H. Isaacson, M. E. Everett, D. Fischer, G. Torres, J. A. Johnson, M. Endl, P. MacQueen, S. T. Bryson, J. Dotson, M. Haas, J. Kolodziejczak, J. Van Cleve, H. Chandrasekaran, J. D. Twicken, E. V. Quintana, B. D. Clarke, C. Allen, J. Li, H. Wu, P. Tenenbaum, E. Verner, "Kepler Planet-Detection Mission: Introduction and First Results," *Sci.*, vol. 327, Feb. 2010, Art. no. 977.

[6] R. Laureijs, J. Amiaux, S. Arduini, J.-L. Augueres, J. Brinchmann, R. Cole, M. Cropper, C. Dabin, L. Duvet, A. Ealet, et al., "Euclid definition study report," *Euclid Consortium*, European Space Agency, *arXiv:1110.3193*, no. 1.1, 2011.

[7] P. A. Abell, J. Allison, S. F. Anderson, J. R. Andrew, J. R. P. Angel, L. Armus, D. Arnett, S. J. Asztalos, T. S. Axelrod, S. Bailey, et al., "Lsst science book, version 2.0," *arXiv preprint arXiv:0912.0201*, 2009.

[8] R. Bryant, R. H. Katz, and E. D. Lazowska, "Big-data computing: Creating revolutionary breakthroughs in commerce, science and society," A white paper prepared for the Computing Community Consortium committee of the Computing Research Association, 2008. [Online]. Available: http://cra.org/ccc/resources/ccc-led-whitepapers/

[9] Z. D. Stephens, S. Y. Lee, F. Faghri, R. H. Campbell, C. Zhai, M. J. Efron, R. Iyer, M. C. Schatz, S. Sinha, and G. E. Robinson, "Big data: Astronomical or genomical?" *PLoS Biol.*, vol. 13, no. 7, 2015, pp. e1002195.

[10] G. Efstathiou, W. J. Sutherland, and S. Maddox, "The cosmological constant and cold dark matter," *Nature*, vol. 348, no. 6303, pp. 705–707, 1990.

[11] R. Massey, T. Kitching, and J. Richard, "The dark matter of gravitational lensing," *Reports Progress Phys.*, vol. 73, no. 8, 2010, Art. no. 086901.

[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[13] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

[14] H. K. Hartline, "The response of single optic nerve fibers of the vertebrate eye to illumination of the retina," *Amer. J. Physiology–Legacy Content*, vol. 121, no. 2, pp. 400–415, 1938.

[15] N. Benitez, "Bayesian photometric redshift estimation," *Astrophysical J.*, vol. 536, no. 2, 2000, Art. no. 571.

[16] C. Bonnett, "Using neural networks to estimate redshift distributions. an application to cfhtlens," *Monthly Notices Roy. Astron. Soc.*, vol. 449, no. 1, pp. 1043–1056, 2015.

[17] I. Sadeh, F. B. Abdalla, and O. Lahav, "Annz2: Photometric redshift and probability distribution function estimation using machine learning," *Publications Astron. Soc. Pacific*, vol. 128, no. 968, 2016, Art. no. 104502.

[18] D. W. Gerdes, A. J. Sypniewski, T. A. McKay, J. Hao, M. R. Weis, R. H. Wechsler, and M. T. Busha, "Arborz: photometric redshifts using boosted decision trees," *Astrophysical J.*, vol. 715, no. 2, 2010, Art. no. 823.

[19] K. Glazebrook, A. R. Offer, and K. Deeley, "Automatic redshift determination by use of principal component analysis i: Fundamentals," *Astrophysical J.*, vol. 492, pp. 98–109, 1998.

[20] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016, http://www.deeplearningbook.org

[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," in *Spin Glass Theory and Beyond: An Introduction to the Replica Method and Its Applications*. Singapore: World Scientific, 1987, pp. 411–415.

[23] K. Fukushima, "Neocognitron: A hierarchical neural network capable of visual pattern recognition," *Neural Netw.*, vol. 1, no. 2, pp. 119–130, 1988.

[24] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-conv olutional-neural-networks.pdf

[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014.

[27] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4353–4361.

[28] G. Tsagkatakis, M. Jaber, and P. Tsakalides, "Goal!! event detection in sports video," *Electron. Imaging*, vol. 2017, no. 16, pp. 15–20, 2017.

[29] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1725–1732.

[30] K. Fotiadou, G. Tsagkatakis, and P. Tsakalides, "Deep convolutional neural networks for the classification of snapshot mosaic hyperspectral imagery," *Electron. Imaging*, vol. 2017, no. 17, pp. 185–190, 2017.

[31] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sens.*, vol. 7, no. 11, pp. 14 680–14 707, 2015.

[32] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sens.*, vol. 2015, 2015, Art. no. 258619.

[33] D. Tuccillo, E. Decencière, S. Velasco-Forero, et al., "Deep learning for studies of galaxy morphology," *Proc. Int. Astron. Union*, vol. 12, no. S325, pp. 191–196, 2016.

[34] D. Tuccillo, E. Decenciére, S. Velasco-Forero, H. Domínguez Sánchez, P. Dimauro, et al., "Deep learning for galaxy surface brightness profile fitting," *Monthly Notices Roy. Astron. Soc.*, pp. 894–909, 2017.

[35] A. Aniyan and K. Thorat, "Classifying radio galaxies with the convolutional neural network," *Astrophysical J. Suppl. Series*, vol. 230, no. 2, 2017, Art. no. 20.

[36] F. Gieseke, S. Bloemen, C. van den Bogaard, T. Heskes, J. Kindler, R. A. Scalzo, V. A. Ribeiro, J. van Roestel, P. J. Groot, F. Yuan, et al., "Convolutional neural networks for transient candidate vetting in large-scale surveys," *Monthly Notices Roy. Astron. Soc.*, vol. 472, no. 3, pp. 3101–3114, 2017.

[37] E. J. Kim and R. J. Brunner, "Star-galaxy classification using deep convolutional neural networks," *Monthly Notices Roy. Astron. Soc.*, vol. 464, no. 4, pp. 4463–4475, 2017.

[38] C. Petrillo, C. Tortora, S. Chatterjee, G. Vernardos, L. Koopmans, G. Verdoes Kleijn, N. Napolitano, G. Covone, P. Schneider, A. Grado, et al., "Finding strong gravitational lenses in the kilo degree survey with convolutional neural networks," *Monthly Notices Roy. Astron. Soc.*, vol. 472, no. 1, pp. 1129–1150, 2017.

[39] F. Lanusse, Q. Ma, N. Li, T. E. Collett, C.-L. Li, S. Ravanbakhsh, R. Mandelbaum, and B. Poczos, "CMU deeplens: Deep learning for automatic image-based galaxy–galaxy strong lens finding," *Monthly Notices Roy. Astron. Soc.*, vol. 473, no. 3, pp. 3895–3906, 2017.

[40] F. Rosenblatt, "Principles of neurodynamics. perceptrons and the theory of brain mechanisms," Cornell Aeronautical Lab Inc, Buffalo NY, USA, Tech. Rep. VG-1196-G-8, 1961.

[41] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," in *Neural Networks: Tricks of the Trade*. New York, NY, USA: Springer, 2012, pp. 437–478.

[42] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ. San Diego La Jolla Inst. Cognitive Sci., La Jolla, CA, USA, Tech. Rep. ICS 8506, 1985.

[43] Y. LeCun, et al., "Generalization and network design strategies," *Connectionism Perspective*, pp. 143–155, vol. 19, 1989.

[44] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural Netw.*, vol. 4, no. 2, pp. 251–257, 1991.

[45] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv:1505.00853*, 2015.

[46] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *Int. J. Uncertainty Fuzziness Knowl.-Based Syst.*, vol. 6, no. 02, pp. 107–116, 1998.

[47] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[48] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[49] S. Jouvel, J.-P. Kneib, O. Ilbert, G. Bernstein, S. Arnouts, T. Dahlen, A. Ealet, B. Milliard, H. Aussel, P. Capak, et al., "Designing future dark energy space missions-i. building realistic galaxy spectro-photometric catalogs and their first applications," *Astronomy Astrophysics*, vol. 504, no. 2, pp. 359–371, 2009.
[50] P. Capak, H. Aussel, M. Ajiki, H. McCracken, B. Mobasher, N. Scoville, P. Shopbell, Y. Taniguchi, D. Thompson, S. Tribiano, et al., "The first release cosmos optical and near-ir data and catalog," *Astrophysical J. Suppl. Series*, vol. 172, no. 1, p. 99, 2007.
[51] O. Ilbert, P. Capak, M. Salvato, H. Aussel, H. McCracken, D. Sanders, N. Scoville, J. Kartaltepe, S. Arnouts, E. Le Floc'h, et al., "Cosmos photometric redshifts with 30-bands for 2-deg2," *Astrophysical J.*, vol. 690, no. 2, 2008, Art. no. 1236.
[52] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/
[53] F. Chollet, et al., "Keras," 2015. [Online]. Available: https://github.com/fchollet/keras
[54] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, no. Jul, pp. 2121–2159, 2011.
[55] R. Bellman, *Dynamic Programming*. Princeton, NJ, USA: Princeton Univ. Press, 1957.
[56] D. Machado, A. Leonard, J.-L. Starck, F. Abdalla, and S. Jouvel, "Darth fader: Using wavelets to obtain accurate redshifts of spectra at very low signal-to-noise," *Astronomy Astrophysics*, vol. 560, 2013, Art. no. A83.
[57] K. Fotiadou, G. Tsagkatakis, B. Moraes, F. B. Abdalla, and P. Tsakalides, "Denoising galaxy spectra with coupled dictionary learning," in *Proc 25th Eur. Signal Process. Conf.*, 2017, pp. 498–502.
[58] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. IT-13, no. 1, pp. 21–27, Jan. 1967.
[59] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[60] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[61] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, "Classification and regression trees," 1984.
[62] R. O. Duda, P. E. Hart, D. G. Stork, et al., *Pattern Classification*, vol. 2. New York, NY, USA: Wiley, 1973.
[63] M. J. Kurtz and D. J. Mink, "Rvsao 2.0: Digital redshifts and radial velocities," *Publications Astron. Soc. Pacific*, vol. 110, no. 750, 1998, Art. no. 934.
[64] C. Stoughton, R. H. Lupton, M. Bernardi, M. R. Blanton, S. Burles, F. J. Castander, A. Connolly, D. J. Eisenstein, J. A. Frieman, G. Hennessy, et al., "Sloan digital sky survey: Early data release," *Astron. J.*, vol. 123, no. 1, 2002, Art. no. 485.
[65] D. Ramachandram and G. W. Taylor, "Deep multimodal learning: A survey on recent advances and trends," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 96–108, Nov. 2017.
[66] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.
[67] M. Levi, C. Bebek, T. Beers, R. Blum, R. Cahn, D. Eisenstein, B. Flaugher, K. Honscheid, R. Kron, O. Lahav, et al., "The desi experiment, a whitepaper for snowmass 2013," *arXiv:1308.0847*, 2013.

Radamanthys Stivaktakis received the BSc degree in computer science from the University of Crete, in March 2017 and the MSc degree from the University of Crete, in March 2019. He is currently working toward the PhD degree in computer science at the University of Crete, under the supervision of Prof. Panagiotis Tsakalides. He is affiliated with the Signal Processing Laboratory, Institute of Computer Science (ICS), Foundation for Research and Technology—Hellas (FORTH) since February 2016. His current research interests mainly include deep learning applications in the fields of remote sensing and astrophysics.

Grigorios Tsagkatakis received the diploma and MS degrees in electronics and computer engineering from the Technical University of Crete, Greece, in 2005 and 2007, respectively, and the PhD degree in imaging science from the Rochester Institute of Technology, New York, in 2011. He is currently working as a research associate with the Signal Processing Laboratory, Institute of Computer Science (ICS), Foundation for Research and Technology Hellas (FORTH). His research is focused on topics related to signal processing and machine learning with application in remote sensing and astrophysics and has been funded by numerous national and European projects. He has co-authored more than 50 peer-reviewed conference papers, journals and book chapters, has been awarded one US patent and three best paper awards. He has served as organizing committee in a number of workshops including Cyber-Physical Systems for Smart Water Networks (CySWater) and the workshop on Computational Intelligence in Remote Sensing and Astrophysics (CIRSA).

Bruno Moraes received the BS degree in physics from Universidade Federal do Rio de Janeiro (UFRJ), in 2006, the "Ingénieur de l'École Polytechnique" diploma from École Polytechnique, Paris, France, in 2006, the MS degree in theoretical physics from joint École Polytechnique/École Normale Supérieure Paris, Paris, France, in 2007, and the PhD degree in theoretical cosmology from Université Montpellier 2, Montpellier, France, in 2010. He is currently an assistant professor with UFRJ. His main research focus is on observational cosmology, with emphasis on statistical and computational methods and their application to astrophysical and cosmological surveys. He is a co-author in more than 30 peer-reviewed refereed technical articles, and a current or past participant in multiple scientific collaborations focused on the above-mentioned themes, such as the CFHT Stripe 82 Survey (CS82), the Dark Energy Survey (DES), and the DEDALE collaboration.

Filipe Abdalla received the MPhys and PhD degrees from Oxford University, in 2002 and 2005, respectively. He is a Royal Society University research fellow and reader in the Cosmology subgroup of the Astrophysics Group of the Department of Physics and Astronomy, University College London. Previously he was a Leverhulme early careers research fellow in the same group. His research encompasses a wide range of topics around cosmology. His central focus is the study of the Universe and its contents during different epochs, which can be observed since the dawn of time. He has been involved with the Dark Energy Survey (DES) and the Euclid satellite mission. He has positions of leadership in both these collaborations, being the Spectroscopic Working group Lead in DES and the OU-LE3 lead in Euclid. He has developed tools which are the state-of-the-art tools in the LOFAR Epoch of Reionisation survey. He is a core member of the LOFAR EoR working group and one of the senior leads of the Cosmology Working group in the SKA.

Jean-Luc Starck received the PhD degree from Nice Observatory and an habilitation from University Paris XI. He is the director of research and head of the CosmoStat laboratory, Institute of Research into the Fundamental Laws of the Universe, Département d'Astrophysique, CEA-Saclay, France. He was a visitor at the European Southern Observatory in 1993, at UCLA in 2004 and at Stanford's Department of Statistics in 2000 and 2005. Since 1994, he is a tenured researcher with CEA. He created in 2010 the CosmoStat laboratory and is strongly involved in the Euclid ESA space mission. He received the EADS prize of the French Academy of Science in 2011, the International Astrostatistics Association (IAA) Fellow Prize in 2016 and the 2018 Gruber Price in Cosmology (as member of the ESA Planck team). He has published more than 200 refereed papers in astrophysics, cosmology, signal processing and applied mathematics, and he is also author of three books.

Panagiotis Tsakalides received the diploma degree in electrical engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1990, and the PhD degree in electrical engineering from the University of Southern California, Los Angeles, California, in 1995. He is a professor in computer science and the vice rector for finance with the University of Crete, Heraklion, Greece, and the Head of the Signal Processing Laboratory, FORTH-ICS. He has an extended experience of transferring research and interacting with the industry. During the last 10 years, he has been the project coordinator in seven European Commission and 12 national research and innovation projects totaling more than 5 M € in actual funding for FORTH-ICS and the University of Crete. His research interests include the field of statistical signal processing with emphasis in non-Gaussian estimation and detection theory, sparse representations, and applications in sensor networks, audio, imaging, and multimedia systems. He has coauthored more than 180 technical publications in these areas.