Eur. Phys. J. C (2023) 83:304
https://doi.org/10.1140/epjc/s10052-023-11435-9

THE EUROPEAN
PHYSICAL JOURNAL C

Regular Article - Theoretical Physics

Neural network reconstructions for the Hubble parameter, growth rate and distance modulus

Isidro Gómez-Vargas[1,2,a], **Ricardo Medel-Esquivel**[1,2,b], **Ricardo García-Salcedo**[2,3,c], **J. Alberto Vázquez**[1,d]

[1] ICF, Universidad Nacional Autónoma de México, 62210 Cuernavaca, Morelos, Mexico
[2] CICATA-Legaria, Instituto Politécnico Nacional, 11500 Mexico City, Mexico
[3] Departamento Ingeniería Civil, División de Ingeniería, Universidad de Guanajuato, C.P. 36000, Gto., México

Received: 11 May 2022 / Accepted: 26 March 2023 / Published online: 19 April 2023

Abstract This paper introduces a new approach to reconstruct cosmological functions using artificial neural networks based on observational measurements with minimal theoretical and statistical assumptions. By using neural networks, we can generate computational models of observational datasets, and then we compare them with the original ones to verify the consistency of our method. This methodology is applicable to even small-size datasets. In particular, we test the proposed method with data coming from cosmic chronometers, $f\sigma_8$ measurements, and the distance modulus of the Type Ia supernovae. Furthermore, we introduce a first approach to generate synthetic covariance matrices through a variational autoencoder, using the systematic covariance matrix of the Type Ia supernova compilation.

1 Introduction

One of the biggest challenges for the cosmological community is the explanation of the current accelerated Universe expansion. A theoretical conception, commonly called Dark Energy (DE), is introduced to explain this mysterious phenomenon and whose nature is still unraveled [1–3]. The standard model of cosmology, or simply ΛCDM, is the homogeneous and isotropic cosmological model whose material content is as follows: ordinary matter, the simplest form of Dark Energy known as cosmological constant Λ and, finally, a key component for the formation of structures in the Universe called Cold Dark Matter (CDM). It has had significant achievements, such as being in excellent agreement with most of the currently available data, for example, measurements coming from the Cosmic Microwave Background radiation [4], Supernovae Ia (SNeIa) [5], Cosmic Chronometers (CC) [6] and Baryon Acoustic Oscillations (BAO) [7]. Nevertheless, the ΛCDM model has its drawbacks: on theoretical grounds, the cosmological constant faces several problems, i.e., fine-tuning and cosmic coincidence [8,9], and from an observational point of view, it also suffers to the so-called Hubble tension, a measurement disagreement of the Hubble parameter H_0 among different datasets [10]. The presence of these issues opens the possibility to extensions beyond the standard cosmological model by considering, for instance, a dynamical DE, modifications to the general theory of relativity [11] or other approaches.

The search for possible signatures for cosmological models beyond the ΛCDM has led to the creation of an impressive set of high accuracy surveys, already underway or being planned [12–14], to gather a considerable amount of information that constrains the properties of the universe. A viable cosmological model that leads to the current accelerating universal expansion is demanded to comply with all the relevant observational data. Extensions to the cosmological constant that allow a redshift-dependent equation-of-state (EoS) $w(z)$ include extra dimensions [15], modified gravity [16], scalar fields [17], scalar-tensor theories with non-minimal derivative coupling to the Einstein tensor [18], graduated dark energy [19], just to mention a few. However, in the absence of a fundamental and definitive theory of dark energy, a time-dependent behavior can also be investigated by choosing an EoS mathematically appealing or a parameterized form in a simple way; examples of these forms in terms of redshift include a Taylor expansion [20], polynomial [21], logarithmic [22], oscillatory [23,24], a combined form of them [25] or in terms of cosmic time [26]. Nonetheless, the a priori assumption of a specific theoreti-

[a] e-mail: igomez@icf.unam.mx
[b] e-mail: rmedel@ipn.mx
[c] e-mail: rigarcias@ipn.mx (corresponding author)
[d] e-mail: javazquez@icf.unam.mx

cal model may lead to misleading model-dependent results regardless of the dark energy properties. Hence, instead of committing to a particular cosmological model, the non-parametric inference techniques allow extract information directly from the data to detect features within cosmological functions, for instance, $w(z)$. The main aim of a non-parametric approach is to infer (reconstruct) an unknown quantity based mainly on the data and make as few assumptions as possible [8,27]. Several non-parametric techniques are used to perform model-independent reconstructions for cosmological functions directly from the data, such as histogram density estimators [28], Principal Component Analysis (PCA) [29], smoothed step functions [30], gaussian processes [31–34], Simulation Extrapolation method (SIMEX) [35] and Bayesian nodal free-form methods [36,37].

After the reconstruction is performed, the function can be considered as a new model to look for possible deviations from the standard ΛCDM. In other words, the result of a model-independent reconstruction may be used to analyze its similarity with different theoretical models and, therefore, to select its best description for the data. There are several examples of model-independent reconstructions of cosmological functions, some of them focus on dark energy features [28,30,38,39], on the cosmic expansion [35], deceleration parameter [34], growth rate of structure formation [33], luminosity distance [40,41] and primordial power spectrum [42,43], among many others.

On the other hand, the recent increase in computing power and the vast amount of coming data have allowed the incursion of machine learning methods as analysis tools in observational cosmology [44–49]. In this work, we focus on the computational models called Artificial Neural Networks (ANNs). They have been used in a variety of applications, such as image analysis [50,51], N-body simulations [52,53] and statistical methods [54–58].

In a similar way that model-independent reconstructions are used to recover the baseline function, the main goal of this paper is to propose a new method based on artificial neural networks using solely the current datasets with the most minimal theoretical assumptions. Here, we refer to the neural networks output as model-independent reconstructions because they do not incorporate any a priori cosmological assumption to generate the model from the datasets. This work is similar to previous research in which neural networks produce reconstructions of cosmological functions [59–61]. However, the novel differences here are the exploration of more cosmological datasets, the null consideration of a fiducial cosmology in the reconstructions, the exclusive use of the observational data to train the neural networks (even if they are small), and the new treatment to the non-diagonal error covariance matrices.

A benefit of using well-trained neural networks is that these do not consider a fiducial cosmology; the data generated can be assumed as new observations of the exact nature of the original dataset. Another advantage of neural networks over other standard interpolation techniques is that, due to their nonlinear modeling capabilities, these do not require consideration of any statistical distribution for the data. In addition, neural networks also allow us to generate computational models for the errors of the observational datasets; when the errors have no correlations (diagonal covariance matrices), we develop a single neural network model that considers measurements and errors from the original dataset. However, we must generate a different neural model when the error matrices are non-diagonal. We show that our methodology can apply to several astronomical datasets, including full covariance matrices with correlations among measurements, for which we introduce a special treatment with variational autoencoders.

The rest of the paper has the following structure. In Sect. 2, we briefly introduce the cosmological and statistical concepts used throughout this work: cosmological models, functions and observations in Sect. 2.1; a short summary of Bayesian inference in Sect. 2.2 and an overview of neural networks in section 2.3. Section 3 describes the methodology used during the neural network training to generate computational models based on cosmological data. Section 4 contains our results, in Sect. 4.1 we show the generation of model-independent reconstructions using neural networks from observational measurements of the Hubble distance $H(z)$, a combination of the growth rate of cosmological perturbations times the matter power spectrum normalization $f\sigma_8(z)$ and the distance modulus $\mu(z)$ along with its covariance matrix. In Sect. 4.2, we use Bayesian inference on two cosmological models to check the consistency of our reconstructions in comparison with the original data and the expected values of the cosmological parameters. Finally, in Sect. 5 we expose our final comments. Furthermore, within the appendices, a brief description of feedforward neural networks and variational autoencoders is included, as well as the training process used for the networks and our experimental method to learn the covariance matrix.

2 Cosmological and statistical background

This section introduces the cosmological models, functions, and datasets used throughout this work. The datasets are used to develop the model-independent reconstructions with our method and the cosmological models are used to compare these reconstructions with the theoretical predictions. We also provide a brief overview of the relevant concepts of Bayesian inference, which we use as a consistency test for the results of our neural network reconstructions, and of the essential elements of Artificial Neural Networks, which are the core of our proposed method. Throughout this paper we use the geometric unit system where $\hbar = c = 8\pi G = 1$.

2.1 Cosmological models and datasets

Models

The Friedmann equation describing the late-time dynamical evolution for a flat-ΛCDM model can be written as:

$$H(z)^2 = H_0^2 \left[\Omega_{m,0}(1+z)^3 + (1-\Omega_{m,0}) \right], \qquad (1)$$

where H is the Hubble parameter and Ω_m is the matter density parameter, subscript 0 attached to any quantity denotes its present ($z = 0$) value. In this case, the DE EoS is $w(z) = -1$.

A step further to the standard model is to consider the dark energy being dynamic, where the evolution of its EoS is usually parameterized. A commonly used form of $w(z)$ is to take into account the next contribution of a Taylor expansion in terms of the scale factor $w(a) = w_0 + (1-a)w_a$ or in terms of redshift $w(z) = w_0 + \frac{z}{1+z} w_a$; we refer to this model as CPL [20,62]. The parameters w_0 and w_a are real numbers such that at the present epoch $w|_{z=0} = w_0$ and $dw/dz|_{z=0} = -w_a$; we recover ΛCDM when $w_0 = -1$ and $w_a = 0$. Hence the Friedmann equation for the CPL parameterization turns out to be:

$$H(z)^2 = H_0^2 \Big[\Omega_{m,0}(1+z)^3 + (1-\Omega_{m,0})(1+z)^{3(1+w_0+w_a)} e^{-\frac{3w_a z}{1+z}} \Big]. \qquad (2)$$

Datasets

Cosmic chronometers (CC) are galaxies that evolve slowly and allow direct measurements of the Hubble parameter $H(z)$. These measurements have been collected over several years [6,63–69], and now 31 data points are available within redshifts between 0.09 and 1.965, along with their independent statistical errors.

The **growth rate measurement** is usually referred to the product of $f\sigma_8(a)$ where $f(a) \equiv d\ln\delta(a)/d\ln a$ is the growth rate of cosmological perturbations given by the density contrast $\delta(a) \equiv \delta\rho/\rho$, being ρ the energy density and σ_8 the normalization of the power spectrum on scales within spheres of $8h^{-1}$Mpc [70]. Therefore, the observable quantity $f\sigma_8(a)$, or equivalently $f\sigma_8(z)$, is obtained by solving numerically:

$$f\sigma_8(a) = a\frac{\delta'(a)}{\delta(1)}\sigma_{8,0}. \qquad (3)$$

The $f\sigma_8$ data are obtained through the peculiar velocities from Redshift Space Distortions (RSD) measurements [71] observed in redshift survey galaxies or by weak lensing [72], where the density perturbations of the galaxies are proportional to the perturbations of matter. We used an extended version of the Gold-2017 compilation available in [73], which includes 22 independent measurements of $f\sigma_8(z)$ with their statistical errors obtained from redshift space distortion measurements across various surveys (see references therein); the authors explain that the data used from the $f\sigma_8$ combination has been shown to be unbiased.

Table I of [73] contains the $f\sigma_8(z)$ measurements along with their standard deviations used in this work to form our training dataset. In the same Table, it is indicated the reference matter density parameter $\Omega_{m,0}$ for each measurement and other details of the dataset.

Supernovae (SNeIa). The SNeIa dataset used in this work corresponds to the Joint Lightcurve Analysis (JLA), a compilation of 740 Type Ia supernovae. It is available in a binned version that consists of 31 data points with a covariance matrix $C_{JLA} \in \mathbb{R}^{31\times31}$ related to the systematic measurement errors [5]. As a proof of the concept, we focused on the binned version because, even though the treatment of a matrix in $\mathbb{R}^{740\times740}$ from the entire dataset is a straightforward process, it is very computationally expensive (see Appendix 1 for details). However, it can be implemented on more powerful computers.

Let us assume a spatially flat universe, for which the relationship between the luminosity distance d_L and the comoving distance $D(z)$ is given by:

$$d_L(z) = \frac{1}{H_0}(1+z)D(z), \qquad \text{with} \qquad D(z) = H_0 \int \frac{dz}{H(z)}. \qquad (4)$$

Using d_L defined in Eq. (4), and considering that the distance is expressed in Mega parsecs, the distance modulus is defined as follows:

$$\mu(z) = m - M = 5\log_{10} d_L(z) + 25, \qquad (5)$$

where m is the apparent magnitude and M refers to the absolute magnitude. According to Ref. [5], in order to use the JLA binned data, and to perform the Bayesian parameter estimation, we need to apply the following likelihood:

$$\log \mathcal{L} = r^T \cdot C_{\text{JLA}}^{-1} \cdot r, \qquad (6)$$

where $r = \mu_b - M - 5\log_{10} d_L(z)$ and μ_b is the distance modulus obtained from the binned JLA dataset.

We can use the definition for the theoretical distance modulus from Eq. (5) and obtain $r = \mu_b - \mu(z) + (25 - M)$, and for simplicity, we fixed M because the prior knowledge suggests a constant value [2]. However, reference [5] warns about the importance of treating the absolute magnitude M as a free parameter in the Bayesian inference when using the binned dataset to avoid any potential issues with the estimated value of the Hubble parameter. Nevertheless, as our main aim is to use the Bayesian inference as a proof of the concept for our methodology and not to draw a cosmological conclusion from the results, we have fixed M to the same value in all the tests for the sake of simplifying computations with the data and their covariance matrix.

Details about the calibration of the Type IA supernovae binned dataset, and its covariance matrix used in this work are contained in appendices E and F of the Ref. [5].

2.2 Bayesian inference

Given a set of observational data and a mathematical expression for a cosmological model, a conditional probability function can be constructed regarding the model parameters and the observables. There are many ways to infer the combination of parameters that best fit the data. In cosmology, Bayesian inference algorithms have been used prominently [74–76]; however, methods such as the Laplace approximation [77], genetic algorithms [46,78], simulated annealing [79] or particle swarm optimization [80] have also been explored.

Bayesian statistics is a paradigm in which probabilities are computed given the prior knowledge of the data [81,82]. It can perform two essential tasks in data analysis: parameter estimation and model comparison. The Bayes' Theorem on which it is based is as follows:

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)}, \tag{7}$$

where D represents the observational dataset and θ is the set of free parameters in the theoretical model. $P(\theta)$ is the prior probability density function and represents the previous knowledge of the parameters. $\mathcal{L} = P(D|\theta)$ is the likelihood function and indicates the conditional probability of the data D given the parameters θ of a model. Finally, $P(D)$ is a normalization constant, that is, the likelihood marginalization, and is called the Bayesian evidence. This quantity is very useful in model comparison, for example, it has been used in several papers to compare dark energy models through the Bayes factor and Jeffrey's scale [17,36].

Considering the datasets described above, we use the following log-likelihoods:

$$\log \mathcal{L}_i = -\frac{1}{2}(D^i_{\text{th}} - D^i_{\text{obs}})^T \cdot C_i^{-1} \cdot (D^i_{\text{th}} - D^i_{\text{obs}}), \tag{8}$$

where $i = 1, 2, 3$ correspond to the three datasets: cosmic chronometers $[D^{i=1} = H(z)]$ and growth rate measurements $[D^{i=2} = f\sigma_8(z)]$. D_{obs} represents the observational measurements, while D_{th} is the theoretical value for the cosmological models. $C_{i=1}$ and $C_{i=2}$ are diagonal covariance matrices. The log-likelihood for the SNeIa has been previously defined (see Eq. 6).

2.3 Artificial neural networks

Artificial Neural Networks (ANNs) are computational models that learn the intrinsic patterns of a dataset. A neural network consists of several sets of neurons or nodes grouped into layers, and between the nodes of different layers some connections are associated with numbers called weights. The training of a neural network aims to find the best values for all the weights to produce a generalization of the data, and this is done through the minimization of an error function (called loss function) that measures the difference between the values predicted by the neural network and the actual values of the dataset (see Appendix A for more details and in Ref. [83] for an introduction to the subject).

The *Universal Approximation Theorem* [84] states that an Artificial Neural Network with at least one hidden layer with a finite number of neurons can approach any continuous function if the activation function is continuous and nonlinear. Therefore an ANN is capable of learning the intrinsic functions inside cosmological datasets and generating a model based only on the data. Two types of artificial neural networks are implemented in this work: Feedforward Neural Networks (FFNN) and AutoEncoders (AE). The FFNN, also called multilayer perceptrons or deep feedforward networks, are the quintessential deep learning models [85]. In this type of ANN, the connections and information flow are feedforward, i.e., from the first to the last layer without loops. These consist of one input layer, at least one hidden layer, and an output layer. The input consists of the dataset's independent variables (or features), while the output contains the dependent variables (or labels).

On the other hand, the autoencoders [86] are trained to generate a copy of its input on its output. We use this type of network to learn the errors of a dataset when they conform to a non-diagonal covariance matrix. We use the Variational Autoencoders (VAE). Details about autoencoders are in the Appendix B.

3 Methodology

The datasets in this work contain redshifts, an observable for each redshift and the corresponding statistical errors. Our goal is to generate neural network models for the data despite the complex dependency of these three variables. That is, we take advantage of the ability of neural networks to model the relationship between these variables. Neural networks, with a structure based on multiple neurons and nonlinear activation functions, allow us to generate computational models utterly independent of any existing cosmological model or statistical assumptions.

Even though neural networks are standard in the treatment of large datasets, there is no mathematical constraint in using

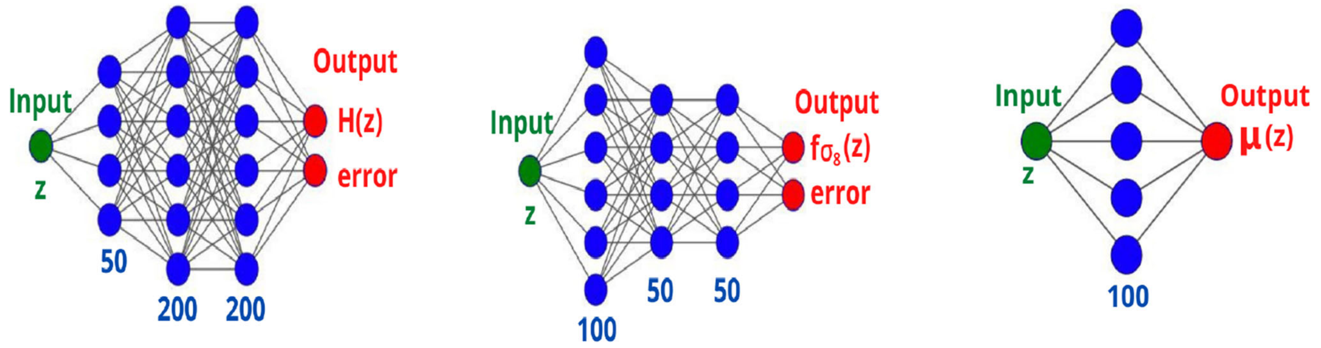

Fig. 1 Neural network architectures chosen to model the data from cosmic chronometers (CC), $f\sigma_8$ measurements and the JLA SNeIa compilation, respectively; the batch size found for each case was: 16, 1 and 1. In the last architecture, there is only one node in the output layer because the errors are computed with a variational autoencoder (described in the Appendix D) given the original non-diagonal covariance matrix of the systematic errors. Blue numbers indicate the nodes in each layer. It is worth mentioning that, in all diagrams, both the redshift functions and the errors in the output layers refer to observational measurements, and no functional form of any cosmology is being considered

them for any size of a given dataset, and it is probed in [87]; in particular, it is demonstrated that neural models can be used with a total number of weights larger than the number of sample data points. New approaches in neural network research that focus on small datasets are the references [88,89], and a machine learning field, so-called *few shot learning* [90], uses only a few of samples to train the network. It is worth mentioning that using small datasets, although the computing resources are not demanding, it could become challenging to find the hyperparameters that generate an acceptable model. By monitoring the behavior of the loss function both in the training and the validation sets, we can check the equilibrium between the bias and variance to have certainty about the excellent calibration of the neural network.

In all our datasets, we use their lowest and highest redshifts as the limits of the training set, and then we select a random 20% as the validation set. We do not use a test set due to the small size of the dataset. However, we test several combinations of parameters to select those that generate an excellent neural network model.

For the analysis of cosmic chronometers and $f\sigma_8$ measurements, we work with the FFNNs because their diagonal covariance matrices can be arranged into a single column of the same length as the number of observational measures. For these networks, we use the mean squared error (MSE) as a loss function which is a usual selection in regression problems:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i}^{n}(Y_i - \hat{Y}_i)^2, \tag{9}$$

where Y_i is a vector with predictions of the ANN, $\hat{Y}_i$ a vector with the expected values, and n is the number of predictions (or the length of Y_i and $\hat{Y}_i$).

In the case of SNeIa data, we use a FFNN to learn the distance modulus and a Variational Autoencoder for the non-diagonal covariance matrix of the systematic errors.

In addition, we implemented the Monte Carlo Dropout (MC-DO) method [91] in all our FFNNs. This method allows the output of a neural network to have an uncertainty associated with it and to generate robust models due to the dropout being a regularization technique. In the last part of Appendix A, we describe the basic definitions of Dropout and MC-DO.

We found the best architectures (shown in Fig. 1) among several combinations of the intrinsic parameters (hyperparameters) of the neural networks. Appendix C describes our careful selection of the hyperparameters of the feedforward neural networks, such as epochs, number of nodes, and how we have applied the MC-DO method. On the other hand, Appendix D explains how we configure the VAE neural network, its loss function and other details about its training, with the non-diagonal covariance matrix of the binned JLA compilation.

Once the neural networks are well trained, they constitute a model-independent reconstruction, for which we can compare with observations and theoretical predictions. As a consistency test of our neural reconstructions, we perform Bayesian inference for ΛCDM and CPL models, and the expected posterior probabilities would be very similar between the reconstruction and the original datasets; otherwise, another neural network architecture must be chosen. We use the following flat priors: for the matter density parameter today $\Omega_m \in [0.05,\ 0.5]$, for the physical baryon density parameter $\Omega_b h^2 \in [0.02,\ 0.025]$, for the reduced Hubble constant $h \in [0.4,\ 0.9]$, and for the amplitude of the (linear) power spectrum $\sigma_8 \in [0.6,\ 1.0]$. When assuming the CPL parameterisation, we use $w_0 \in [-2.0,\ 0.0]$ and $w_a \in [-2.0,\ 2.0]$. The h parameter refers to the dimensionless reduced Hubble parameter today $H/100\ \mathrm{kms}^{-1}\mathrm{Mpc}^{-1}$.

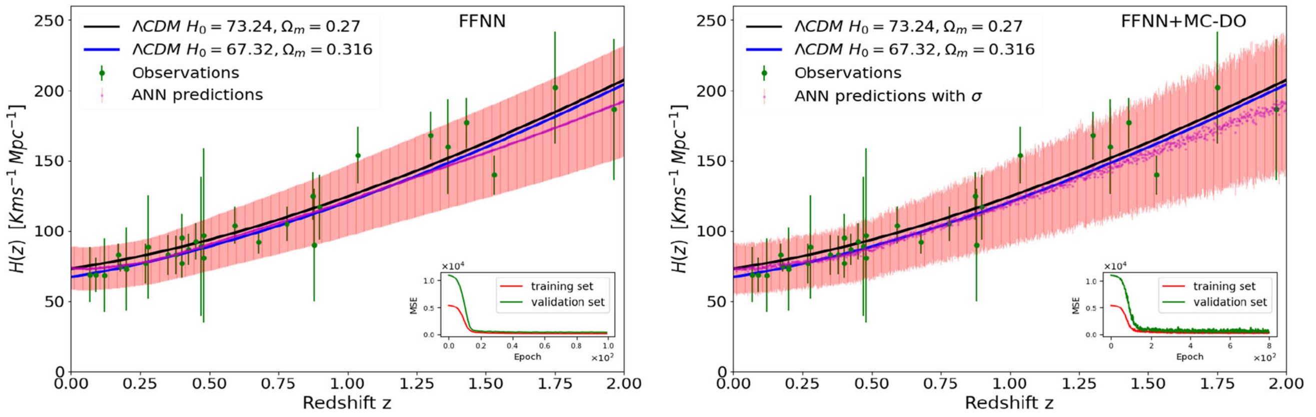

Fig. 2 Hubble distance reconstruction with FFNNs. Left: Purple points represent the FFNN predictions for $H(z)$ along with their error bars in red color. Right: Similarly to FFNN but adding MC-DO, we executed 100 times the Monte Carlo dropout to compute the uncertainties of the predictions. Therefore the purple points are the average predictions of the MC-DO executions, and the red error bars are the uncertainties of the FFNN plus the error predictions (see Eq. 16). In both cases, we compare the neural reconstructions with the original cosmic chronometers (green bars) and $H(z)$ from ΛCDM, as shown in the labels. The small panels show the individual behavior of the loss function (MSE) in the training (red) and validation (green) sets, and these plots suggest that it is an excellent neural network model with no overfitting or underfitting

4 Results

From the observational datasets, we have trained the neural networks to reconstruct the Hubble parameter $H(z)$, the growth rate $f\sigma_8(z)$ and the distance modulus $\mu(z)$, their predictions conform the corresponding model-independent reconstructions. Finally, we have performed the parameter estimation to test the consistency of the reconstructions.

4.1 Reconstructions

$H(z)$ data

To visualize the $H(z)$ reconstructions performed by the FFNN using the CC, we generate predictions of $H(z)$ and their corresponding errors given 1000 different redshifts. In Fig. 2 we show the FFNN alone (left) and the FFNN using MC-DO (right), where the original data points with their statistical errors are green, while in magenta the neural network reconstruction along with their predicted errors. Also in this figure, we compare the outputs of the neural network models with the theoretical predictions of ΛCDM using the two values that yield the Hubble tension $H_0 = 73.24$ km s^{-1} Mpc^{-1} and $\Omega_m = 0.27$ coming from the Cepheid variables [92] and, on the other hand, $H_0 = 67.40$ km s^{-1} Mpc^{-1} and $\Omega_m = 0.316$ measured by the Planck mission [4].

We can notice that the FFNN alone and with MC-DO generate reconstructions in agreement with the theoretical predictions, exclusively based on the observable measurements and their statistical errors. The observational data points are only a few, therefore the scatter of the measurements is underestimated; however, based on their curves for the loss function, we can confirm that the neural networks generate good models. The dispersion in the FFNN+MC-DO reconstruction is higher because it performs statistics on several predictions and includes the uncertainty of the method itself, therefore its results are more robust and reliable than FFNN alone.

It is worth mentioning that the reconstructions performed by our method with FFNNs are consistent with the $H(z)$ reconstructions of other works performed with Gaussian processes [47,93–95] and with neural networks [60], where the training dataset consists of $H(z)$ evaluations from a flat ΛCDM cosmology, redshifts are distributed under a gamma distribution, and errors are produced by an analytical expression [96]. In this sense, the advantages of our results are that they have no statistical assumptions on the data as Gaussian processes usually do, we do not use either the Friedmann equation or another cosmological equation to augment the datasets, and the neural networks learned directly to model observational errors without imposing some analytical expression beforehand.

$f\sigma_8(z)$ data

We trained the FFNNs with the extended Gold-2017 compilation of growth rate measurements and their statistical uncertainties, we generate 1000 predictions from the trained neural nets to visualize the $f\sigma_8(z)$ reconstructions. In Fig. 3, we plot the original data with their uncertainties (green), while the neural network predictions and their errors are displayed in red (left panel is the FFNN alone and right panel corresponds to FFNN+MC-DO). We also draw some curves of $f\sigma_8(z)$ from the analytical evaluation of the CPL model for different values of w_0 and w_a. We notice that the models are

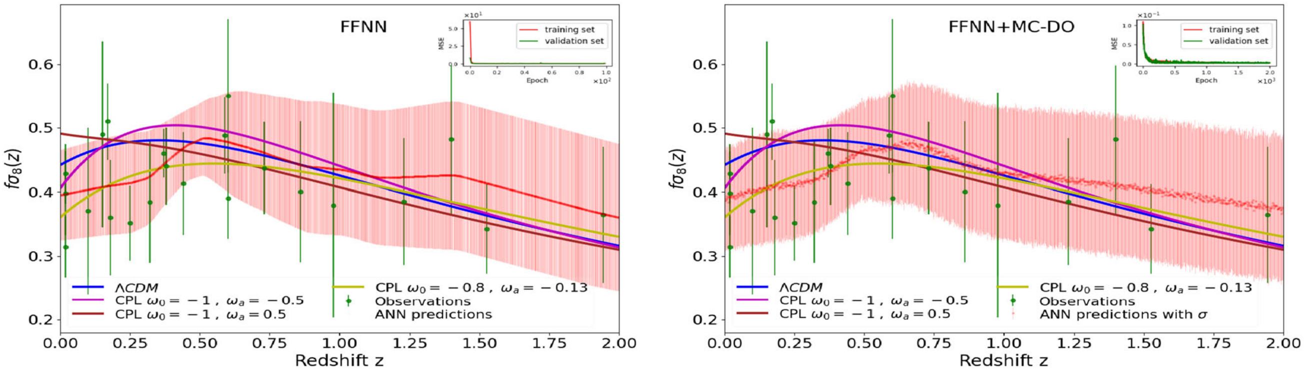

Fig. 3 Neural reconstructions for $f\sigma_8(z)$ (red lines) and their respective errors. The original observations are in green with their uncertainties. Left: The $f\sigma_8$ reconstruction performed by FFNN alone. Right: $f\sigma_8$ reconstruction predicted by the FFNN using Monte Carlo dropout, the averages of 100 executions of MC-DO are indicated with the red data and their standard deviations are added to the error predictions. In both cases, the small panels display the behavior of the loss function (MSE) in the training (red curve) and validation (green curve) sets; in this case, these curves also show a good neural network model

within the reconstructions in both cases. Hence, this dataset by itself may provide loose constraints on the CPL parameters, mainly because there are very few points and relatively large statistical errors. However, the values $w_0 = -0.8$ and $w_a = -0.4$ (yellow line) seem to have a better agreement with the reconstruction.

We can analyze Fig. 3 to compare the two results. We can deduce that it is better to use the MC-DO than just the FFNN alone because MC-DO provides a dropout as a regularization technique, avoiding overfitting and producing a more general data model. The small dataset makes for the FFNN alone difficult to learn at redshifts close to zero; however, FFNN+MC-DO performs better in that respect. Regarding the MC-DO improvement, it can be noticed that in the case of the FFNN method, several data points are outside the reconstruction, while in the reconstruction generated by FFNN+MC-DO only the $f\sigma_8(z = 0.17) = 0.51$ point is excluded. Despite the significant errors and its sparsity, the FFNNs could generate a model consistent with the underlying cosmological theory of the ΛCDM and CPL models. Moreover, the reconstructions produced by the FFNNs have a similar trend to other model-independent reconstructions of $f\sigma_8(z)$ made by Gaussian processes [33,97] with the advantage of letting aside any statistical assumption of the data distribution.

Distance modulus $\mu(z)$ data

Our reconstruction methodology for the distance modulus differs from those previously presented; in this case, the main aim is modeling the errors of the observational measurements when they are correlated, that is, when the covariance matrix is non-diagonal. For this purpose, we introduce a new method based on a variational autoencoder (VAE) along with an FFNN to perform the whole neural network modeling for this dataset.

With the distance modulus reconstruction, performed by the FFNNs, we have generated synthetic data points from 31 log-uniformly distributed redshift values $z \in [0.01, 1.3]$ plus a small Gaussian noise for both the FFNN alone and the FFNN+MC-DO. For comparison, in the Fig. 4 are the percentage differences between the ΛCDM predictions with the original observations from the binned JLA compilation (in green), and with the neural networks reconstructions (in red).

We can generate several points at any different values of redshift from the neural network models trained with the distance modulus and model the errors with a VAE neural network (see Appendix D for details of the developed method). Our motivation for using autoencoders for the covariance matrix is that an autocoder is trained to generate an output of the same nature as the input while encoding a compressed representation in the part between the encoder and the decoder. In addition, if we use a VAE, during training this compressed representation is also sampled through variational inference and, at the end of training, we can know the probability distribution that characterizes it and perform interpolations, sweeping the latent space, to generate new covariance matrices. Furthermore, we can force the dimension of this compressed representation (latent space) to be one-dimensional, for easier interpretation or to map to another 1D distribution.

A limitation of our method is that the new points, and errors, should correspond to the dimensionality of the matrix, in our case 31. Figure 5 shows the absolute error for the outputs of the VAE trained with the non-diagonal covariance matrix of the JLA systematic errors; it can be seen that in both cases (VAE+FFNNN and VAE+FFNNN+MC-DO) the

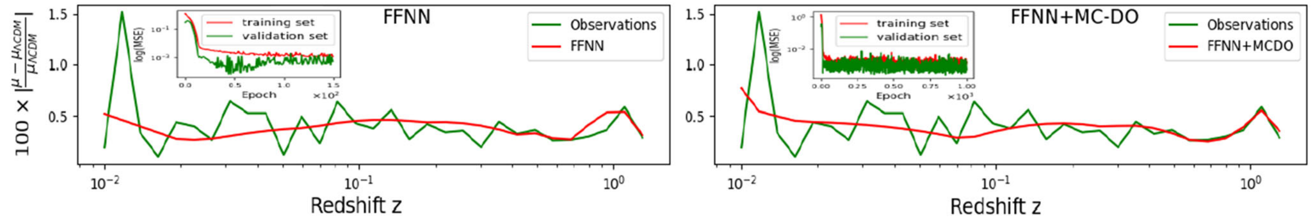

Fig. 4 Comparison between the percentage error times 100 between the ΛCDM theoretical predictions for the distance modulus with the observational measurements and the neural network reconstructions. In the small panels of both figures, the behavior of the loss function, in logarithmic scale, is shown for both the validation (green curve) and training (red curve) sets along with the number of epochs for each case (300 and 1800); it can be seen that after the training process, we obtain acceptable models for the binned JLA dataset. Left: With FFNN alone. Right: FFNN with Monte Carlo Dropout

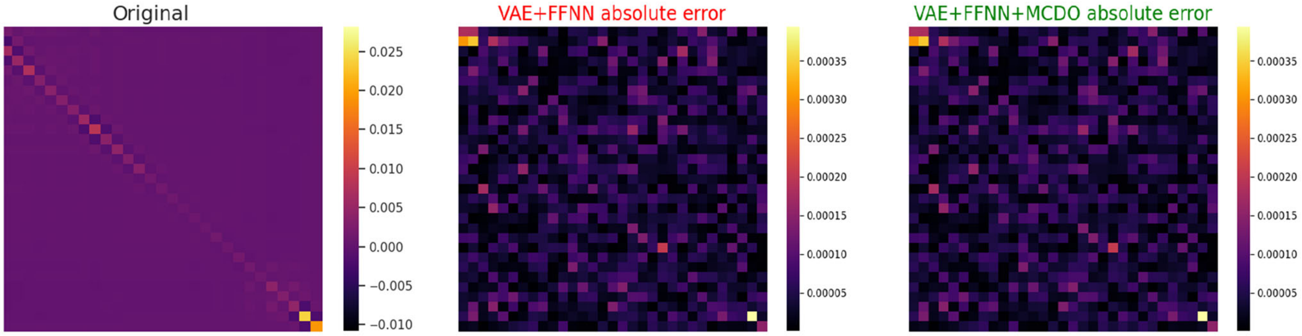

Fig. 5 Left: Original covariance matrix with systematic errors from JLA compilation (binned version) with 961 entries. Middle: Absolute error for the covariance matrices predicted by the VAE+FFNN concerning the original ones. Right: Absolute error for the covariance matrices predicted by the VAE+FFNN with MC-DO

differences are two or more orders of magnitude lower than the original matrix; therefore the new matrices are in agreement with the original one. Nonetheless, in Sect. 4.2, we test these covariance matrices predicted by the neural networks in a Bayesian inference framework to verify whether they are statistically consistent with the original data.

From Fig. 4, it can be seen that the reconstructions are in better agreement with the ΛCDM model than to the original data points; this may occur because when the neural network generates a model for all data points, it underestimates some of the observational variances and focuses more on the similarity of all observations. The FFNN alone has a smaller error in the first prediction, but the FFNN+MC-DO reconstructs the last redshifts better; however, based on the behavior of the loss function, we can say that the computational models generated by the neural networks for the binned JLA compilation are acceptable, both in the case of the FFNN alone, and with MC-DO.

4.2 Testing reconstructions with Bayesian inference

We use a Bayesian inference process for testing the consistency of the reconstructions obtained with the neural networks. In addition to the three original datasets (cosmic chronometers, $f\sigma_8$ measurements, and binned JLA compilation), we have created two datasets for each type of observation from the trained FFNNs with and without MC-DO. As proof of the concept, the new datasets for CC and $f\sigma_8$ consist of 50 random uniformly distributed points in redshift. At the same time, for SNeIa, they were 31 log-uniformly distributed in redshift (same size as the original dataset). We also generated its respective covariance matrix for the SNeIa case with the decoder part of the trained VAE. We performed the Bayesian inference with the data from the neural networks reconstructions and with the original data to evaluate the quality of the reconstructions. For this purpose, we analyze the ΛCDM and CPL models. The idea is that if the neural network reconstructions are satisfactory, the Bayesian estimation of the parameters for the theoretical models should be very similar from those obtained with the original observations, i.e., they should have similar means and standard deviations in the posterior distributions. If this condition is not satisfied, it is necessary to retrain the neural networks or use another hyperparameter configuration.

We have used the data from CC, $f\sigma_8$ measurements, and JLA separately. The most representative results are in Fig. 6, along with Table 1, which contains mean values and standard deviations, and they have been sorted according to the

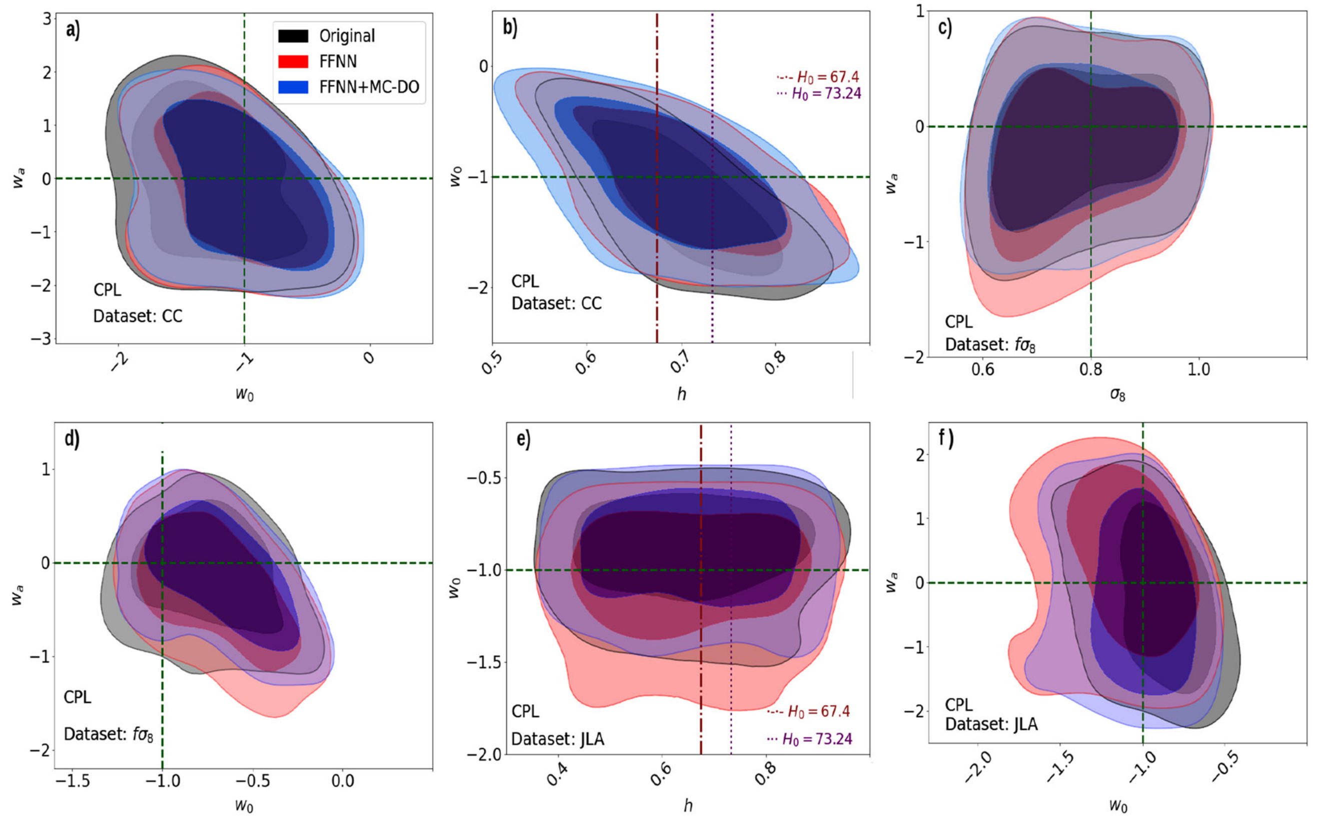

Fig. 6 2D marginalized posterior distributions from different combinations of datasets: original data, reconstructions data points from FFNN and FFNN+MC-DO. The green dashed lines ($w_0 = -1, w_a = 0$) and ($\Omega_1 = 0, \Omega_2 = 0$) correspond to the ΛCDM model. The constraints are plotted with 1σ and 2σ confidence contours

datasets used as a source (original, FFNN, and FFNN+MC-DO), and to the two models involved (ΛCDM and CPL). Results are displayed for the reduced Hubble parameter h, σ_8, w_0 and w_a parameters for the CPL model. In addition, the last column of the Table 1 contains the $-2 \ln \mathcal{L}_{\max}$ of the Bayesian inference process for each case. One thing to note is that the neural networks make models that could be thought of as a function $g : z \in \mathbb{R} \rightarrow v \in \mathbb{R}^2$, $v = (f(z), err(f(z)))$, where both $f(z)$ and the error of the observational measurements are being modeled, so when neural network predictions are used to make Bayesian inference, the errors are of the same order of magnitude as the original ones. Before analyzing each scenario separately, it is worth mentioning some generalities in the results. First, it can be noted that when using a single source separately, the constraints are consistent. They all have a similar best fit (maximum likelihood), and secondly, the results agree with the ΛCDM model.

In the case of parameter estimation, displayed in Table 1, and posterior distributions shown in Fig. 6, we notice that the best-fit values are mutually contained within their 1σ standard deviations, in agreement with the ΛCDM and CPL values. Therefore, the neural network models generated by cosmic chronometers, $f\sigma_8(z)$ measurements and distance modulus, through the Bayesian parameter estimation, are statistically consistent with each other.

5 Conclusions

Throughout this work, we generated neural network models for cosmological datasets between redshifts $z = 0$ and $z = 2$ (cosmic chronometers, $f\sigma_8$ measurements, and SNeIa). We used the neural models to generate model-independent reconstructions of $H(z)$, $f\sigma_8(z)$ and $\mu(z)$. Then, we applied Bayesian inference to data points from the reconstructions to verify that they can reproduce the expected values of the cosmological parameters in ΛCDM and CPL models.

We have shown that well-calibrated artificial neural networks can produce computational models for cosmological data, even when the original datasets are small. The neural network models generate model-independent reconstructions of the Hubble distance $H(z)$, $f\sigma_8(z)$ and distance modulus $\mu(z)$ exclusively from observational data and without assuming any cosmological model. Our results are consistent

Table 1 Parameter estimation using Bayesian inference with datasets from different sources: original, FFNN alone, and FFNN using Monte Carlo dropout

		Datasets	CC			
Source	Model	h	w_0	w_a		$-2\ln\mathcal{L}_{\max}$
Original	ΛCDM	0.678 ± 0.039	–	–		14.502
	CPL	0.703 ± 0.064	-1.223 ± 0.447	-0.061 ± 1.075		14.290
FFNN	ΛCDM	0.698 ± 0.057	–	–		0.176
	CPL	0.703 ± 0.071	-1.072 ± 0.431	-0.179 ± 1.025		0.042
FFNN+MC-DO	ΛCDM	0.699 ± 0.063	–	–		0.346
	CPL	0.689 ± 0.078	-1.014 ± 0.450	-0.227 ± 1.003		0.284
		Datasets	$f\sigma_8$			
		h	w_0	w_a	σ_8	
Original	ΛCDM	0.648 ± 0.147	–	–	0.787 ± 0.115	11.932
	CPL	0.638 ± 0.135	-0.742 ± 0.264	-0.144 ± 0.468	0.777 ± 0.111	11.908
FFNN	ΛCDM	0.650 ± 0.144	–	–	0.694 ± 0.172	0.292
	CPL	0.648 ± 0.142	-0.701 ± 0.271	-0.290 ± 0.540	0.777 ± 0.111	0.284
FFNN+MC-DO	ΛCDM	0.651 ± 0.147	–	–	0.652 ± 0.170	0.984
	CPL	0.632 ± 0.140	-0.674 ± 0.270	-0.156 ± 0.489	0.775 ± 0.110	0.960
		Datasets	JLA			
		h	w_0	w_a		
Original	ΛCDM	0.638 ± 0.146	–	–		33.214
	CPL	0.652 ± 0.141	-0.901 ± 0.238	-0.216 ± 0.899		32.354
FFNN	ΛCDM	0.645 ± 0.144	–	–		14.670
	CPL	0.640 ± 0.137	-1.092 ± 0.277	0.287 ± 0.957		13.888
FFNN+MC-DO	ΛCDM	0.643 ± 0.142	–	–		16.446
	CPL	0.641 ± 0.135	-1.037 ± 0.248	-0.245 ± 0.996		16.274

with previous works using different non-parametric inference techniques.

In general, the results of the neural networks with MC-DO are better because they are considering the uncertainty of the produced models, and the dropout technique provides regularization generating a more robust model. On the other hand, the standard deviations (or variance) of the FFNN+MC-DO predictions are small, which gives us the certainty that the neural network is well-trained. The FFNN+MC-DO predictions may have more variance than the FFNN alone, and the fact that the results obtained with both models are close allows us to conclude that the FFNN predictions are acceptable.

Because we are taking into account the original statistical errors as part of the training datasets, in the reconstructions of $H(z)$ and $f\sigma_8$, the errors have also been modeled by the neural networks. We are generating models for the errors; therefore, the new error bars are independent of a real data point at a given redshift, which is not the case in the Gaussian processes.

As seen in the appendices, a disadvantage of our method is that the neural networks training and their hyperparameter tuning are computationally more complex and have a higher CPU consumption than other interpolation or non-parametric inference techniques. However, our method offers some advantages that can make it a viable alternative:

- Well-trained neural network models can be generated even with few data points.
- No fiducial cosmology is necessary to generate model-independent neural reconstructions consistent with cosmological theory.
- No assumptions have to be made about the statistical distribution of the data.
- It allows computational models for observational data and their errors, even if they have correlations among them.

We have explored the generation of synthetic covariance matrices through a VAE neural network, and the results have allowed us to carry out Bayesian inference without drawbacks. The results we have obtained, as a first approach, are in agreement with other techniques. For larger datasets, we consider that using more complex architectures of autoencoders and a slightly different approach for dealing with the computing demand will be convenient.

It is worth mentioning that the results obtained in this work are for the chosen observations and have been sufficient to show some interesting features from the data alone. In this way, we can see that using neural networks for the model-independent reconstructions can complement the analysis of cosmological models and improve the interpretations of their behaviors. We plan to apply similar techniques to other data types, including a full set of covariance matrices, and also incorporate more sophisticated hyperparameter tuning to improve reconstructions.

Acknowledgements Thanks to the referees who helped improve this paper. CONACyT-Mexico Ph.D. and postdoctoral scholarships partially supported this work. RG-S acknowledges the support provided by SIP20210500-IPN, SIP20220481-IPN, and SIP20230505 grants and FORDECYT-PRONACES-CONACYT CF-MG-2558591. JAV acknowledges the support provided by FOSEC SEP-CONACYT Investigación Básica A1-S-21925, FORDECYT-PRONACES-CONACYT 304001, and UNAM-DGAPA-PAPIIT IA104221. IGV, thanks to ICF-UNAM.

Data Availability Statement This manuscript has no associated data or the data will not be deposited. [Authors' comment: The cosmological data used in this paper (SNIa, cosmic chronometers, and growth rate measurements) are third-party datasets that have been previously used in established works and are publicly available. As such, we are unable to provide original data or deposit them again. However, in Sect. 2.1, we have properly referenced and cited the sources of these datasets to ensure transparency and reproducibility.]

Funded by SCOAP^3. SCOAP^3 supports the goals of the International Year of Basic Sciences for Sustainable Development.

Appendix A: Neural networks basics

Here, we present the learning mechanism of an ANN with some of the settings we have used throughout this work:

- Before the ANN training, we split the original datasets into training and validation sets with 80% and 20%, respectively. The first set is used to train the ANN, while the validation set contains unseen values. Therefore, it helps test the performance of the ANN and evaluate its ability to produce an excellent model for the input dataset.
- The first layer of neurons reads the dataset's features (or columns). Each connection between neurons is assigned a random number called weight (we use random numbers with a normal distribution centered on 0 with a standard deviation of 0.01). The input data make up a matrix X_1 and provide the values for the first layer of nodes. The X_i refers to the values of nodes in the ith layer. The weights make up another matrix W_i, which are the values for the connections between the ith and the $(i+1)$th layers. In addition, each connection has a bias term b_i. The product Z of these two matrices plus the bias is as follows:

$$Z_{i+1} = W_i^T X_i + b_i, \tag{10}$$

  where $W_i \in \mathbb{R}^{m\times n}$, with m and n the number of nodes in the ith and $(i+1)$th layers respectively. X_i corresponds to the ith layer, therefore, has m dimensions. It is worth applying the transpose of W_i to allow the matrix product.
- An activation (or transfer) function ϕ modulates Z_i and assigns values to the next layer of neurons. This process, known as forward propagation, is repeated until the last layer is reached. The values of neurons in subsequent layers are given by:

$$X_{i+1} = \phi(Z_{i+1}). \tag{11}$$

- The value of the neurons in the last layer must be evaluated by an error function (or loss function) which measures the difference between the value given by the ANN and the expected one. In order to find the better values of weights, the loss function is minimized by an optimization method such as *gradient descent* combined with the *backpropagation* algorithm to compute gradients [98,99].
- During backpropagation, the weights are updated, then forward propagation is performed again. This is repeated until the loss function reaches the desired precision, and then the neural network is trained and ready to make predictions. The number of samples propagated through the network before updating the weights is known as *batch size*, and each iteration of the entire dataset constitutes an *epoch*.

Another essential concept is the dropout (DO), a regularization technique [100] that allows smaller values to be achieved in the loss function and prevents overfitting. It consists in randomly turning off neurons during training, so the neurons that operate at each epoch are different. The associated hyperparameter is a scalar value that indicates the probability of turning off a neuron in each epoch. Due to its random nature, the dropout can be used as a Monte Carlo simulation. When an ANN is trained, the dropout can be interpreted as a Bayesian approximation of a gaussian probabilistic model, during training the active neurons are different, hence different weight configurations are saved, and in each prediction, the neural network uses a different one, such as in each case we use a different neural network, with other neurons turned off. Therefore, it is possible to make several predictions and thus obtain the average and standard deviations. Using this formalism, dubbed Monte Carlo dropout (MC-DO) [91], we can get a statistical uncertainty of a trained ANN model. We apply the dropout method to the FFNNs implemented in this work and compare the results with those solely with FFNNs.

The intrinsic parameters of an ANN are known as hyperparameters: the number of layers, number of nodes, batch size, dropout value, optimizer algorithm, or number of epochs, among others. It is worth carefully selecting a good combination of them to guarantee that the ANN model has the capability of generalization. An incorrect choice of them can produce undesirable models, either underfitted or overfitted concerning the data.

Appendix B: Autoencoders

Autoencoders can be thought as two symmetrical coupled ANNs, where the first (encoder) makes a dimensional reduction for the input and obtains a coded representation (vector embedding or latent space) of the original data. The second part (decoder) takes the coded representation of the data and recovers an instance with the same data type and dimension as the original input. The encoder is a function f that maps the input x with dimension l to an encoded vector h with dimension m, with $m < l$:

$$f : x \in \mathbb{R}^l \to h \in \mathbb{R}^m, \qquad (12)$$

where $h_i := f_i(x) = \phi_e(W_i^T X_i + b_i^e),\ \ i = 1, 2, ..., m$ with ϕ being the activation function and the e index refers to the encoder. The decoder is the following g function that maps the encoded representation with dimension m into an output $\hat{x}$ with the same dimension l as the original input x:

$$g : h \in \mathbb{R}^m \to \hat{x} \in \mathbb{R}^l, \qquad (13)$$

with $\hat{x}_j = g_j(h) = \phi_d(W_j' h + b_j^d),\ \ i = 1, 2, ..., l$, where the d index refers to the decoder. The goal of the autoencoder training is to find the parameters of the functions shown in Eqs. (12) and (13): $[W_1, ..., W_m]$, b^e for the encoder and $[W_1', ..., W_m']$, b^d for the decoder. If the activation function is the identity function, i.e., $\phi(x) = x$, then this type of neural network is analogous to the Principal Component Analysis (PCA) technique. In this work, we use a particular type called variational autoencoder (VAE), which belongs to the so-called generative neural networks [101,102].

VAE neural networks use variational inference to sample the compressed representation (or latent space) and, therefore, allow us to know the probability function associated precisely, with the compressed representation. Unlike classical autoencoders, such as those described earlier in this work, two layers of the same dimension as the latent space are designed before the compressed representation, whose function is to generate values to sample the mean μ and variance σ, which are the parameters of the statistical distribution that produces an input data (matrix or image) of the VAE to generate a point z of the latent space.

To construct a latent space distribution similar to the proposed Gaussian distribution, the Kullback–Leiber divergence (KL) is used [103]. Thus, the selection of the relevant loss function to train the VAE is as follows:

$$loss_{\text{VAE}} = \text{MSE} + \mathbf{KL}(q(z|x)||p(z)), \qquad (14)$$

where $q(z|x)$ is the probability density function to generate a z point of the latent space given an input x. On the other hand, we can assume that $p(z) = N(0, I)$ with p a probability density function of the z points in latent space and N a normal distribution centered at 0 with covariance matrix equal to the identity matrix. Because VAEs are widely used in image processing, it is more common to choose *binary cross entropy* [104] instead of MSE. However, our interest is in the numerical information of the covariance matrices and not just in a classification problem in image generation.

For an extended review about Variational Autoencoders, we recommend [105,106].

Appendix C: Feedforward neural networks training

This appendix describes some aspects we consider in training our feedforward neural networks. Although the goal of the neural network training is to minimize the loss function, also the following relationship should be taken into account:

$$\text{MSE} = \text{bias}^2 + \text{variance}, \qquad (15)$$

where the bias measures how far away the neural network predictions are from the actual value, while the variance refers to how much the prediction varies at nearby points. As the ANN model gets more complex, the bias can decrease while the variance can increase. This is called the bias-variance dilemma [107]. A model with high variance will be overfitted, while a model with high bias will be insufficient to learn the complexity of the data (underfitting).

In both cases, the model generated by the neural network would have inaccurate predictions. One way to avoid this problem is by monitoring the behavior of the loss function throughout the training epochs, both in the training set and in the validation set, both of them must have similar behavior. For example, in Fig. 7, we can see the effect of the number of epochs in two of our neural networks used in this work. A common practice to prevent an incorrect fitting in the ANN model is increasing the training set's size. Otherwise,

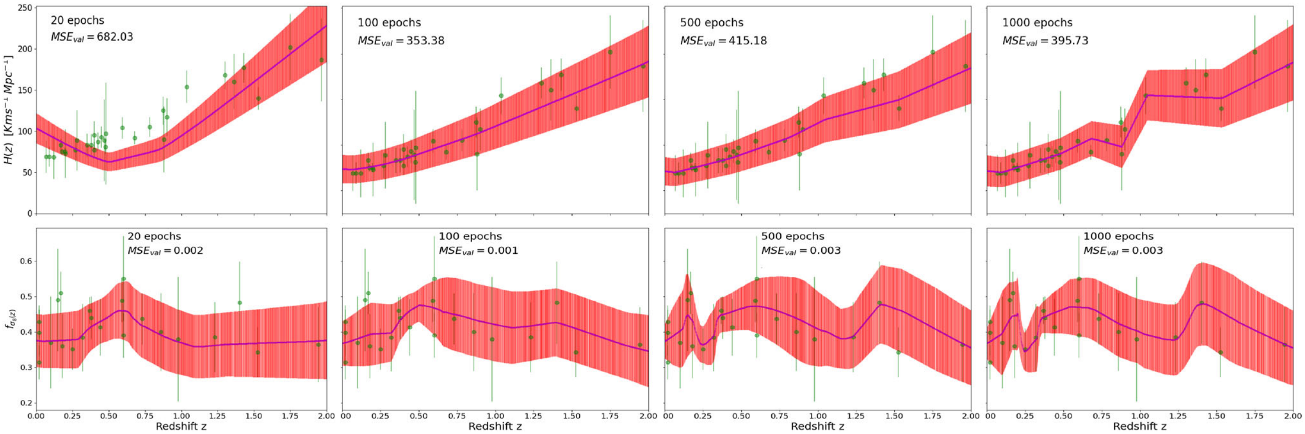

Fig. 7 Reconstructions from Artificial Neural Networks trained with different numbers of epochs. The effect of the epochs can be noticed in training with the CC dataset (top) and with the $f\sigma_8(z)$ measurements (bottom). The first case (20 epochs) shows underfitting while considering 1000 epochs shows overfitting. In the $f\sigma_8$ dataset, the cases for 500 and 1000 epochs present overfitting. In both cases, we choose 100 epochs due to the lowest value of MSE in the validation set. Green points display real data points with error bars, and in purple the neural reconstructions along with red error bars

it is worth it to carefully calibrate the ANN models' hyperparameters to achieve acceptable results. There are several approaches to tune the hyperparameters [108–111]. Because our ANNs have relatively simple architectures (between two and five hidden layers and just a few thousand neurons), we use a common empirical strategy based on a grid of hyperparameters [108] where several combinations of hyperparameters are evaluated to choose the best of them.

In general, for the three types of cosmological observations (CC, $f\sigma_8$ and SNeIa), we have followed the next steps to find out a suitable neural network model for the corresponding data:

- We train several neural network configurations to gain insights into the complexity of their architecture to model the data. According to the loss function results, we choose a number of layers.
- Several values are suggested for each hyperparameter of the neural network. Based on the intuition achieved in the first step, a grid is formed that must be traversed to find the combination that provides the minimum value of the loss function. Among the hyperparameters are the batch size, the number of nodes per layer, and, in some cases, the dropout.
- The best FNNN architectures found for each case are shown in Fig. 1. The first two correspond to the CC and $f\sigma_8$ datasets, respectively, for which 320 combinations were tested up to three hidden layers: number of nodes in {50, 100, 150, 200} and the batch size in {1, 4, 8, 16, 32}. We found that for the compressed JLA dataset, a one-layer neural network works best, so we refined the third architecture among 20 combinations, varying the number of nodes in {30, 50, 100, 150, 200} and the batch size in {1, 2, 4, 8}.
- We train the neural network with the combination of hyperparameters chosen in the previous step with a correct number of epochs. We verify the behavior of the loss function in the training and validation sets to check that our model is neither underfitted nor overfitted.
- Once the neural network is trained, we can generate model-independent reconstructions with several predictions and compare it with the original data.
- We compare the parameter estimation using data points from reconstructions with the original datasets to verify they are statistically consistent; if they are not, the neural networks must be retrained or other architecture to be used. For Bayesian inference, we use the `SimpleMC`[1] package, initially released at [112], along with a modified version of the `dynesty` nested sampling library [113], which allows to do the parameter estimation.

On the other hand, through the analysis of the JLA SNeIa compilation, we also use an FFNN to learn the behavior of the data from measurements of the distance modulus in a similar fashion we did for the CC and $f\sigma_8$. However, to handle the full covariance matrix, we use a VAE as described in the Appendix D; using this type of neural network allows us to map the distribution of the distance modulus data to the distribution of the coded representation of the autoencoder to generate new covariance matrices.

One restriction of this method to bear in mind is that the new matrix must have the same dimension as the original one. However, we can generate any matrix given a combination

[1] www.github.com/ja-vazquez/SimpleMC.

of new redshifts, provided that this set has the same length as the original measurements.

In addition to the above procedure, we slightly modify the FFNNs to implement MC-DO. In this way, we add dropout between the layers of the FFNNs and run the MC-DO several times to obtain average values and uncertainties for each prediction (as described in Sect. 2.3). We combine our FFNN designs with the implementation of MC-DO layers from `astronn` [114] and compare the results of this method with the previous ANNs implementations. Because dropout is a regularization technique, the number of epochs is irrelevant for a large enough set. The error predictions and the uncertainties are independent; therefore, the total standard deviation is:

$$\sigma = \sqrt{er_p + \sum_i u_i^2}, \tag{16}$$

where u_i is the epistemic uncertainty involved with the FFNN used and er_p is the error prediction.

Besides the intrinsic error associated with the datasets, we consider an uncertainty related to the FFNN by adding a Monte Carlo dropout between each layer of the chosen FNNN architecture. Among several tests to dropout values between [0, 0.1, 0.2, 0.3, 0.4, 0.5], we evaluated the loss function of the neural networks trained with these dropout values and found that the lower value of the loss function in the test and validation sets is with a dropout of 0.3 along 1000 epochs to the cosmic chronometers data, 0.1 along 2000

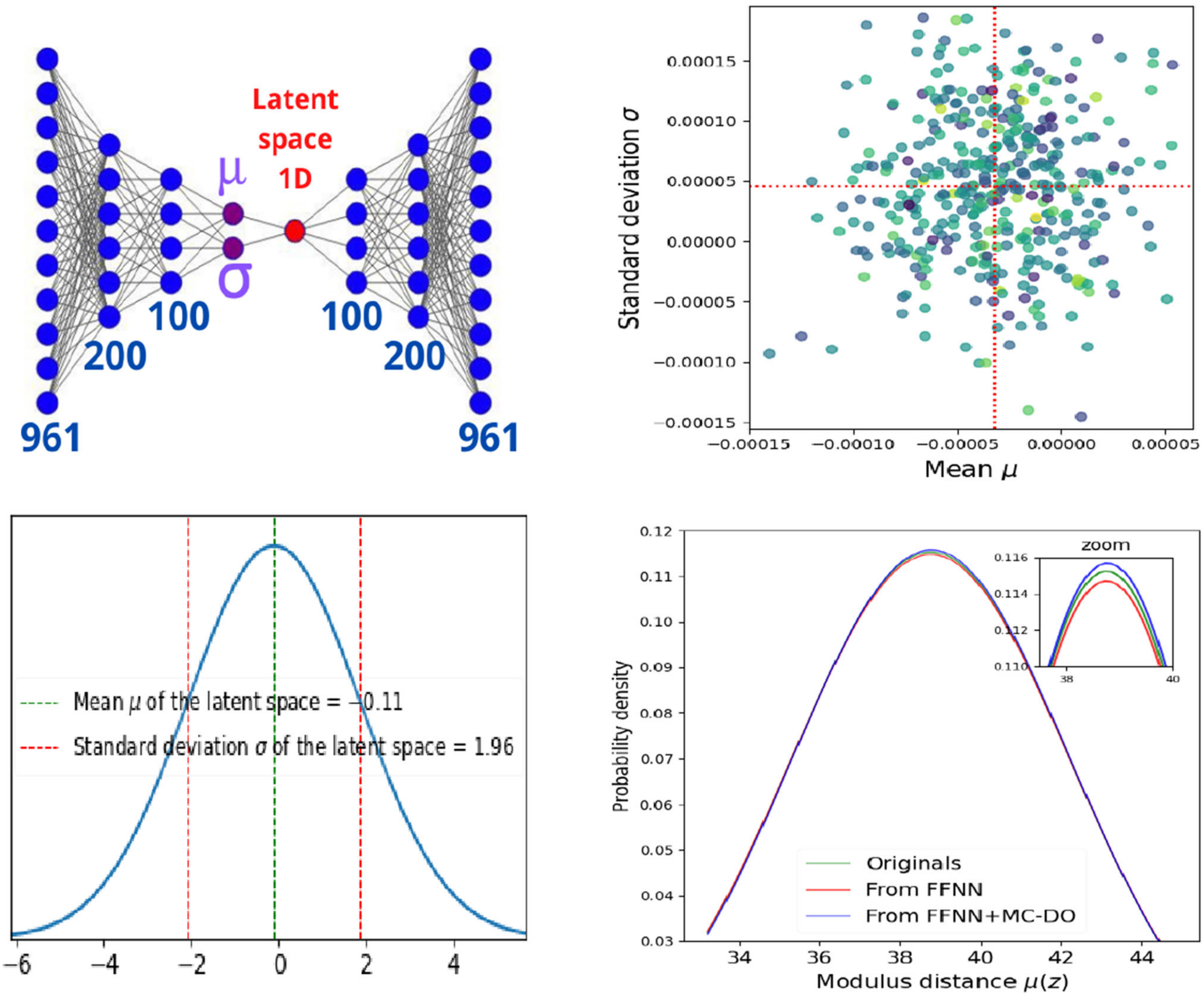

Fig. 8 VAE results. Top left: VAE architecture. Top right: Samples of the mean and variance layers. Lower left: Sampled distribution of the latent space. Lower right: Comparison between the distributions for the modulus distances from different sources mapped into the latent space to generate a new covariance matrix with the VAE decoder

epochs to $f_{\sigma 8}$ data, and 0.01 to the SNeIa data with 1000 epochs. After training the FFNNs with MC-DO, we made 100 executions of MC-DO for each prediction.

Appendix D: Variational autoencoder for non-diagonal covariance matrix

In this appendix, we explain a new method used to generate synthetic covariance matrices from the original matrix of the JLA SNeIa binned version. When the VAE is trained, it samples the probability distribution of the latent space, and because this covariance matrix is associated with the distance modulus measurements, therefore, we map its distribution to the latent space distribution.

To have a dataset to train our VAE, we generated thousands of matrices by adding Gaussian noise of the same order of magnitude for each entry of the original covariance matrix. We assume that predictions at new redshifts within the current range of redshifts could have a similar covariance matrix, this assumption may be useful to test the method in a first approximation. We designed the VAE architecture, shown in the first panel of Fig. 8, to generate synthetic covariance matrices from a single point in the latent space. This VAE was trained on a dataset created from the systematic error covariance matrix of the JLA binned version. μ and σ represent two layers connected to the last layer of the encoder and the latent space; in this case, both layers have a single neuron (the same dimension as the latent space). We use a batch size of 32 and the hyperbolic tangent as the activation function.

Since we are interested on mapping the distribution of the distance modulus to the latent space, we design the VAE with a 1-dimensional latent space, so its mean μ and variance σ are also 1-dimensional. After training the VAE, as in other generative neural networks, we can use the decoder part to generate new covariance matrices that traverse the latent space. Once the VAE is trained, we can explore the mean, variance, and latent space layers, as seen in the second and third panels of Fig. 8. To generate covariance matrices from the predictions of the modular distances coming from the FFNNs, using their means and standard deviations, we have assigned them a Gaussian distribution (fourth panel in Fig. 8). We have related the original measurements to the most likely region of the latent space. The deviations from the original measurements can be linearly mapped to the latent space to generate a new covariance matrix, as shown in Fig. 5.

In our case, the VAE only learns to generate 31 × 31 covariance matrices, so we can only generate 31 predicted SNeIa sets. We use a variational autoencoder because the latent space has a probability distribution sampled during training. We map this distribution and the distance modulus using the decoder map. We generate a new covariance matrix for a different modulus of distance values; it is an experimental procedure, but it seems to work.

References

1. E.J. Copeland, M. Sami, S. Tsujikawa, Dynamics of dark energy. Int. J. Mod. Phys. D **15**(11), 1753–1935 (2006)
2. L. Amendola, S. Tsujikawa, *Dark Energy: Theory and Observations* (Cambridge University Press, Cambridge, 2010)
3. P. Ruiz-Lapuente, *Dark Energy: Observational and Theoretical Approaches* (Cambridge University Press, Cambridge, 2010)
4. N. Aghanim, Y. Akrami, M. Ashdown, J. Aumont, C. Baccigalupi, M. Ballardini, A.J. Banday, R.B. Barreiro, N. Bartolo, S. Basak et al., Planck 2018 results-vi. Cosmological parameters. Astron. Astrophys. **641**, A6 (2020)
5. M. Betoule, R. Kessler, J. Guy, J. Mosher, D. Hardin, R. Biswas, P. Astier, P. El-Hage, M. Konig, S. Kuhlmann et al., Improved cosmological constraints from a joint analysis of the sdss-ii and snls supernova samples. Astron. Astrophys. **568**, A22 (2014)
6. D. Stern, R. Jimenez, L. Verde, M. Kamionkowski, S. Adam, Stanford, cosmic chronometers: constraining the equation of state of dark energy. I: $h(z)$ measurements. J. Cosmol. Astropart. Phys. **2010**(02), 008 (2010)
7. S. Alam, M. Ata, S. Bailey, F. Beutler, D. Bizyaev, J.A. Blazek, A.S. Bolton, J.R. Brownstein, A. Burden, C.-H. Chuang et al., The clustering of galaxies in the completed sdss-iii baryon oscillation spectroscopic survey: cosmological analysis of the dr12 galaxy sample. Mon. Not. R. Astron. Soc. **470**(3), 2617–2652 (2017)
8. V. Sahni, The cosmological constant problem and quintessence. Class. Quantum Gravity **19**(13), 3435 (2002)
9. P.J.E. Peebles, The cosmological constant and dark energy. Rev. Mod. Phys. **75**(2), 559 (2003)
10. S.M. Feeney, D.J. Mortlock, N. Dalmasso, Clarifying the hubble constant tension with a Bayesian hierarchical model of the local distance ladder. Mon. Not. R. Astron. Soc. **476**(3), 3861–3882 (2018)
11. A. Joyce, L. Lombriser, F. Schmidt, Dark energy versus modified gravity. Annu. Rev. Nucl. Part. Sci. **66**, 95–122 (2016)
12. J.A. Tyson, Large synoptic survey telescope: overview. Surv. Telesc. Technol. Discov. **4836**, 10–20 (2002)
13. A. Aghamousa, J. Aguilar, S. Ahlen, S. Alam, L.E. Allen, C. Allende Prieto, J. Annis, S. Bailey, C. Balland, O. Ballester, et al., The desi experiment part i: science, targeting, and survey design (2016). Preprint arXiv:1611.00036
14. L. Amendola, S. Appleby, A. Avgoustidis, D. Bacon, T. Baker, M. Baldi, N. Bartolo, A. Blanchard, C. Bonvin, S. Borgani et al., Cosmology and fundamental physics with the euclid satellite. Living Rev. Relativ. **21**(1), 2 (2018)
15. P. Brax, C. van de Bruck, Cosmology and brane worlds: a review. Class. Quantum Gravity **20**(9), R201 (2003)
16. T. Clifton, P.G. Ferreira, A. Padilla, C. Skordis, Modified gravity and cosmology. Phys. Rep. **513**(1–3), 1–189 (2012)
17. J.A. Vázquez, D. Tamayo, A.A. Sen, I. Quiros, Bayesian model selection on scalar ϵ-field dark energy. Phys. Rev. D **103**(4), 043506 (2021)
18. I. Quiros, T. Gonzalez, U. Nucamendi, R. Garcia-Salcedo, F.A. Horta-Rangel, J. Saavedra, On the phantom barrier crossing and the bounds on the speed of sound in non-minimal derivative coupling theories. Class. Quantum Gravity **35**(7), 075005 (2018)
19. Ö. Akarsu, J.D. Barrow, L.A. Escamilla, J.A. Vazquez, Graduated dark energy: observational hints of a spontaneous sign switch in the cosmological constant. Phys. Rev. D **101**(6), 063528 (2020)

20. M. Chevallier, D. Polarski, Accelerating universes with scaling dark matter. Int. J. Mod. Phys. D **10**(02), 213–223 (2001)
21. I. Sendra, R. Lazkoz, Supernova and baryon acoustic oscillation constraints on (new) polynomial dark energy parametrizations: current results and forecasts. Mon. Not. R. Astron. Soc. **422**(1), 776–793 (2012)
22. S.D. Odintsov, V.K. Oikonomou, A.V. Timoshkin, E.N. Saridakis, R. Myrzakulov, Cosmological fluids with logarithmic equation of state. Ann. Phys. **398**, 238–253 (2018)
23. D. Tamayo, J.A. Vazquez, Fourier-series expansion of the dark-energy equation of state. Mon. Not. R. Astron. Soc. **487**(1), 729–736 (2019)
24. J. Liu, H. Li, J.-Q. Xia, X. Zhang, Testing oscillating primordial spectrum and oscillating dark energy with astronomical observations. J. Cosmol. Astropart. Phys. **2009**(07), 017 (2009)
25. G. Arciniega, M. Jaber, L.G. Jaime, O.A. Rodríguez-López, One parameterisation to fit them all (2021). Preprint arXiv:2102.08561
26. Ö. Akarsu, T. Dereli, J.A. Vazquez, A divergence-free parametrization for dynamical dark energy. J. Cosmol. Astropart. Phys. **06**, 049 (2015)
27. L. Wasserman, *All of Nonparametric Statistics* (Springer Science & Business Media, Berlin, 2006)
28. V. Sahni, A. Starobinsky, Reconstructing dark energy. Int. J. Mod. Phys. D **15**(12), 2105–2132 (2006)
29. R. Sharma, A. Mukherjee, H.K. Jassal, Reconstruction of late-time cosmology using principal component analysis (2020). Preprint arXiv:2004.01393
30. F. Gerardi, M. Martinelli, A. Silvestri, Reconstruction of the dark energy equation of state from latest data: the impact of theoretical priors. J. Cosmol. Astropart. Phys. **2019**(07), 042 (2019)
31. C.K.I. Williams, C.E. Rasmussen, *Gaussian Processes for Machine Learning*, vol. 2 (MIT Press, Cambridge, 2006)
32. E.R. Keeley, A. Shafieloo, G.-B. Zhao, J.A. Vazquez, H. Koo, Reconstructing the universe: testing the mutual consistency of the pantheon and SDSS/eBOSS BAO data sets with gaussian processes. Astron. J. **161**(3), 151 (2021)
33. B. L'Huillier, A. Shafieloo, D. Polarski, A.A. Starobinsky, Defying the laws of gravity i: model-independent reconstruction of the universe expansion from growth data. Mon. Not. R. Astron. Soc. **494**(1), 819–826 (2020)
34. P. Mukherjee, N. Banerjee, Revisiting a non-parametric reconstruction of the deceleration parameter from observational data (2020). Preprint arXiv:2007.15941
35. A. Montiel, R. Lazkoz, I. Sendra, C. Escamilla-Rivera, V. Salzano, Nonparametric reconstruction of the cosmic expansion with local regression smoothing and simulation extrapolation. Phys. Rev. D **89**(4), 043007 (2014)
36. J.A. Vazquez, M. Bridges, M.P. Hobson, A.N. Lasenby, Reconstruction of the Dark Energy equation of state. J. Cosmol. Astropart. Phys. **09**, 020 (2012)
37. S. Hee, J.A. Vázquez, W.J. Handley, M.P. Hobson, A.N. Lasenby, Constraining the dark energy equation of state using Bayes theorem and the kullback-leibler divergence. Mon. Not. R. Astron. Soc. **466**(1), 369–377 (2017)
38. T. Holsclaw, U. Alam, B. Sanso, H. Lee, K. Heitmann, S. Habib, D. Higdon, Nonparametric dark energy reconstruction from supernova data. Phys. Rev. Lett. **105**(24), 241302 (2010)
39. G.-B. Zhao, M. Raveri, L. Pogosian, Y. Wang, R.G. Crittenden, W.J. Handley, W.J. Percival, F. Beutler, J. Brinkmann, C.-H. Chuang et al., Dynamical dark energy in light of the latest observations. Nat. Astron. **1**(9), 627–632 (2017)
40. J.-J. Wei, W. Xue-Feng, An improved method to measure the cosmic curvature. Astrophys. J. **838**(2), 160 (2017)
41. H.-N. Lin, X. Li, L. Tang, Non-parametric reconstruction of dark energy and cosmic expansion from the pantheon compilation of type ia supernovae. Chin. Phys. C **43**(7), 075101 (2019)
42. J.A. Vazquez, M. Bridges, M.P. Hobson, A.N. Lasenby, Model selection applied to reconstruction of the primordial power spectrum. J. Cosmol. Astropart. Phys. **06**, 006 (2012)
43. W.J. Handley, A.N. Lasenby, H.V. Peiris, M.P. Hobson, Bayesian inflationary reconstructions from Planck 2018 data. Phys. Rev. D **100**(10), 103511 (2019)
44. W.H. Lin, M. Tegmark, D. Rolnick, Why does deep and cheap learning work so well? J. Stat. Phys. **168**(6), 1223–1247 (2017)
45. A. Peel, F. Lalande, J.-L. Starck, V. Pettorino, J. Merten, C. Giocoli, M. Meneghetti, M. Baldi, Distinguishing standard and modified gravity cosmologies with machine learning. Phys. Rev. D **100**(2), 023508 (2019)
46. R. Arjona, S. Nesseris, What can machine learning tell us about the background expansion of the universe? Phys. Rev. D **101**(12), 123525 (2020)
47. G.-J. Wang, X.-J. Ma, J.-Q. Xia, Machine learning the cosmic curvature in a model-independent way. Mon. Not. R. Astron. Soc. **501**(4), 5714–5722 (2021)
48. I. Gómez-Vargas, R.M. Esquivel, R. García-Salcedo, J.A. Vázquez, Neural network within a Bayesian inference framework. J. Phys. Conf. Ser. **1723**(1), 012022 (2021)
49. J. Chacón, J.A. Vázquez, E. Almaraz, Classification algorithms applied to structure formation simulations 6 (2021). arXiv:2106.06587
50. S. Dieleman, K.W. Willett, J. Dambre, Rotation-invariant convolutional neural networks for galaxy morphology prediction. Mon. Not. R. Astron. Soc. **450**(2), 1441–1459 (2015)
51. M. Ntampaka, J. ZuHone, D. Eisenstein, D. Nagai, A. Vikhlinin, L. Hernquist, F. Marinacci, D. Nelson, R. Pakmor, A. Pillepich et al., A deep learning approach to galaxy cluster x-ray masses. Astrophys. J. **876**(1), 82 (2019)
52. A.C. Rodríguez, T. Kacprzak, A. Lucchi, A. Amara, R. Sgier, J. Fluri, T. Hofmann, A. Réfrégier, Fast cosmic web simulations with generative adversarial networks. Comput. Astrophys. Cosmol. **5**(1), 4 (2018)
53. S. He, Y.Y. Li, S.H. Feng, S. Ravanbakhsh, W. Chen, B. Póczos, Learning to predict the cosmological structure formation. Proc. Natl. Acad. Sci. **116**(28), 13825–13832 (2019)
54. T. Auld, M. Bridges, M.P. Hobson, S.F. Gull, Fast cosmological parameter estimation using neural networks. Mon. Not. R. Astron. Soc. Lett. **376**(1), L11–L15 (2007)
55. J. Alsing, T. Charnock, S. Feeney, B. Wandelt, Fast likelihood-free cosmology with neural density estimators and active learning. Mon. Not. R. Astron. Soc. **488**(3), 4440–4458 (2019)
56. S.-Y. Li, Y.-L. Li, T.-J. Zhang, Model comparison of dark energy models using deep network. Res. Astron. Astrophys. **19**(9), 137 (2019)
57. H.J. Hortúa, L. Malagò, R. Volpi, Constraining the reionization history using Bayesian normalizing flows. Mach. Learn. Sci. Technol. **1**(3), 035014 (2020)
58. H.J. Hortúa, R. Volpi, D. Marinelli, L. Malagò, Parameter estimation for the cosmic microwave background with bayesian neural networks. Phys. Rev. D **102**(10), 103509 (2020)
59. C. Escamilla-Rivera, M.A.C. Quintero, S. Capozziello, A deep learning approach to cosmological dark energy models. J. Cosmol. Astropart. Phys. **2020**(03), 008 (2020)
60. G.-J. Wang, X.-J. Ma, S.-Y. Li, J.-Q. Xia, Reconstructing functions and estimating parameters with artificial neural networks: a test with a hubble parameter and sne ia. Astrophys. J. Suppl. Ser. **246**(1), 13 (2020)
61. K. Dialektopoulos, J.L. Said, J. Mifsud, J. Sultana, K.Z. Adami, Neural network reconstruction of late-time cosmology and null tests (2021). arXiv preprint arXiv:2111.11462
62. E.V. Linder, Exploring the expansion history of the universe. Phys. Rev. Lett. **90**(9), 091301 (2003)

63. R. Jimenez, L. Verde, T. Treu, D. Stern, Constraints on the equation of state of dark energy and the hubble constant from stellar ages and the cosmic microwave background. Astrophys. J. **593**(2), 622 (2003)
64. J. Simon, L. Verde, R. Jimenez, Constraints on the redshift dependence of the dark energy potential. Phys. Rev. D **71**(12), 123001 (2005)
65. M. Moresco, L. Verde, L. Pozzetti, R. Jimenez, A. Cimatti, New constraints on cosmological parameters and neutrino properties using the expansion rate of the universe to $z \sim 1.75$. J. Cosmol. Astropart. Phys. **2012**(07), 053 (2012)
66. C. Zhang, H. Zhang, S. Yuan, S. Liu, T.-J. Zhang, Y.-C. Sun, Four new observational $h(z)$ data from luminous red galaxies in the sloan digital sky survey data release seven. Res. Astron. Astrophys. **14**(10), 1221 (2014)
67. M. Moresco, Raising the bar: new constraints on the hubble parameter with cosmic chronometers at $z \sim 2$. Mon. Not. R. Astron. Soc. Lett. **450**(1), L16–L20 (2015)
68. M. Moresco, L. Pozzetti, A. Cimatti, R. Jimenez, C. Maraston, L. Verde, D.T.A. Citro, R. Tojeiro, D. Wilkinson, A 6% measurement of the hubble parameter at $z \sim 0.45$: direct evidence of the epoch of cosmic re-acceleration. J. Cosmol. Astropart. Phys. **2016**(05), 014 (2016)
69. A.L. Ratsimbazafy, S.I. Loubser, S.M. Crawford, C.M. Cress, B.A. Bassett, R.C. Nichol, P. Väisänen, Age-dating luminous red galaxies observed with the southern African large telescope. Mon. Not. R. Astron. Soc. **467**(3), 3239–3254 (2017)
70. K. Said, M. Colless, C. Magoulas, J.R. Lucey, M.J. Hudson, Joint analysis of 6dfgs and sdss peculiar velocities for the growth rate of cosmic structure and tests of gravity. Mon. Not. R. Astron. Soc. **497**(1), 1275–1293 (2020)
71. N. Kaiser, Clustering in real space and in redshift space. Mon. Not. R. Astron. Soc. **227**(1), 1–21 (1987)
72. L. Amendola, M. Kunz, D. Sapone, Measuring the dark side (with weak lensing). J. Cosmol. Astropart. Phys. **2008**(04), 013 (2008)
73. B. Sagredo, S. Nesseris, D. Sapone, Internal robustness of growth rate data. Phys. Rev. D **98**(8), 083543 (2018)
74. R. Trotta, Bayes in the sky: Bayesian inference and model selection in cosmology. Contemp. Phys. **49**(2), 71–104 (2008)
75. F. Feroz, M.P. Hobson, M. Bridges, Multinest: an efficient and robust Bayesian inference tool for cosmology and particle physics. Mon. Not. R. Astron. Soc. **398**(4), 1601–1614 (2009)
76. F. Leclercq, Bayesian optimization for likelihood-free cosmological inference. Phys. Rev. D **98**(6), 063511 (2018)
77. A.N. Taylor, T.D. Kitching, Analytic methods for cosmological likelihoods. Mon. Not. R. Astron. Soc. **408**(2), 865–875 (2010)
78. S. Nesseris, J. Garcia-Bellido, A new perspective on dark energy modeling via genetic algorithms. J. Cosmol. Astropart. Phys. **2012**(11), 033 (2012)
79. S. Hannestad, Stochastic optimization methods for extracting cosmological parameters from cosmic microwave background radiation power spectra. Phys. Rev. D **61**(2), 023002 (1999)
80. J. Prasad, T. Souradeep, Cosmological parameter estimation using particle swarm optimization. Phys. Rev. D **85**(12), 123008 (2012)
81. L.E. Padilla, L.O. Tellez, L.A. Escamilla, J.A. Vazquez, Cosmological parameter inference with Bayesian statistics. Universe **7**(7), 213 (2021)
82. R.M. Esquivel, I. Gómez-Vargas, J.A. Vázquez, R.G. Salcedo, An introduction to Markov chain Monte Carlo. Boletín de Estadística e Investigación Operativa **1**(37), 47–74 (2021)
83. J. de Dios, R. Olvera, I. Gómez-Vargas, J.A. Vázquez, Observational cosmology with artificial neural networks. Universe **8**(2), 120 (2022)
84. K. Hornik, M. Stinchcombe, H. White, Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. Neural Netw **3**(5), 551–560 (1990)
85. I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, *Deep Learning*, vol. 1 (MIT Press, Cambridge, 2016)
86. P. Baldi, K. Hornik, Neural networks and principal component analysis: Learning from examples without local minima. Neural Netw **2**(1), 53–58 (1989)
87. S. Ingrassia, I. Morlini, Neural network modeling for small datasets. Technometrics **47**(3), 297–311 (2005)
88. H.-W. Ng, V.D. Nguyen, V. Vonikakis, S. Winkler, Deep learning for emotion recognition on small datasets using transfer learning, in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, (2015), p. 443–449
89. A. Pasini, Artificial neural networks for small dataset analysis. J. Thorac. Dis. **7**(5), 953 (2015)
90. Y. Wang, Q. Yao, J.T. Kwok, L.M. Ni, Generalizing from a few examples: a survey on few-shot learning. ACM Comput. Surv. (CSUR) **53**(3), 1–34 (2020)
91. Y. Gal, Z. Ghahramani, Dropout as a Bayesian approximation: insights and applications. Deep Learn. Workshop ICML **1**, 2 (2015)
92. A.G. Riess, L.M. Macri, S.L. Hoffmann, D. Scolnic, S. Casertano, A.V. Filippenko, B.E. Tucker, M.J. Reid, D.O. Jones, J.M. Silverman et al., A 2.4% determination of the local value of the hubble constant. Astrophys. J. **826**(1), 56 (2016)
93. H. Singirikonda, S. Desai, Model comparison of λcdm vs $r_h = ct$ using cosmic chronometers. Eur. Phys. J. C **80**(8), 1–9 (2020)
94. P. Mukherjee, A. Mukherjee, Assessment of the cosmic distance duality relation using gaussian process. Mon. Not. R. Astron. Soc. **504**(3), 3938–3946 (2021)
95. A. Bonilla, S. Kumar, R.C. Nunes, Measurements of h_0 and reconstruction of the dark energy properties from a model-independent joint analysis. Eur. Phys. J. C **81**(2), 1–13 (2021)
96. C. Ma, T.-J. Zhang, Power of observational hubble parameter data: a figure of merit exploration. Astrophys. J. **730**(2), 74 (2011)
97. Z.-Y. Yin, H. Wei, Non-parametric reconstruction of growth index via gaussian processes. Sci. China Phys. Mech. Astron. **62**(9), 1–10 (2019)
98. D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors. Nature **323**(6088), 533–536 (1986)
99. Y.A. LeCun, L. Bottou, G.B. Orr, K.-R. Müller, Efficient backprop. Neural networks: Tricks of the trade 9–48 (2012)
100. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
101. D.P. Kingma, M. Welling, Auto-encoding variational bayes (2013). Preprint arXiv:1312.6114
102. D.J. Rezende, S. Mohamed, D. Wierstra, Stochastic backpropagation and approximate inference in deep generative models, in *The International Conference on Machine Learning* (2014), p. 1278–1286
103. S. Kullback, R.A. Leibler, On information and sufficiency. Ann. Math. Stat. **22**(1), 79–86 (1951)
104. D. Ramos, J. Franco-Pedroso, A. Lozano-Diez, J. Gonzalez-Rodriguez, Deconstructing cross-entropy for probabilistic binary classifiers. Entropy **20**(3), 208 (2018)
105. C. Doersch, Tutorial on variational autoencoders (2016). Preprint arXiv:1606.05908
106. D.P. Kingma, M. Welling, An introduction to variational autoencoders (2019). Preprint arXiv:1906.02691
107. S. Geman, E. Bienenstock, R. Doursat, Neural networks and the bias/variance dilemma. Neural Comput. **4**(1), 1–58 (1992)
108. H. Larochelle, D. Erhan, A. Courville, J. Bergstra, Y. Bengio, An empirical evaluation of deep architectures on problems with many factors of variation, in *Proceedings of 24th International Conference Machine Learning* (2007), p. 473–480

109. F. Hutter, H.H. Hoos, K. Leyton-Brown, T. Stützle, Paramils: an automatic algorithm configuration framework. J. Artif. Intell. Res. **36**, 267–306 (2009)

110. R. Bardenet, M. Brendel, B. Kégl, M. Sebag, Collaborative hyperparameter tuning, in *Proceedings of 30th International Conference Machine Learning*, vol. 28(2) (2013), p. 199–207

111. X. Zhang, X. Chen, L. Yao, C. Ge, M. Dong, Deep neural network hyperparameter optimization with orthogonal array tuning, in *The International Conference on Neural Information Processing* (2019), p. 287–295

112. É. Aubourg, S. Bailey, J.E. Bautista, F. Beutler, V. Bhardwaj, D. Bizyaev, M. Blanton, M. Blomqvist, A.S. Bolton, J. Bovy et al., Cosmological implications of baryon acoustic oscillation measurements. Phys. Rev. D **92**(12), 123516 (2015)

113. S. Joshua Speagle, dynesty: a dynamic nested sampling package for estimating Bayesian posteriors and evidences. Mon. Not. R. Astron. Soc. **493**(3), 3132–3158 (2020)

114. H.W. Leung, Deep learning of multi-element abundances from high-resolution spectroscopic data. Mon. Not. R. Astron. Soc. **483**(3), 3255–3277 (2019)