SISSA

A deep learning approach to cosmological dark energy models

View the article online for updates and enhancements.

IOP ebooks™

Bringing you innovative digital publishing with leading voices
to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of
every title for free.

A deep learning approach to cosmological dark energy models

Celia Escamilla-Rivera,[a,1] **Maryi A. Carvajal Quintero**[b] **and Salvatore Capozziello**[c,d,e]

[a]Instituto de Ciencias Nucleares, Universidad Nacional Autónoma de México,
  Circuito Exterior C.U., A.P. 70-543, México D.F. 04510, México
[b]Universidad de Antioquia,
  Calle 70 No. 52-21, Apartado Aéreo 1226, Antioquia, Colombia
[c]Dipartimento di Fisica, University of Napoli "Federico II"
  and Istituto Nazionale di Fisica Nucleare (INFN), Sezione di Napoli,
  via Cinthia 9, I-80126 Napoli, Italy
[d]Gran Sasso Science Institute (GSSI),
  via F. Crispi 7, I-67100 L'Aquila, Italy
[e]Laboratory for Theoretical Cosmology,
  Tomsk State University of Control Systems and Radioelectronics (TUSUR),
  634050 Tomsk, Russia
  E-mail: celia.escamilla@nucleares.unam.mx, maryi.carvajal@udea.edu.co,
  capozziello@na.infn.it

Abstract. We propose a novel deep learning tool in order to study the evolution of dark energy models. The aim is to combine two architectures: the Recurrent Neural Networks (RNN) and the Bayesian Neural Networks (BNN), we named this full network as RNN+BNN. The first one is capable of learning complex sequential information to classify objects like supernovae and use the light-curves directly to learn information from the sequence of observations. Since RNN is not capable to calculate the uncertainties, BNN emerges as a solution for problems in deep learning like, for example, the overfitting. For the trainings we use measurements of the distance modulus $\mu(z)$, such as those provided by Pantheon Supernovae Type Ia. In view of our results, the reported approach turns out to be a first promising step on how we can train a new neural network that can compute their own confidence regions for specific cosmological data. It is worth stressing that the new technique allows to reduce the computational load of expensive codes for dark energy models and probe the necessity of modified dark energy models at large redshifts for a supernovae trained sampler.

[1]https://orcid.org/0000-0002-8929-250X.

Contents

1 Introduction

What is dark energy? This question has been addressed in a fundamental context since the discovery of the cosmic acceleration [1, 2] and what can be driving it. Its nature is one of the most fundamental issues facing cosmologists today and it is the subject of several current and future experiments that will survey the sky, e.g DESI [3], DES [4], WFIRST [6] and LSST [5]. Adjoint to these surveys, current observations as SNeIa, BAO, CMBR anisotropies, LSS formation and WL set a strong confirmation of the present-day accelerated expansion of the universe, consistent with the current standard cosmological model ΛCDM, where the Λ is usually related to an extra constant component of cosmological fluid (with equation of state — EoS — $w = -1$). Such a cosmological constant, if evolving, is dubbed dark energy. However, we cannot measure directly dark energy because we can only observe its effects on the Hubble flow measuring observable components like matter and radiation. Since the standard model interplays gravity and dark energy, we can encode the distribution of matter in the universe in $w = p(\rho)$, where p is the pressure and ρ the matter-energy density. As a simple choice, we can assume $w = p/\rho$. It can be characterised by a distribution derived from statistical measures in two (or three)-point correlation functions [7, 8] or other frequentist statistics [9–13].

Currently, a great debate is around the validity of ΛCDM model, where tensions are arising between Planck [14] and other cosmological measurements as: Cefeids (SH0ES), strong lensing time delays (H0LiCOW), tip of the red giant branch (TRGB), megamasers, Oxygen-rich Miras and surface brightness fluctuations [15]. These tensions which justify the study of possible alternatives to the concordance model. One of the most interesting approach seeks for dynamical properties of dark energy, which should be able to mimic Λ at the present time as required by the cosmological observations. The approach starts from quintessence scenarios [16, 17], dark energy parameterisations [18–22], modify gravity [23, 24], extended theories

of gravity [25], $w(z)$ reconstructions [26], non-parametric reconstructions of $w(z)$ [27, 28], till Bayesian reconstruction of a time-dependent EoS [29] or dynamical w_x from $f(R)$ models [30, 31] which offers a large overview on how we are trying to explain the effects of dark energy. However, a consensus of a unique model is still missing and all the proposals imply a dependency which can be significantly different by imposing a different theoretical landscape. Clearly, current and future data from the surveys will certainly clarify the issue by improving the determination of H_0 and $w(z)$, meanwhile we are dealing with wide distributions of matter proposals.

In order to present a relevant step in the study of dark energy, one can use a technique capable to combine all features in the distribution of matter that can give an insight into the nature of dark energy. A powerful technique is the deep learning (DL), which is a field of machine learning (ML) that uses several layers of nonlinear processing neurons to obtain and transform at each successive layer an output from the previous layer [33]. The algorithms for this subset are created and operate similar to those in ML, but there are numerous layers of these algorithms, each providing a different interpretation to the data it feeds on. Many problems in modern cosmological data analysis are tightly related to fundamental problems in ML, such as classifying stars and galaxies and cluster finding of dense galaxy populations [34, 35]. Other typical problems include data reduction, probability density estimation, how to deal with missing data and how to combine data from different surveys [36–38]. Unfortunately, up to now, there is not target distribution of $w(z)$ for dark energy where one has an efficient parameterisation with an error parameter space. An increasing part of modern cosmology aims at the development of new statistical data analysis tools and the study of their behaviour and systematics is often not aware of recent developments in ML and computational statistics. Of course, with the emergence of big data in recent years, the DL architecture will optimise algorithms that can handle the complexity of new cosmological models.

Here we present a novel technique to study dark energy models using a DL trained network in order to identify theoretical cosmological models through the data processed within layers of the network. In comparison to ML, DL technique does not require any labeled/structure data, as it relies on the different outputs processed by each layer which is amalgamated to form an unified way to identify theoretical models via images (matrices of observational data). Another novel result of our proposal is the ability to compute the confidence regions for each kind of training, within so far in ML astrophysical methods this has not been employed before. For this purpose, we combine the architecture of Recurrent Neural Networks (RNN) with the architecture of Bayesian Neural Networks (BNN) in view of a join network RNN+BNN representation learning, which can yield parameter spaces for the cosmological models. The advantage of this idea is that the proposed RNN+BNN network is trained and tested on observed supernovae data, and otherwise has no physical parameterisation related to the physical processes of cosmic acceleration. This is different in comparison if we train simulated data, e.g CMB and gravitational lensing, where a physical parameterisation needs to be used.

This paper is organised as follow: in section 2 we describe four cosmological models: two standard ones and two unified dark fluid equations of state, in order to point out the issues to achieve the cosmological analysis. In section 3 we denote the astrophysical sampler used. In section 4 we explain the basics of the deep learning technique employed to design a standard RNN architecture and how to add the Bayesian architecture in order to solve problems like overfitting. In section 5 we present the methodology to train the observational supernovae data using four activation functions on our proposed neural network RNN+BNN.

In section 6 we perform a bayesian comparison on the robustness of our RNN+BNN network approach against the proposed cosmologies, and use a ΛCDM simple scenario to demonstrate how to carry out parameter estimation from our networks results. Finally in section 7 we discuss our former results.

2 Models for dark energy equations of state

In this paper we are going to consider four dark energy models: from the standard scenarios as ΛCDM and CPL to unified dark energy fluid scenarios as Generalised Chaplygin gas (GCG) and Modified Chaplygin gas (MCG) models. In the following lines we enlisted these dark energy models with their respectively mathematical derivation.

ΛCDM is a Friedmann-Robertson-Walker (FRW) cosmology with a partition ρ into (at least) three components: matter ρ_m, radiation ρ_r and a poorly understood dark energy ρ_Λ, where the latter goes one step further by also invoking the constraint $w = -1/3$. Towards this direction, many attempts have already been done, starting from the above simple relation to several complex ones, always an explicit relation between the pressure p and density ρ.

- ΛCDM model. We shall take the standard model: $H(z)^2/H_0^2 = \Omega_m(1+z)^3 + (1-\Omega_m)$, where $w_\Lambda = -1$, which provide a good fit for a large number of observational data compilations without addressing some important theoretical problems, such as the cosmic coincidence and the fine tuning of the Λ value [39].

- CPL model. So-called by Chevallier-Polarsky-Linder [40, 41], can be represented by two parameters that exhibit the present value of the EoS w_0 and its overall time evolution w_a:

$$w(z)_{\text{CPL}} = w_0 + \left(\frac{z}{1+z}\right) w_a. \tag{2.1}$$

As an extrapolation of these standard scenarios, in the same homogeneous and isotropic universe framework, we consider that the gravitational sector to which the matter sector is minimally coupled, is described by the standard General Relativity. We also assume that the total energy of the universe comes in the form of photons (γ), baryons (b), neutrinos (ν) and a unified dark fluid (UDF, X_U) [42]. This component can behave as dark energy, dark matter or a different type of fluid while the universe expands. Therefore, the full energy budget is denoted by p_i/ρ_i, where $i = \gamma, b, \nu, X_U$. Additionally, each fluid i obeys a continuity equation of the form $\dot{\rho}_i + 3\frac{\dot{a}}{a}(p_i + \rho_i) = 0$. Standard solutions are: $\rho_b \propto a^{-3}$ and $\rho_{\gamma,\nu} \propto a^{-4}$. For a UDF, a constant adiabatic sound speed c_s is assumed and can be modelled as $p = c_s^2(\rho - \tilde{\rho})$, where c_s and $\tilde{\rho}$ are positive constants. One part of this expression behaves as the usual barotropic cosmic fluid and the other as a Λ, which unifies the dark energy and dark matter components (effect so-called as dark degeneracy). By integrating for a UDF we obtain:

$$\rho = \rho_\Lambda + \rho_{X_U} a^{-3(1+c_s^2)}, \tag{2.2}$$

$$p = -\rho_\Lambda + c_s^2 \rho_{X_U} a^{-3(1+c_s^2)}, \tag{2.3}$$

where $\rho_\Lambda = c_s^2 \tilde{\rho}/(1+c_s^2)$ and $\rho_{X_U} = \rho_0 - \rho_\Lambda$, with ρ_0 as the dark energy density at the present time. The dynamical EoS is given by:

$$w = -1 + \frac{1 + c_s^2}{\left(\frac{\rho_\Lambda}{\rho_{X_U}}\right)(1+z)^{-3(1+c_s^2)} + 1}. \tag{2.4}$$

At this point, a specific form which specifies p_{X_U} as a function of ρ_{X_U}: $p_{X_U} = f(\rho_{X_U})$ needs to be considered:

- Generalised Chaplygin gas (GCG) model. For a given X_U is characterised by: $p_{\mathrm{gcg}} = -\frac{A}{(\rho_{\mathrm{gcg}})^\alpha}$, where A and $0 \le \alpha \le 1$ are two free parameters. The case $\alpha = 1$ correspond to the original Chaplygin gas model. Solving the continuity equation using the evolution of the GCG energy density, one gets:

$$\rho_{\mathrm{gcg}}(a) = \rho_{\mathrm{gcg},0} \left[b_s + (1-b_s)a^{-3(1+\alpha)} \right]^{\frac{1}{1+\alpha}}, \qquad (2.5)$$

where $\rho_{\mathrm{gcg},0}$ denotes the energy density of the GCG fluid at present time and $b_s = A\rho_{\mathrm{gcg},0}^{-(1+\alpha)}$. This density contains all the i components described above. Then we can compute the GCG dynamical equation of state w_{gcg} as a function of the redshift z:

$$w_{\mathrm{gcg}}(z) = -\frac{b_s}{b_s + (1-b_s)\left(\frac{1}{1+z}\right)^{-3(1+\alpha)}}. \qquad (2.6)$$

This establish the regions of dominations for an effective dark matter component and an effective dark energy one, with an intermediate region for $\alpha = 1$.

- Modified Chaplygin gas (MCG) model. Its relation between pressure p_{mcg} and energy density ρ_{mcg} is given by: $p_{\mathrm{mcg}} = B\rho_{\mathrm{mcg}} - \frac{A}{(\rho_{\mathrm{mcg}})^\alpha}$, where again A, B, and α are three real constants with $0 \le \alpha \le 1$. If $A = 0$, the MCG behaves as a perfect fluid with $w = B$, whereas, if $B = 0$, we can recover the GCG model. Again, the standard Chaplygin gas model can be obtained by setting $\alpha = 0$. Solving the equation we find that the evolution of the MCG energy density with all its i components:

$$\rho_{\mathrm{mcg}}(a) = \rho_{\mathrm{mcg},0} \left[B_s + (1-B_s)a^{-3(1+B)(1+\alpha)} \right]^{\frac{1}{1+\alpha}}, \qquad (2.7)$$

where $\rho_{\mathrm{mcg},0}$ denotes the energy density of the MCG fluid at present time, and $B_s = A\rho_{\mathrm{gcg},0}^{-(1+\alpha)}/(1+B)$. The evolution of the MCG EoS is given by:

$$w_{\mathrm{mcg}}(z) = B - \frac{B_s(1+B)}{B_s + (1-B_s)\left(\frac{1}{1+z}\right)^{-3(1+B)(1+\alpha)}}. \qquad (2.8)$$

As an extension of the GCG model, the MCG model behaves accordingly with each cosmological regions described.

3 Pantheon Type Ia supernovae compilation

Since we are interested in the behaviour of the cosmic late acceleration using the proposed dark energy models, in this paper we employ the recent Type Ia supernovae (SNeIa) sample called Pantheon [43].

This sampler consists of 1048 SNeIa in 40 bins compressed. It is the largest spectroscopically confirmed SNeIa sample to date. This characteristic makes it attractive to develop with our DL test. Since we are performing tests of EoS's that, at some point, recover ΛCDM, the binned catalog is not a problem in the sense of favoring this model.

SNeIa can give determinations of the distance modulus μ, whose theoretical prediction is related to the luminosity distance d_L according to:

$$\mu(z) = 5 \log \left[\frac{d_L(z)}{1\text{Mpc}} \right] + 25,$$

(3.1)

where the luminosity distance is given in Mpc. In the standard statistical analysis, one adds to the distance modulus the nuisance parameter M, an unknown offset sum of the supernovae absolute magnitude (and other possible systematics), which is degenerate with H_0.

As we are assuming spatial flatness, the luminosity distance is related to the comoving distance D via $d_L(z) = \frac{c}{H_0}(1 + z)D(z)$, where c is the speed of light, so that, we can obtain

$$D(z) = \frac{H_0}{c}(1 + z)^{-1} 10^{\frac{\mu(z)}{5} - 5}.$$

(3.2)

Therefore, the normalised Hubble function $H(z)/H_0$ can be obtained by taking the inverse of the derivative of $D(z)$ with respect to the redshift

$$D(z) = \int_0^z H_0 d\tilde{z}/H(\tilde{z}),$$

(3.3)

where H_0 is the Hubble constant consider as a prior value to normalise $D(z)$.

4   Basics on deep learning

The deep learning technique allows algorithms to learn more complex patterns by adding more layers to the Neural Network (NN), using Convolutional (CNN) or Recurrent Neural Networks (RNN) [44]. DL has demonstrated its versatility in the application of new methods where rich data (as the supernovae Pantheon sampler presented above) will play an important role. Additionally, it allows to automatically built a (usually highly nonlinear) model that maps from a given input (e.g a redshift z) to an output (e.g. a modulus distance $\mu(z)$) with different algorithms to use several prescriptions to build up a specific model.

Can this information be used by the DL algorithm to gain a better understanding of a dark energy cosmological model? Currently, the interest over these kind of algorithms are bringing new opportunities for data-driven cosmological discovery, but they will also present new challenges for adopting DL methodologies and understanding the results when the data are too complex for traditional model development and fitting with statistics. Some proposals in that regards have been done in order to explore the DL methods to measure cosmological parameters for future large-scale photometric surveys [34] and from density fields [45]. Even more, the advantage to use DL is that the overfitting can be avoided, it increases the robustness of classifier and carries a coverage of trained data. In our case, we select the extensive knowledge of supernovae to obtain a DL training data sampler.

In this paper we used, as first piece of the new network proposed, a supervised learning training called Recurrent Neural Network (RNN), which is a type of algorithm that adopts the real target to teach the NN and penalizes when it is far from the real data. Then this algorithm output is compared with the using of a loss function, i.e. we take into account the Mean Squared Error (MSE) function, where the objective of the algorithm is to minimise the loss function. Afterwards, we use the Adam optimizer [46] to find the minimum and obtain the best cosmological model. In order to compute the confidence errors for our cosmological

models, we unify this first piece of architecture with our second piece, a Bayesian Neural Network (BNN). Along the rest of this paper we refer to the union of these architectures as our RNN+BNN network.

In the next two subsections, we describe each architecture separately. Afterwards, we will explain how to join both in order to design our RNN+BNN network.

4.1 Training Recurrent Neural Networks (RNN)

There are many applications of DL for large photometric surveys, such as: (1) the measurement of galaxy shapes from images; (2) automated strong lens identification from multi-band images; (3) automated classification of supernovae; (4) galaxy cluster identification. We will focus on a non-linear regression-like method for supernovae using deep RNN. It was show in [34, 47] that deep RNN are capable to learn complex sequential information to classify supernovae. In our approach, instead of performing feature extraction before classification, we use the redshifts of light-curves directly as inputs to a RNN, which is able to learn information from the sequence of observations. Due to the intrinsic relation of our data with time, we employ RNN that has show a successful behaviour with sequential series, being used in problems like translation or transcription. These kind of networks are similar to Feed Forward Neural Networks (FFNN) [33, 48], but have an essential difference in comparison to RNN since a recurrent network feed itself during the training, e.g. one network with a neuron will have a connection from the input and also from the output of its previous time step — from which feed itself — a characteristic which make them adequate to train a supernovae data sampler. We will explain mathematically this aspect in the next system of equations.

Even more, these connections allow the algorithm to have a contest above all the sequence, alike FFNN where we must show all the sequence at the same time, so, these networks could not identify who is first and who is not. Then, the FFNN only see a gate as a free parameter which it does not feed itself with the latter value, i.e there is not a direct connection with the backward data.

To design a RNN architecture, we start with a cell were the output of the previous time step is used to compute the new one (see figure 1). The same weights are used to compute all the sequence, and each cell is feeding with the output before using the following expression:

$$h^{<t>} = g\left(W_h \cdot h^{<t-1>} + W_x \cdot x^{<t>} + b_a\right), \tag{4.1}$$
$$y^{<t>} = g\left(W_y \cdot h^{<t>} + b_y\right), \tag{4.2}$$

where $h^{<t>}$ is called the hidden state and its $h^{<t-1>}$ value represent the hidden state before it, g is the activation function and $y^{<t>}$ is the output. Here, t is the time referring to the first sequence ($t = 1$, for the first redshift) with the data and b are the bias. From now on, the superscript $<>$ indicates a vector and the dot is a matrix product.

However, basic RNN fails in large sequences due to the loss of information from the initial inputs. In order to improve our training performance we use a modification of RNN cells, the so-called Long Short Term Memory (LSTM) cells [33]. The goal of these kind of cells are their capability of forgetting and adding information step by step. However, a lot of modifications have been done to these cells by incrementing the number of matrix functions

for each layer which correspond to [49]:

$$i^{<t>} = \sigma \left(W_{xi}^T \cdot x^{<t>} + W_{hi}^T \cdot h^{<t-1>} + b_i \right), \tag{4.3}$$

$$f^{<t>} = \sigma \left(W_{xf}^T \cdot x^{<t>} + W_{hf}^T \cdot h^{<t-1>} + b_f \right), \tag{4.4}$$

$$o^{<t>} = \sigma \left(W_{xo}^T \cdot x^{<t>} + W_{ho}^T \cdot h^{<t-1>} + b_o \right), \tag{4.5}$$

$$g^{<t>} = A_f \left(W_{xg}^T \cdot x^{<t>} + W_{hg}^T \cdot h^{<t-1>} + b_g \right), \tag{4.6}$$

$$c^{<t>} = f^{<t>} \otimes c^{<t-1>} + i^{<t>} \otimes g^{<t>}, \tag{4.7}$$

$$y^{<t>} = h^{<t>} = o^{<t>} \otimes A_f(c^{<t>}), \tag{4.8}$$

where W are the weights of each layer, σ is the sigmoid function that takes values between 0 and 1. Here $\otimes$ is the direct product. In the rest of the paper, we denote the superscript T as the transpose of the quantity where it is indicated. Notice how the feed process is explicitly given in the second r.h.s. term of the eqs. (4.3)–(4.6).

We compare several trainings for $\mu(z)$ using four kind of activation functions A_f defined as[1][50]:

$$A_{f_{\mathrm{Tanh}}} = \tanh(x), \quad \text{in} \quad (-1,1), \tag{4.9}$$

$$A_{f_{\mathrm{ReLU}}} = \begin{cases} 0 \text{ for } x \leq 0, \\ x \text{ for } x > 0. \end{cases} \quad \text{in} \quad [0,\infty), \tag{4.10}$$

$$A_{f_{\mathrm{ELU}}} = \begin{cases} \alpha(e^x - 1) \text{ for } x \leq 0, \\ x \qquad\qquad \text{ for } x > 0, \end{cases} \text{in} \quad (-\alpha,\infty), \tag{4.11}$$

$$A_{f_{\mathrm{SELU}}} = \begin{cases} \alpha\lambda(e^x - 1) \text{ for } x \leq 0, \\ x \qquad\qquad \text{ for } x > 0, \end{cases} \text{in} \quad (-\alpha\lambda,\infty). \tag{4.12}$$

In this full system of equations, we consider three layers called *gate controllers*: (a) *Input gate:* this gate is giving by (4.3) and helps the cell to know which values must remember in a long term. (b) *Forget gate:* giving by (4.4), this gate allows the cells to forget certain values. (c) *Output gate:* it controls which values will be the output and which not, as it is seen in (4.5). Eq. (4.6) gives a possible new state to remember in a long term. However, according to the LSTM, the final state (4.7) and (4.8) give the output of the cell, e.g if we obtain a null vector for the forget gate, taking into account (4.7), the network will not remember anything about the previous state. Whereas, if we obtain values of one, it will remember everything.

Nevertheless, NN have a large number of parameters, which can cause a high possibility of overfitting. In addition of these technical parameters, we have at hand the free parameters from each cosmological models. Therefore, we improve this by using a *regularization* technique [48]. But, in RNN few regularization models works well and a good suggestion is the *Variational Dropout* (VD) [51]. *Dropout* is a powerful and inexpensive method of regularization, which randomly turns off some neurons to avoid overfitting and the probability of turning off a neuron is given by the hyperparameters. These can be adjusted depending on size, epochs, layers, neurons and batchsize adopted. In the VD, we repeat the same dropout mask for inputs, hidden states and outputs. This method allows a good behaviour for dropout in RNN and is very useful in BNN to compute the corresponding confidence contours for our trainings.

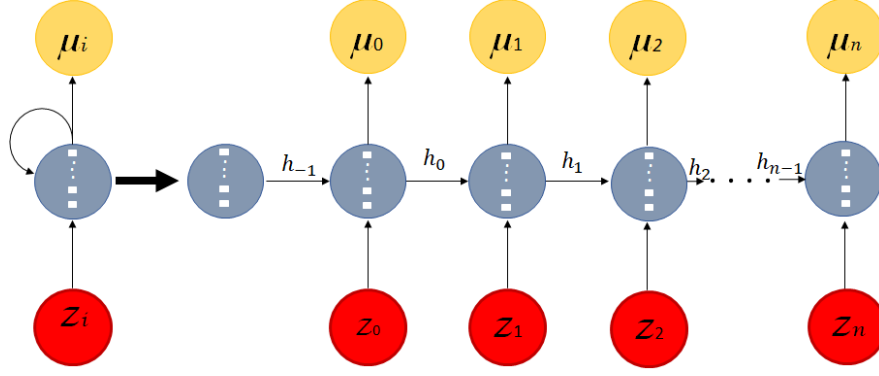[1]We consider here the values for $\alpha = 1.673$ and $\lambda = 1.051$ for a monotic behaviour.

Figure 1. RNN architecture. The input z_i are the redshifts and the output y_i are the corresponding modulus distance μ_i. This network example has one layer, and the small white squares inside each circle node represent the number of neurons h_i. In our proposal the number of neurons is equal to 100. Each neuron receives inputs from other neurons through the paths. After several steps there is some processing on the input and the calculated result is passed to other neurons connected through another path. Each node is associated to an activation function (Tanh, ReLU, ELU and SELU) which transforms the input to an output value. The output value of these functions acts as an input to the next connected neurons. Final output helps in deciding the class of input data. The number of steps is given by n.

4.2 Bayesian Neural Networks (BNN)

They emerge as a solution for problems in DL like overfitting and the incapability of networks to calculate their uncertainty. Uncertainties are crucial to evaluate the validity of a reliable result. So far, for astrophysical scenarios, the error calculations are misleading due the deterministic performing of the networks and at this point they are unable to provide a good estimation of the error.

Therefore, we propose a combination of RNN and bayesian approaches by BNN that can solve this problem by making the networks probabilistic. We positioned them in order to deal with the uncertainty issue and take all assumptions into account, i.e. we can obtain a probability distribution over the weights, given the distributional training data, to be used as next inputs. At the end of the process, we are capable to obtain an entire distribution over the network output, which increases our predicting accuracy and confidence regions. For this purpose, we need to know what kind of distribution follows the weights. Having a prior distribution on their weights and their biases, and giving a dataset $\mathbf{X}, \mathbf{Y}$, we search for the posterior distribution over the space of parameters: $p(\omega|X, Y)$. Hence, we can predict an output for a new input point x by integrating the following:

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{Y}) = \int p(\mathbf{y}^*|\mathbf{x}^*, \omega)p(\omega|\mathbf{X}, \mathbf{Y})\mathbf{d}\omega. \tag{4.13}$$

This *dropout* could be used in model uncertainty estimation and it is an approximation of the Gaussian process. In this way, we can calculate model uncertainty applying dropout in the training n-times. We consider a BNN modelled with a simple dropout which offers a same mask at each time step leading to improved results and without overfit quickly [51, 52].

5 RNN+BNN architecture for deep learning dark energy EoS

In this section we describe the methodology to train the observational data given by the Pantheon supernovae sample in section 3 using our proposed RNN+BNN network. We analise four different architectures of activation functions. The implementation of these analyses were done using TENSORFLOW [32]. All of them were built out of layers shown in figure 1. The first layer of the network is a RNN with one activation function. The last layer is a connected layer with no activation function to map the output value of the last residual neuron to the desired output size. The value of the size needs to be fixed at hand in order to reduce the dispersion of the data at low redshifts while the NN adjusts during the training. The control of this value can be achieved by computing the χ^2 for each activation function. The RNN+BNN network was trained on two cosmological parameters: an input z and the output μ. The advantage of this process if that we can add a BNN to the architecture to always predict a valid covariance matrix. Finally, we used a negative log-likelihood loss function

$$L = \frac{1}{2} \left(\ln \zeta + (\theta_p - \theta_t)^T \zeta^{-1} (\theta_p - \theta_t) \right), \tag{5.1}$$

where θ_p if the vector of predicted parameters, ζ^{-1} is the predicted inverse covariance matrix given by BNN and θ_t are the input (real) parameters. It is important to remark that at this first stage of the DL technique we can only use data that share the same cosmological observations, e.g. luminosity (modulus) distance. While the results of several data samplers are already statistically promising, we found that combining the prediction of different datasets have high noise levels which make difficult the convergence of our RNN+BNN network.

In order to construct and train our network, we carry out the following algorithm:

(i) Construction of the neural network. For the RNN, our model uses the LSTM cell described in section 4.1. We consider one layer with 100 neurons and use a projector to obtain the same shape of the input. The input is z and the output μ. We construct the sequence from lower to higher redshift z.

(ii) Organising the cosmological data. After ordering from lower to higher redshift z, we need to re-arrange the data using the number of steps n. According to our architecture in figure 1, we choose a number of steps $n = 4$.

(iii) Computing the BNN. Our dropout has the following parameter: the probability to drop the input is 0. This is because, after testing it several times, we found that our models could not be trained with an input dropout due the lost of information. Alternatively, the state dropout is 0.4 and the output is 0.5 [33]. Finally, the cost function is MSE type, therefore we can use the Adam optimizer.

(iv) Training the RNN+BNN neural network. We start the training by consider 1000 epochs. Afterwards, we read the model and apply 500 times the same dropout to our initial model. This allows to obtain the uncertainty contours (figure 2). After the training, we predicted a data sampler of 1000 data points.

(v) Calculation of the bestfits. We use the CLASS[2] and Monte Python codes[3] to constrain the models using the new Pantheon sampler obtained from the RNN+BNN training (Step v). For each theoretical model proposed we obtained the values reported in tables 8, 9, 10, 11.

---

[2]https://github.com/lesgourg/class_public.
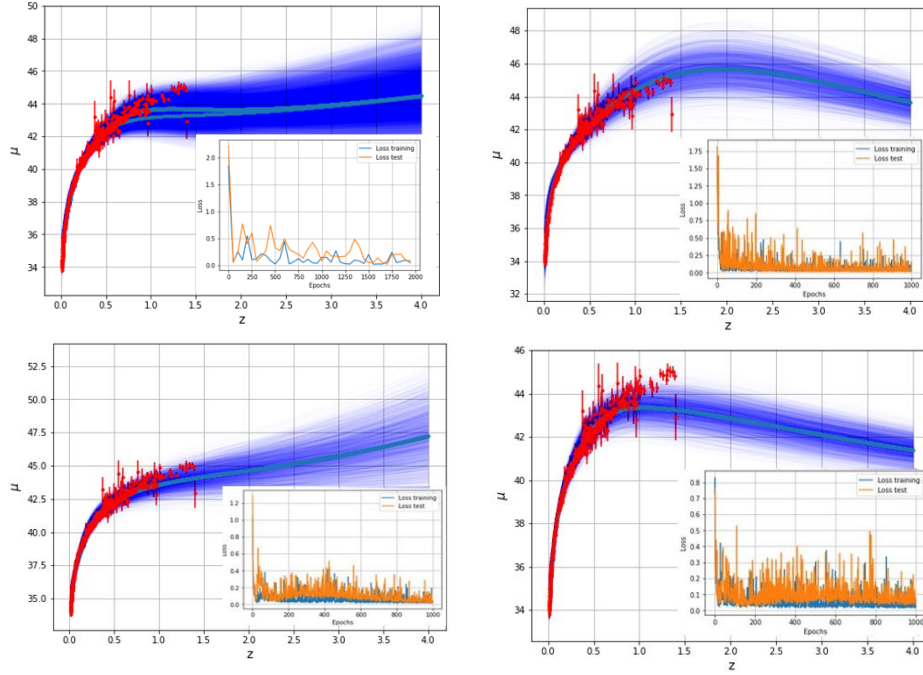[3]https://github.com/baudren/montepython_public.

Figure 2. RNN-BNN network training models for Pantheon sampler using activation functions (4.9)–(4.12) from top (left) to bottom (right), respectively. *Outside plot:* the uncertainty contours at 1–2σ are represented by blue color, the green line represent the mean data trained and the red color dots are the real data. *Inside plot: Loss* plots were also created to show the mean absolute error loss over the training epochs for both the train (blue color) and test (orange color) sets.

(vi) Calculation of $\mu(z)$. For each of our cosmological models, we consider now 2000 epochs to redo (iv). Once we obtain the new predicted sampler, we perform the integration in (3.3) to compute the functions $\mu(z)$ and $E^2(z)$ by using a specific dark energy EoS in terms of z. The results for each model are given in figure 3.

Given the stochastic nature of our training algorithm for Pantheon sampler, some specific results may vary. We run our chain up to 100 times in order to obtain a homogeneous result and to extrapolate the redshift up to $z = 4$, given as a result that the model learned the problem, achieving a near zero error, at least to two decimal places. In figure 2, it is worth noticing that for each activation function the training models seems to have converged. The loss plots show good convergence behaviour, although somewhat bumpy. The model may be good configured given no sign of over or under fitting and the learning rate or batch size may be tuned to even out the smoothness of the convergence in this case. We change the activation functions to improve our performance and reduce uncertainty contours. As it is expected, uncertainty contours change with the activation functions due the change on the covariance matrix. As a main result, the activation function (4.9) gives the better evolution to perform our dark energy trainings. This result will be probed from a bayesian point of view in the next section. According to the trainings (figure 3), we notice that unified dark fluids models seems to be preferred at large redshift less than 1-σ in comparison to the standard models.

The models in figure 2 are in agreement of what it is expected of the activation functions: ReLU is not implemented in models with one layer and neither in LSTM cells since their unbounded values could easily explode. LSTM cells commonly use the sigmoid and Tanh
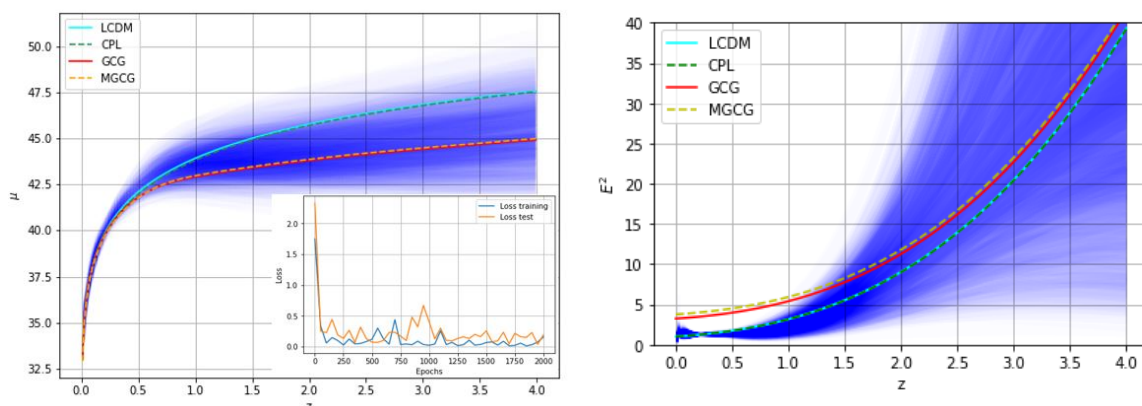
Figure 3. *Left:* trainings for dark energy EoS: ΛCDM model (blue line), CPL model (green-dashed line), GCG model (red line) and MGCG (yellow-dashed line) using the activation function (4.9). *Right:* trainings for $E^2(z)$ for each model.

activation functions. Our results support the use of this last activation function in these types of cells.

Our proposed RNN+BNN network algorithm for SNeIa is composed by 8 hyperparameters:

- Size=4, Epochs=1000, Layers=1, Neurons=100, Bathsize=10.

- A variational dropout composed by: an input z, a hidden state h and an output $\mu(z)$.

As a first tentative exploration, these DL parameters can be interpreted as the price we pay in order to deal with a model-independent dark energy cosmology. Moreover, a homogeneous sampler as SNeIa is not a complex model,[4] therefore it only needs one layer as much, since we do not require greater depth in the model. If we add more layers the computational cost is unnecessary and can lead us to overfitting. Also, we notice that the number of neurons is high, but it is important to remark that the dropout controls any overfitting that this can carried out. Each of the latter hyperparameters can add complexity to the model, which can lead us to overfitting (or underfitting), but none of them — at least for this homogeneous SNeIa sampler) — can change drastically the covariance matrices, as the A_f can indeed do it [51].

6 Bayesian versus deep learning significance

In this section we present a Bayesian model selection to describe the relationship between the cosmological models described above and their DL trainings, the supernovae data and the prior information about the free parameters. Using Bayes theorem [53] we can updated the prior model probability to the posterior model probability. Additionally, when we compare models, the evidence can be used to evaluate the model's evolution using the sampler available. The evidence can be denoted by

$$\mathcal{E} = \int \mathcal{L}(\theta)P(\theta)d\theta, \tag{6.1}$$

---

[4]More complex models, e.g natural language, requires thousands of inhomogeneous data.

where θ is the vector of free cosmological parameters for each model and $P(\theta)$ is the prior distribution of these parameters. The evidence can be computed with several methods [54, 55], but for our purpose we applied a multinested sampling algorithm [56] which has proven practicable in cosmology applications [57].

6.1 Information criteria results

Since we are working with four cosmological models with different number of cosmological parameters, the simplest procedure to compare them is the likelihood ratio test, which can be applied when the simple model is nested within a more complex model [59]. The quantity $2 \ln \mathcal{L}_{\text{simple}} / \mathcal{L}_{\text{complex}}$, where $\mathcal{L}$ is the maximum likelihood of the model under consideration, is approximately chi-squared distributed and Jeffreys's scale can be used to look up the significance of any increase in likelihood against the number of extra parameters introduced. Also, due the number of free cosmological parameters and data sampler, we consider two statistical criterias in order to compare our DL architecture results (tables 2, 3, 4, 5) with the observational ones (table 1).

The Akaike information criterion (AIC) is defined as

$$\text{AIC} = -2 \ln \mathcal{L} + 2k \,, \tag{6.2}$$

where $\mathcal{L}$ is the maximum likelihood and k the number of parameters of the model. The best model is the model which minimizes the AIC. Usually models with few parameters give a poor fit to the data and hence have a low log-likelihood, while those with many are penalized by the second term in (6.2).

The Bayesian information criterion (BIC) is defined as

$$\text{BIC} = -2 \ln \mathcal{L} + k \ln N \,, \tag{6.3}$$

where N is the number of data points used in the fit.

6.2 Evidence criteria results

We compute the logarithm of the Bayes factor between two models $\mathcal{B}_{ij} = \mathcal{E}_i / \mathcal{E}_j$, where the reference model ($\mathcal{E}_i$) with highest evidence is the ΛCDM model. To compare between ΛCDM and the other models we use Jeffreys's scale [58]: if $\ln B_{ij} < 1$ there is not significant preference for the model with the highest evidence; if $1 < \ln B_{ij} < 2.5$ the preference is substantial; if $2.5 < \ln B_{ij} < 5$ it is strong; if $\ln B_{ij} > 5$ it is decisive.

The results of these analyses are reported in table 6. We compute AIC and BIC statistical criteria for each activation function and cosmological model. Each value is compared with the pivot ΛCDM model using the trained RNN+BNN. From here we notice that the activation function Tanh is statistically better in comparison to the others. Also, in order to set a conclusive result in regards to this activation function we design a Bayesian portrait (see figure 4) according to the number of free EoS parameters using the values given in table 6. The qualitative color regions are detailed according to the Jeffreys's scale (table 7). We denote the number of free cosmological parameters for each model in the y-axes and the evidence in x-axes. From figure 4 the yellow region indicates that if the models fall there, it means that the trainings used for these models have a statistical preference over ΛCDM. All our models are in this region. The more advanced towards the yellow region are the models, a strongest evidence against ΛCDM appears. Notice that the MCG model using a Tanh activation function is the best of all. ReLU and Tanh functions compete with each other

Model	Parameterisation	$\chi^2_{\min}$	$-\ln\mathcal{L}_{\min}$
ΛCDM	$w=-1$	1027	513.430
CPL	$w(z)=w_0+\left(\frac{z}{1+z}\right)w_a.$	1028	514.237
GCG	$w(z)=-\dfrac{b_s}{b_s+(1-b_s)\left(\frac{1}{1+z}\right)^{-3(1+\alpha)}}$	1050	525.085
MCG	$w(z)=B-\dfrac{B_s(1+B)}{B_s+(1-B_s)\left(\frac{1}{1+z}\right)^{-3(1+B)(1+\alpha_0)}}$	1028	514.248

Table 1. Results for the dark energy parameterisations with best fits parameter using observational Pantheon sampler. In 3-column it is show the $\chi^2_{\min}$ for each model and in 4-column the likelihood for each model.

Model	Activation function	$\chi^2_{\min}$
ΛCDM	$A_{f_{\text{Tanh}}}=\tanh(x)$	606
	$A_{f_{\text{ReLU}}}=\begin{cases}0 \text{ for } x\leq 0,\\ x \text{ for } x>0.\end{cases}$	647
	$A_{f_{\text{ELU}}}=\begin{cases}\alpha(e^x-1) \text{ for } x\leq 0,\\ x \qquad\;\; \text{ for } x>0,\end{cases}$	570
	$A_{f_{\text{SELU}}}=\begin{cases}\alpha\lambda(e^x-1) \text{ for } x\leq 0,\\ x \qquad\;\; \text{ for } x>0,\end{cases}$	408

Table 2. χ^2 analyses for ΛCDM using the activations functions (4.9), (4.10), (4.11) and (4.12).

Model	Activation function	$\chi^2_{\min}$
CPL	$A_{f_{\text{Tanh}}}=\tanh(x)$	932
	$A_{f_{\text{ReLU}}}=\begin{cases}0 \text{ for } x\leq 0,\\ x \text{ for } x>0.\end{cases}$	971
	$A_{f_{\text{ELU}}}=\begin{cases}\alpha(e^x-1) \text{ for } x\leq 0,\\ x \qquad\;\; \text{ for } x>0,\end{cases}$	741
	$A_{f_{\text{SELU}}}=\begin{cases}\alpha\lambda(e^x-1) \text{ for } x\leq 0,\\ x \qquad\;\; \text{ for } x>0,\end{cases}$	559

Table 3. χ^2 analyses for CPL using the activations functions (4.9), (4.10), (4.11) and (4.12).

(the blue and red circles overlap for CPL model and GCG model), but for MCG model with a ReLU activation function shows a better evidence in comparison to ELU function. SELU is definitely far behind in this scale.

Model	Activation function	$\chi^2_{\min}$
GCG	$A_{f_{\text{Tanh}}} = \tanh(x)$	1314
	$A_{f_{\text{ReLU}}} = \begin{cases} 0 \text{ for } x \leq 0, \\ x \text{ for } x > 0. \end{cases}$	1367
	$A_{f_{\text{ELU}}} = \begin{cases} \alpha(e^x - 1) \text{ for } x \leq 0, \\ x \qquad\quad \text{ for } x > 0, \end{cases}$	1055
	$A_{f_{\text{SELU}}} = \begin{cases} \alpha\lambda(e^x - 1) \text{ for } x \leq 0, \\ x \qquad\quad\;\; \text{ for } x > 0, \end{cases}$	753

Table 4. χ^2 analyses for GCG using the activations functions (4.9), (4.10), (4.11) and (4.12).

Model	Activation function	$\chi^2_{\min}$
MCG	$A_{f_{\text{Tanh}}} = \tanh(x)$	4243
	$A_{f_{\text{ReLU}}} = \begin{cases} 0 \text{ for } x \leq 0, \\ x \text{ for } x > 0. \end{cases}$	1651
	$A_{f_{\text{ELU}}} = \begin{cases} \alpha(e^x - 1) \text{ for } x \leq 0, \\ x \qquad\quad \text{ for } x > 0, \end{cases}$	1180
	$A_{f_{\text{SELU}}} = \begin{cases} \alpha\lambda(e^x - 1) \text{ for } x \leq 0, \\ x \qquad\quad\;\; \text{ for } x > 0, \end{cases}$	922

Table 5. χ^2 analyses for MCG using the activations functions (4.9), (4.10), (4.11) and (4.12).

Model	$A_{f_{\text{Tanh}}}$			$A_{f_{\text{ReLU}}}$			$A_{f_{\text{ELU}}}$			$A_{f_{\text{SELU}}}$		
	$\ln B_{ij}$	AIC	BIC	$\ln B_{ij}$	AIC	BIC	$\ln B_{ij}$	AIC	BIC	$\ln B_{ij}$	AIC	BIC
CPL	5.825	957.29	940	5.819	996.29	979	5.213	766.29	749	5.098	584.29	567
GCG	6.580	1339.29	1322	6.597	1392.29	1375	6.209	1080.29	1063	5.879	778.29	761
MCG	8.204	1682.61	1661	6.930	4274.61	4253	6.444	1211.61	1190	6.278	953.61	932

Table 6. Bayesian Deep Learning Pantheon results for the dark energy parameterisations using the activations functions (4.9), (4.10), (4.11) and (4.12). In each the subcolumn it is show the Bayes factor $\ln B_{ij}$, AIC (6.2) and BIC (6.3), respectively. Each value is compared with the pivot ΛCDM model using the trained RNN+BNN.

$\ln B_{i0}$	Strength of evidence	Color region
> 5	Strong evidence for model i	Yellow
$[2.5, 5]$	Moderate evidence for model i	Red
$[1, 2.5]$	Weak evidence for model i	Blue
$[-1, 1]$	Inconclusive	Green

Table 7. Jeffreys's scale. The Bayes factor of CPL, GCG and MCG parameterisation were computed in comparison to the model 0, ΛCDM. We used the information from table 6.
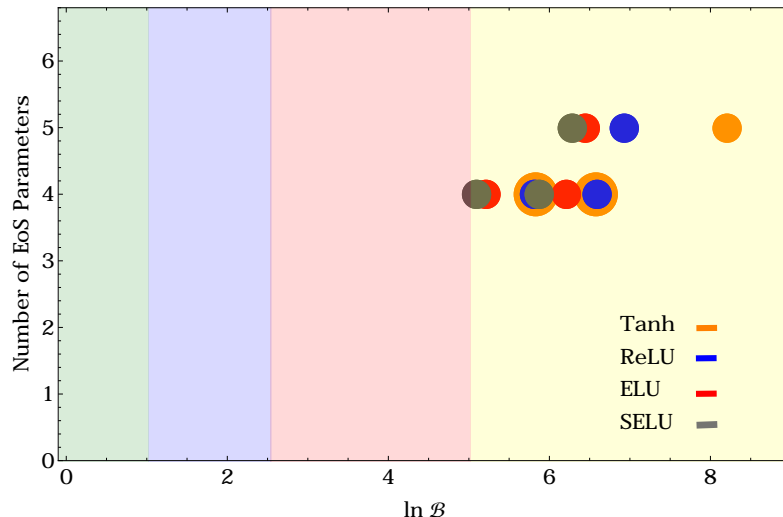
Figure 4. Bayesian portrait according to the number of free EoS parameters using the values given in table 6. The qualitative color regions are detailed in table 7. Each color in the plot legend indicates the activation function used to calculated the value $\ln B_{ij}$ in comparison to ΛCDM model. The MCG model is represented at level $\nu = 5$ and CPL and GCG model at $\nu = 4$.

6.3 Statistical predictions using RNN+BNN network results

In this section we present the parameter spaces using the Tanh activation function eq. (4.9) according the results obtained in the latter sections. We proceed to compute the confidence regions for the models described in section 2 using our deep learning Pantheon sampler. In figures 5, 6, 7, 8 are reported the confidence contours and posteriors for each cosmological model, respectively. Also we reported in tables 8, 9, 10, 11 the best fit parameters within 95% C.L. The results obtained from these analyses show a better χ^2 using a deep learning trained sampler than about 1 unit in comparison to the pivot cosmological model. Furthermore, the tension at low redshifts seems to be reduced using this new sampler, alike the observational sampler, where a significant tension emerges at these z scales.

On one other hand, model uncertainty is indispensable for the RNN. With model confidence at hand we can treat inputs and specific cases explicitly, e.g. in a scenario of classification a model can return a result with high uncertainty. In our RNN+BNN we used a specific Monte Carlo dropout which is supported by the Bayesian evidence. As an extension, several dropouts have been used in the literature [60] in order to obtain the predicted values for the network. Some of them have insensible values for the observational data, and the others have the same inconvenient but with the additional information that the models are uncertain about their predictions. In our case, with the Bayesian evidence developed, our selected dropout do not affect directly the form of the confidence contours in comparison to the activation function A_f. Each proposed A_f can indeed change these confidence regions since they directly change their covariance matrix [51]. Notice that our results using evidence are aligned with the reported in [60] with different dropouts.

On the other hand, a strong case of support for our RNN+BNN network is the following: notice that our selected $A_{f_{\text{Tanh}}}$ is bounded in comparison to the rest of activation functions. This is a hint to follow a physical interpretation over the trend of an observable: since we are using a sampler of SNeIa, we can expect that the observed differences in peak luminosities
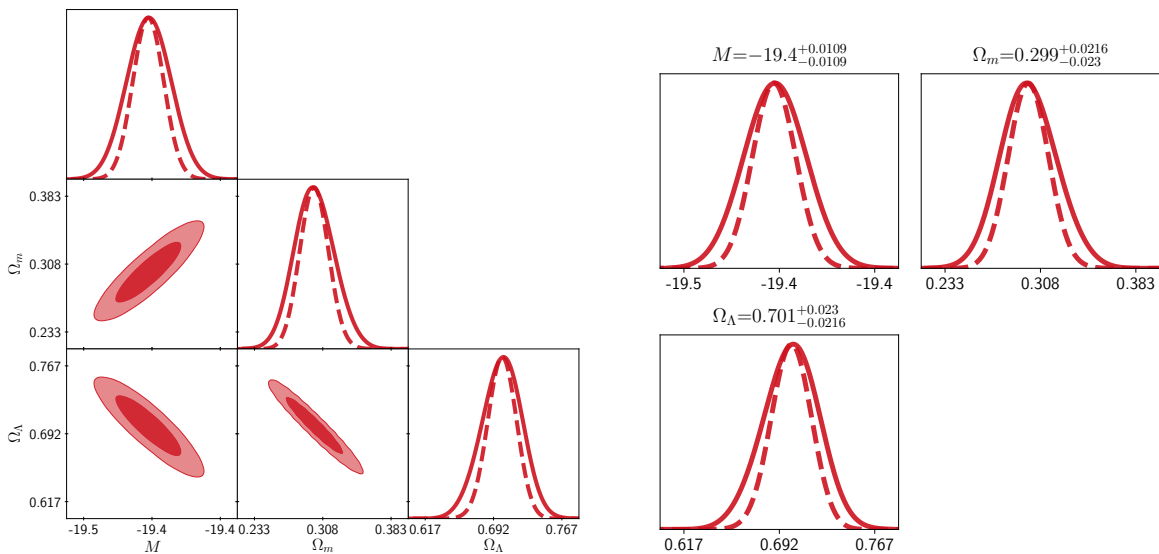
Figure 5. *Left:* confidence contours, *Right:* posterior, for ΛCDM model using Deep Learning Pantheon sampler.

Parameters	Best-fit	mean$\pm\sigma$	95% lower	95% upper
M	-19.43	$-19.43^{+0.011}_{-0.011}$	-19.45	-19.4
Ω_m	0.298	$0.299^{+0.022}_{-0.023}$	0.255	0.344
Ω_Λ	0.702	$0.701^{+0.023}_{-0.022}$	0.656	0.745

Table 8. Best fits values for ΛCDM model using Deep Learning Pantheon sampler.

of SNeIa are very closely correlated with observed differences in the shapes of their light curves, i.e dimmer SNeIa decline more rapidly after maximum brightness, while brighter SNeIa decline more slowly. This behaviour is not given for ReLU and SELU, which makes them non-physical options of activation functions for our training according to the SNeIa trend. Moreover, Tanh and ELU can have a physical trend with the SNeIa, but ELU exhibit a higher error and weak evidence in comparison to Tanh.

Notice that AIC is an estimate of a constant plus the relative distance between the unknown true likelihood function of the trained sampler and the fitted likelihood function of the cosmological model, so lower AIC means a model is considered to be closer to the truth. And the BIC is an estimate of a function of the posterior probability of a model being true, under a certain Bayesian setup, so that a lower BIC means that a model is considered to be more likely to be the true model. According to these ideas and our information criteria analyses, presented in table 6, SELU and ELU gives the lower AIC and BIC values in comparison to Tanh and ReLU. But, since SELU and ELU does not have a physical trend for our observable sample consider, as we explained above, Tanh is the only activation function with a better penalized-likelihood criteria.
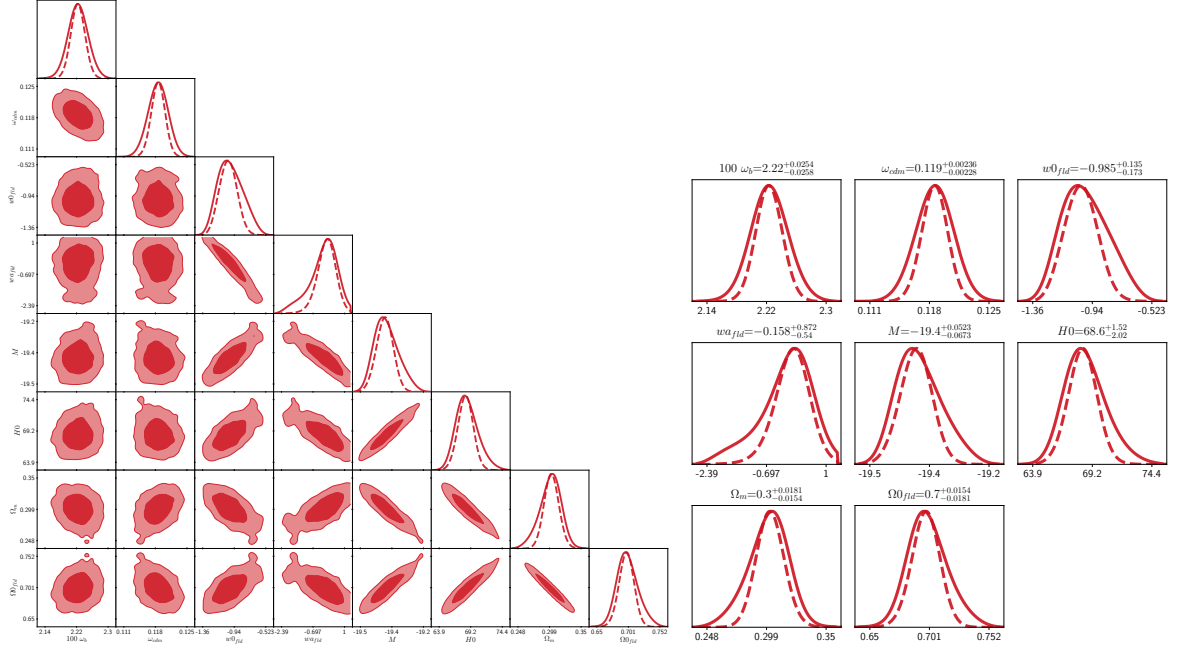
Figure 6. *Left:* confidence contours, *Right:* posterior, for CPL model using Deep Learning Pantheon sampler.

Parameters	Best-fit	mean$\pm\sigma$	95% lower	95% upper
$100\ \omega_b$	2.219	$2.223^{+0.025}_{-0.026}$	2.172	2.275
ω_{cdm}	0.119	$0.119^{+0.002}_{-0.002}$	0.114	0.123
w_0	-1.024	$-0.985^{+0.14}_{-0.17}$	-1.284	-0.641
w_a	0.064	$-0.158^{+0.87}_{-0.54}$	-0.003	0.015
M	-19.4	$-19.39^{+0.052}_{-0.067}$	-19.52	-19.26
H_0	68.41	$68.64^{+1.5}_{-2}$	65.03	72.47
Ω_m	0.301	$0.300^{+0.018}_{-0.015}$	0.265	0.334
Ω_0	0.699	$0.700^{+0.015}_{-0.018}$	0.667	0.735

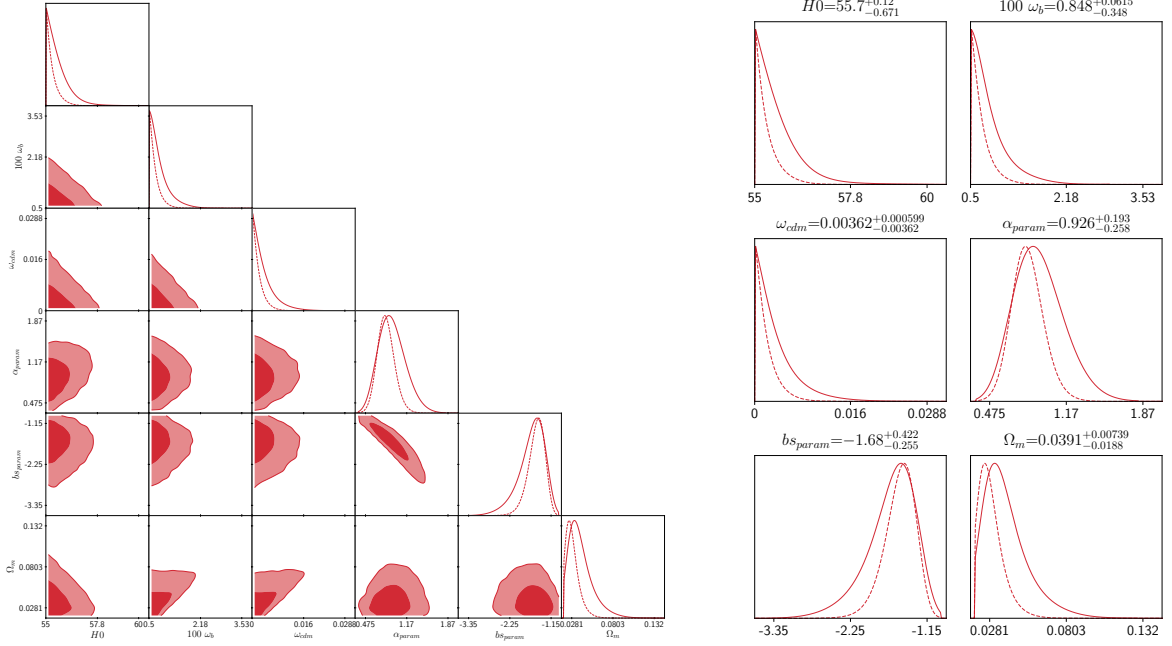**Table 9**. Best fits values for CPL model using Deep Learning Pantheon sampler.

Figure 7. *Left:* confidence contours, *Right:* posterior, for Generalised Chaplygin Gas (GCG) model using Deep Learning Pantheon sampler.

Parameters	Best-fit	mean$\pm\sigma$	95% lower	95% upper
$H0$	55.07	$55.67^{+0.12}_{-0.67}$	55	56.96
$100\ \omega_b$	0.540	$0.848^{+0.061}_{-0.35}$	0.5	1.548
ω_{cdm}	0.001	$0.004^{+0.001}_{-0.004}$	1.499×10^{-7}	0.011
α	0.7628	$0.926^{+0.19}_{-0.26}$	0.4753	1.396
b_s	-1.494	$-1.676^{+0.42}_{-0.26}$	-2.429	-1.012
Ω_m	0.018	$0.039^{+0.007}_{-0.019}$	0.016	0.071

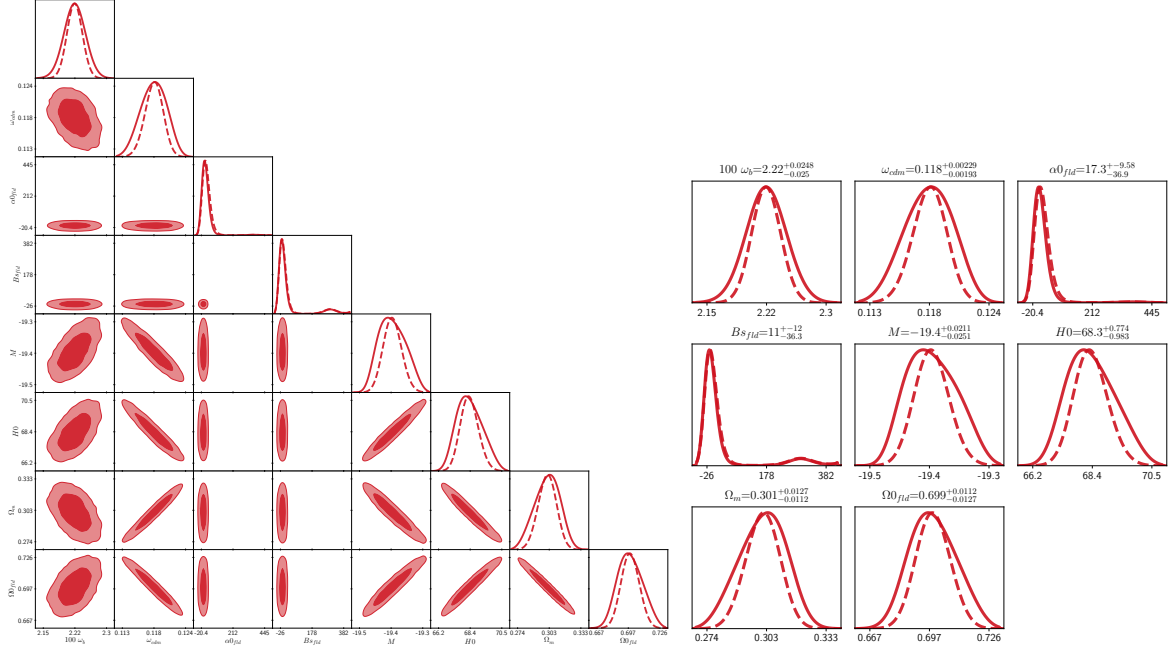**Table 10**. Best fits values for GCG model using Deep Learning Pantheon sampler.

Figure 8. *Left:* confidence contours, *Right:* posterior, for Modified Chaplygin Gas (MCG) model using Deep Learning Pantheon sampler.

Parameters	Best-fit	mean$\pm\sigma$	95% lower	95% upper
$100\ \omega_b$	2.224	$2.224^{+0.025}_{-0.025}$	2.175	2.273
ω_{cdm}	0.118	$0.118^{+0.002}_{-0.002}$	0.114	0.122
α_0	1.089	$17.35^{+-9.6}_{-37}$	0.002	0.002
B_s	-1.901	10.96^{+-12}_{-36}	-0.013	-0.013
M	-19.4	$-19.4^{+0.021}_{-0.025}$	-19.44	-19.35
H_0	68.35	$68.34^{+0.77}_{-0.98}$	66.71	70.14
Ω_m	0.301	$0.301^{+0.013}_{-0.011}$	0.277	0.323
Ω_0	0.699	$0.699^{+0.011}_{-0.013}$	0.677	0.722

Table 11. Best fits values for MCG model using Deep Learning Pantheon sampler.

7 Conclusions

In this work we have trained a new neural network to parameterise efficiently dark energy EoS coming from different models: specifically we have trained the standard ΛCDM, CPL and the unified dark fluids with Chaplygin gas characteristics. With these kind of parameterisations and their neural training, we reproduced the cosmic accelerated expansion currently observed. This novel technique offers the combination of two architectures: (1) Recurrent Neural Networks (RNN) and (2) Bayesian Neural Networks (BNN), which we called BNN+RNN neural network. In comparison to other proposals reported in the DL literature [52], with our proposed network we can train the observational data (which only consist on one input type z and one output type $\mu(z)$), obtain trained data and compute their propagation error with a Bayesian architecture and the way we perform the network allows to the algorithm to be computational faster. Also, a direct physical trend related to the activation functions can be followed in the case of homogeneous sampler as SNeIa. In these kind of processes, the quality in the density of data points is important. In such case, our choice of A_f can be viewed as a *choice of a prior* since it is relevant in regions where the training data is sparse (or even non-existent) and the resulting trained DL sampler explicitly depends of this choice. Therefore, the described scheme is tested with SNeIa data available today. Finally, our new method has a wide range of applicability to many problems in cosmology, for example to explore new constrains on cosmographic parameters. This latter is a working project currently in progress.

Acknowledgments

References

[1] Supernova Search Team collaboration, *Observational evidence from supernovae for an accelerating universe and a cosmological constant*, Astron. J. **116** (1998) 1009 [astro-ph/9805201] [inSPIRE].

[2] Supernova Cosmology Project collaboration, *Measurements of Ω and Λ from 42 high redshift supernovae*, Astrophys. J. **517** (1999) 565 [astro-ph/9812133] [inSPIRE].

[3] http://desi.lbl.gov/.

[4] https://www.darkenergysurvey.org/.

[5] https://www.lsst.org/.

[6] https://wfirst.gsfc.nasa.gov/.

[7] M. Takada and B. Jain, *The three-point correlation function in cosmology*, Mon. Not. Roy. Astron. Soc. **340** (2003) 580 [astro-ph/0209167] [inSPIRE].

[8] WiggleZ collaboration, *The WiggleZ Dark Energy Survey: constraining galaxy bias and cosmic growth with 3-point correlation functions*, Mon. Not. Roy. Astron. Soc. **432** (2013) 2654 [arXiv:1303.6644] [inSPIRE].

[9] S. Tsujikawa, *Dark energy: investigation and modeling*, arXiv:1004.1493 [ɪɴSPIRE].

[10] D. Huterer and M.S. Turner, *Probing the dark energy: Methods and strategies*, *Phys. Rev.* **D 64** (2001) 123527 [astro-ph/0012510] [ɪɴSPIRE].

[11] Y. Wang, *Figure of Merit for Dark Energy Constraints from Current Observational Data*, *Phys. Rev.* **D 77** (2008) 123525 [arXiv:0803.4295] [ɪɴSPIRE].

[12] C. Escamilla-Rivera, R. Lazkoz, V. Salzano and I. Sendra, *Tension between SN and BAO: current status and future forecasts*, *JCAP* **09** (2011) 003 [arXiv:1103.2386] [ɪɴSPIRE].

[13] P. Bull et al., *Beyond ΛCDM: Problems, solutions and the road ahead*, *Phys. Dark Univ.* **12** (2016) 56 [arXiv:1512.05356] [ɪɴSPIRE].

[14] PLANCK collaboration, *Planck 2018 results. VI. Cosmological parameters*, arXiv:1807.06209 [ɪɴSPIRE].

[15] L. Verde, T. Treu and A.G. Riess, *Tensions between the Early and the Late Universe*, in *Nature Astronomy 2019*, 2019, DOI [arXiv:1907.10625] [ɪɴSPIRE].

[16] B. Ratra and P.J.E. Peebles, *Cosmological Consequences of a Rolling Homogeneous Scalar Field*, *Phys. Rev.* **D 37** (1988) 3406 [ɪɴSPIRE].

[17] C. Armendariz-Picon, V.F. Mukhanov and P.J. Steinhardt, *A Dynamical solution to the problem of a small cosmological constant and late time cosmic acceleration*, *Phys. Rev. Lett.* **85** (2000) 4438 [astro-ph/0004134] [ɪɴSPIRE].

[18] I. Sendra and R. Lazkoz, *SN and BAO constraints on (new) polynomial dark energy parametrizations: current results and forecasts*, *Mon. Not. Roy. Astron. Soc.* **422** (2012) 776 [arXiv:1105.4943] [ɪɴSPIRE].

[19] G.-B. Zhao, D. Bacon, R. Maartens, M. Santos and A. Raccanelli, *Model-independent constraints on dark energy and modified gravity with the SKA*, arXiv:1501.03840 [ɪɴSPIRE].

[20] C. Escamilla-Rivera, *Status on bidimensional dark energy parameterizations using SNe Ia JLA and BAO datasets*, *Galaxies* **4** (2016) 8 [arXiv:1605.02702] [ɪɴSPIRE].

[21] M. Rezaei, M. Malekjani, S. Basilakos, A. Mehrabi and D.F. Mota, *Constraints to Dark Energy Using PADE Parameterizations*, *Astrophys. J.* **843** (2017) 65 [arXiv:1706.02537] [ɪɴSPIRE].

[22] C. Escamilla-Rivera and S. Capozziello, *Unveiling cosmography from the dark energy equation of state*, *Int. J. Mod. Phys.* **D 28** (2019) 1950154 [arXiv:1905.04602] [ɪɴSPIRE].

[23] L.G. Jaime, L. Patiño and M. Salgado, *Note on the equation of state of geometric dark-energy in f(R) gravity*, *Phys. Rev.* **D 89** (2014) 084010 [arXiv:1312.5428] [ɪɴSPIRE].

[24] R. Lazkoz, M. Ortiz-Baños and V. Salzano, *f(R) gravity modifications: from the action to the data*, *Eur. Phys. J.* **C 78** (2018) 213 [arXiv:1803.05638] [ɪɴSPIRE].

[25] S. Capozziello, R. D'Agostino and O. Luongo, *Extended Gravity Cosmography*, *Int. J. Mod. Phys.* **D 28** (2019) 1930016 [arXiv:1904.01427] [ɪɴSPIRE].

[26] J. Alberto Vazquez, M. Bridges, M.P. Hobson and A.N. Lasenby, *Reconstruction of the Dark Energy equation of state*, *JCAP* **09** (2012) 020 [arXiv:1205.0847] [ɪɴSPIRE].

[27] M. Seikel, C. Clarkson and M. Smith, *Reconstruction of dark energy and expansion dynamics using Gaussian processes*, *JCAP* **06** (2012) 036 [arXiv:1204.2832] [ɪɴSPIRE].

[28] A. Montiel, R. Lazkoz, I. Sendra, C. Escamilla-Rivera and V. Salzano, *Nonparametric reconstruction of the cosmic expansion with local regression smoothing and simulation extrapolation*, *Phys. Rev.* **D 89** (2014) 043007 [arXiv:1401.4188] [ɪɴSPIRE].

[29] G.-B. Zhao et al., *Dynamical dark energy in light of the latest observations*, *Nat. Astron.* **1** (2017) 627 [arXiv:1701.08165] [ɪɴSPIRE].

[30] S. Capozziello, *Curvature quintessence*, *Int. J. Mod. Phys.* **D 11** (2002) 483 [gr-qc/0201033] [INSPIRE].

[31] L.G. Jaime, M. Jaber and C. Escamilla-Rivera, *New parametrized equation of state for dark energy surveys*, *Phys. Rev.* **D 98** (2018) 083530 [arXiv:1804.04284] [INSPIRE].

[32] https://www.tensorflow.org.

[33] G. Aurelien, *Hands-On Machine Learning with Scikit-Learn and Tensorflow: Concepts, Tools, and Techniques to Build Intelligent Systems*, O'Reilly Media (2017) [ISBN-10:1491962291].

[34] T. Charnock and A. Moss, *Deep Recurrent Neural Networks for Supernovae Classification*, *Astrophys. J.* **837** (2017) L28 [arXiv:1606.07442] [INSPIRE].

[35] A. Mathuriya et al., *CosmoFlow: Using Deep Learning to Learn the Universe at Scale*, arXiv:1808.04728 [INSPIRE].

[36] R. Kessler, A. Conley, S. Jha and S. Kuhlmann, *Supernova Photometric Classification Challenge*, arXiv:1001.5210 [INSPIRE].

[37] A. Moss, *Improved Photometric Classification of Supernovae using Deep Learning*, arXiv:1810.06441 [INSPIRE].

[38] A. Moss, *Accelerated Bayesian inference using deep learning*, arXiv:1903.10860 [INSPIRE].

[39] P.J.E. Peebles and B. Ratra, *The Cosmological Constant and Dark Energy*, *Rev. Mod. Phys.* **75** (2003) 559 [astro-ph/0207347] [INSPIRE].

[40] M. Chevallier and D. Polarski, *Accelerating universes with scaling dark matter*, *Int. J. Mod. Phys.* **D 10** (2001) 213 [gr-qc/0009008] [INSPIRE].

[41] E.V. Linder, *The Dynamics of Quintessence, The Quintessence of Dynamics*, *Gen. Rel. Grav.* **40** (2008) 329 [arXiv:0704.2064] [INSPIRE].

[42] B.C. Paul and P. Thakur, *Observational constraints on modified Chaplygin gas from cosmic growth*, *JCAP* **11** (2013) 052 [arXiv:1306.4808] [INSPIRE].

[43] D.M. Scolnic et al., *The Complete Light-curve Sample of Spectroscopically Confirmed SNe Ia from Pan-STARRS1 and Cosmological Constraints from the Combined Pantheon Sample*, *Astrophys. J.* **859** (2018) 101 [arXiv:1710.00845] [INSPIRE].

[44] M. Ntampaka et al., *The Role of Machine Learning in the Next Decade of Cosmology*, arXiv:1902.10159 [INSPIRE].

[45] J. Schmelzle et al., *Cosmological model discrimination with Deep Learning*, arXiv:1707.05167 [INSPIRE].

[46] S. Ruder, *An overview of gradient descent optimization algorithms*, arXiv:1609.04747 [INSPIRE].

[47] A. Möller and T. de Boissière *SuperNNova: an open-source framework for Bayesian, neural network-based supernova classification*, *Mon. Not. Roy. Astron. Soc.* **491** (2020) 4277 [arXiv:1901.06384].

[48] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT Press (2016) [http://www.deeplearningbook.org].

[49] W. Zaremba and I. Sutskever, *Reinforcement Learning Neural Turing Machines — Revised*, arXiv:1505.00521.

[50] D. Pedamonti, *Comparison of non-linear activation functions for deep neural networks on MNIST classification task*, arXiv:1804.02763.

[51] Y. Gal and Z. Ghahramani, *A Theoretically Grounded Application of Dropout in Recurrent Neural Networks*, arXiv:1512.05287.

[52] C. Louizos and M. Welling, *Structured and Efficient Variational Deep Learning with Matrix Gaussian Posteriors*, `arXiv:1603.04733`.

[53] R. Bayes, *An essay toward solving a problem in the doctrine of chances*, *Phil. Trans. Roy. Soc. Lond.* **53** (1764) 370 [ɪɴSPIRE].

[54] P. Gregory *Bayesian Logical Data Analysis for the Physical Sciences*, Cambridge University Press, New York, U.S.A. (2005).

[55] R. Trotta, *Applications of Bayesian model selection to cosmological parameters*, *Mon. Not. Roy. Astron. Soc.* **378** (2007) 72 [`astro-ph/0504022`] [ɪɴSPIRE].

[56] J. Skilling, *Nested Sampling for General Bayesian Computation*, *Bayesian Annal.* **1** (2006) 833 [http://www.mrao.cam.ac.uk/~steve/maxent2009/images/skilling.pdf].

[57] A.R. Liddle, P. Mukherjee, D. Parkinson and Y. Wang, *Present and future evidence for evolving dark energy*, *Phys. Rev.* **D 74** (2006) 123506 [`astro-ph/0610126`] [ɪɴSPIRE].

[58] H. Jeffreys, *Theory of Probability*, 3rd edition, Oxford University Press, Oxford, U.K. (1998).

[59] A.R. Liddle, *How many cosmological parameters?*, *Mon. Not. Roy. Astron. Soc.* **351** (2004) L49 [`astro-ph/0401198`] [ɪɴSPIRE].

[60] Y. Gal and Z. Ghahramani, *Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning*, `arXiv:1506.02142`.