RAA 2019 Vol. **19** No. **9**, 137(8pp) doi: 10.1088/1674–4527/19/9/137

http://www.raa-journal.org *http://iopscience.iop.org/raa*

Research in Astronomy and Astrophysics

Model comparison of dark energy models using deep network

Shi-Yu Li[1], Yun-Long Li[2] and Tong-Jie Zhang[1]

[1] Department of Astronomy, Beijing Normal University, Beijing 100875, China; *tjzhang@bnu.edu.cn*
[2] National Space Science Center, Chinese Academy of Sciences, Beijing 100190, China

Received 2019 June 3; accepted 2019 June 21

Abstract This work uses a combination of a variational auto-encoder and generative adversarial network to compare different dark energy models in light of observations, e.g., the distance modulus from type Ia supernovae. The network finds an analytical variational approximation to the true posterior of the latent parameters in the models, yielding consistent model comparison results with those derived by the standard Bayesian method, which suffers from a computationally expensive integral over the parameters in the product of the likelihood and the prior. The parallel computational nature of the network together with the stochastic gradient descent optimization technique leads to an efficient way to compare the physical models given a set of observations. The converged network also provides interpolation for a dataset, which is useful for data reconstruction.

Key words: cosmology: dark energy — methods: statistical — methods: data analysis

1 INTRODUCTION

It is well known that predictions about the universe from ΛCDM are in perfect concordance with observations of the Cosmic Microwave Background (CMB) (Aghanim et al. 2018), type Ia supernovae (SNeIa) (Betoule et al. 2014), and Baryon Acoustic Oscillations (BAO) (Alam et al. 2017), making ΛCDM the standard paradigm in cosmology. Such a successful model, however, still has its own theoretical problems, which are known as fine tuning and cosmic coincidence (Sahni 2002; Peebles & Ratra 2003). Moreover, a few observations such as the Hubble parameter at high redshift (Delubac et al. 2015) and the linear redshift-space distortions (Macaulay et al. 2013) have shown tensions with ΛCDM. All of these motivate research on the universe that allows time-evolving dark energy. People have developed different evolving scalar fields to describe the evolution of dark energy, such as the canonical scalar fields (Caldwell et al. 1998) and phantom fields (Caldwell et al. 2003; Elizalde et al. 2004; Scherrer & Sen 2008). Various parametrizations of evolving dark energy that broadly describe a large number of scalar field dark energy models have also been proposed, such as the Chevallier-Polarski-Linder (CPL) (Chevallier & Polarski 2001) and generalized Chaplygin gas (GCG) models (Thakur et al. 2012). Given a specific model and a set of cosmological data, one can study the evolution of dark energy conveniently.

Then a question of model choice naturally arises with the development of various dark energy models. A variety of methods such as the F-test, Akaike information criterion (AIC) (Penny et al. 2006), Mallows C_p, Bayesian information criterion (BIC) (Penny et al. 2006), minimum description length (MDL) (Rissanen 1978) and Bayesian model averaging have been proposed to select a good or useful model in light of observations. MacKay (1992) strongly recommends using Bayesian evidence to assign preferences to alternative models since the evidence is the Bayesian's transportable quantity between models, and the popular easy-to-use AIC and BIC as well as MDL methods are all approximations to the Bayesian evidence (Penny et al. 2006). The Bayesian evidence for model selection has been applied to the study of cosmology for a long time (Trotta 2008; Martin et al. 2011; Lonappan et al. 2018; Basilakos et al. 2018), and recently a detailed study of Bayesian evidence for a large class of cosmological models taking into account around 21 different dark energy models has been performed by Lonappan et al. (2018). Although Bayesian evidence remains the preferred method compared with information criterions, a full Bayesian inference for model selection is very computationally expensive and often suffers from multi-modal posteriors and parameter degeneracies, which lead to a large time consumption to obtain the final result.

The variational auto-encoder (VAE) and generative adversarial network (GAN) which build upon the variational Bayes theory provide an efficient way to tackle the model selection problem. VAE (Kingma & Welling 2014) has the ability to approximate the generative process (generate the observed data given the model parameters) and the inference process (infer the model parameters given the observations) which allow one to interpolate between the observed values, thus it is useful in the reconstruction problem. GAN with semi-supervised learning (Goodfellow et al. 2014; Salimans et al. 2016) has the ability to effectively learn the distribution of the data, and assign probabilities to different models from where the data may come. Thus the combination of VAE and GAN brings us a novel and convenient way to do data reconstruction and model selection at the same time. Since the variational method provides an analytical approximation of the posterior, it is possible to use the fast gradient descent method to find constraints for the parameters rather than using the Monte Carlo Markov Chain approach which may suffer from a low acceptance ratio if the posterior is ill-posed. Moreover, the variational method benefits from natural parallelization of the network computation which can be accelerated by GPU cards.

In this article, we use the VAE-GAN network to learn the distribution of the distance moduli in the ΛCDM, ωCDM and CPL universe models, then feed the observations of SNeIa to the network to reconstruct dark energy and discriminate the most probable model. The statistical background of the VAE and GAN is briefly reviewed in Section 2, and the model structure is described at the end of this section. In Section 3, two toy models are created to test the reconstruction and model discrimination ability of the network. Section 4 describes the observables used in this work, and generation of the training set is introduced. Section 5 reports and discusses the results of the data reconstruction and model comparison given by the network, and some prospects that extend the current work follow the discussion.

2 THE VAE-GAN NETWORK

The VAE-GAN network proposed by Larsen et al. (2016) combines a VAE with a GAN, aiming to use the learned feature representations in the GAN discriminator as the basis for the VAE reconstruction, which results in better capturing the data distribution, improving the quality of the inference and the generative process of the network in light of the data. This section briefly reviews the background of the VAE and GAN, and then introduces the method to do model selection and data reconstruction using VAE-GAN.

2.1 The Variational Autoencoder

A VAE (Kingma & Welling 2014) consists of an encoder $\mathcal{I}$ and a decoder $\mathcal{G}$. The decoder mimics the generative process of a model or a natural phenomenon once given the model parameters or latent variables $\boldsymbol{\xi}$, yielding the likelihood distribution of the data $\tilde{\boldsymbol{x}} \sim \mathcal{G}(\boldsymbol{x} \mid \boldsymbol{\xi})$. The encoder approximates the inverse process that given a set of observations $\boldsymbol{x}$ it infers the posterior distribution of the model parameters or latent variables $\boldsymbol{\xi} \sim \mathcal{I}(\boldsymbol{\xi} \mid \boldsymbol{x})$.

The optimal $\mathcal{I}$ and $\mathcal{G}$ are obtained by maximizing the lower bound of the marginal likelihood of the observations via variational Bayes (Penny et al. 2006; Kingma & Welling 2014),

$$\begin{aligned}\mathcal{L}(\boldsymbol{x}) \geq &- D_{\mathrm{KL}}\left[\mathcal{I}(\boldsymbol{\xi} \mid \boldsymbol{x}) \| p(\boldsymbol{\xi})\right] \\ &+ \mathbb{E}_{\mathcal{I}(\boldsymbol{\xi}|\boldsymbol{x})}\left[\log \mathcal{G}(\boldsymbol{x} \mid \boldsymbol{\xi})\right] .\end{aligned} \tag{1}$$

Here, the first item $D_{\mathrm{KL}}[\cdot\|\cdot]$ is the Kullback-Leibler (KL) divergence which measures the difference between two distributions. $p(\boldsymbol{\xi})$ is the prior distribution of the latent variables. $\log \mathcal{G}(\boldsymbol{x} \mid \boldsymbol{\xi})$ is the likelihood of the data. The marginal likelihood $\mathcal{L}(\boldsymbol{x})$ equals its lower bound if and only if the approximate posterior $\mathcal{I}(\boldsymbol{\xi} \mid \boldsymbol{x})$ is the same as the true posterior $\mathcal{G}(\boldsymbol{\xi} \mid \boldsymbol{x})$. Equation (1) implies that the variational optimal encoder and decoder should constrain the posterior close to the prior while keeping the likelihood as large as possible.

2.2 The Generative Adversarial Network

A GAN (Goodfellow et al. 2014) consists of a generator $\mathcal{G}$ and a discriminator $\mathcal{D}$. The generator functions similarly to the decoder in that it maps the latent variables $\boldsymbol{\xi} \sim p(\boldsymbol{\xi})$ to the data space $\boldsymbol{x} = \mathcal{G}(\boldsymbol{\xi}) \in p_{\mathcal{G}}(\boldsymbol{x})$, but the difference is that the mapping is determinant and the sampling process happens only at the latent space. The discriminator assigns probability $\mathcal{D}(\boldsymbol{x}) \in [0, 1]$ to $\boldsymbol{x}$ to tell how probable the real data are (not produced by the generator). The optimal $\mathcal{G}$ and $\mathcal{D}$ are obtained by searching the Nash equilibrium of the minmax game with the value function

$$\begin{aligned}\min_{\mathcal{G}} \max_{\mathcal{D}} V(\mathcal{G}, \mathcal{D}) = &\mathbb{E}_{p(\boldsymbol{x})}\left[\log \mathcal{D}(\boldsymbol{x})\right] \\ &+ \mathbb{E}_{p_{\mathcal{G}}(x)}\left[\log(1 - \mathcal{D}(\boldsymbol{x}))\right] ,\end{aligned} \tag{2}$$

where $p(x)$ is the distribution of the real data. A small modification of the game (Salimans et al. 2016) allows $\mathcal{D}$ to classify $\boldsymbol{x}$ into one of $K + 1$ possible classes, for example, to tell which one of the K classes of dark energy models is the most probable that $\boldsymbol{x}$ is generated from, or $\boldsymbol{x}$ is just the output of the generator which thus belongs to the

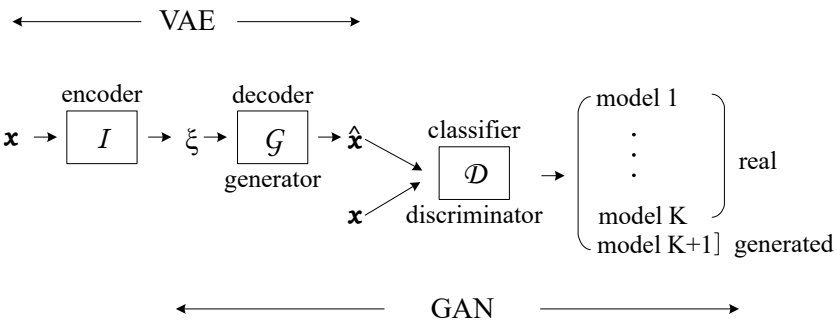

Fig. 1 The structure of the VAE-GAN network (reproduced from Larsen et al. (2016) with an additional classifier described in Salimans et al. (2016)).

$(K+1)$-th class.

$$\begin{aligned}\min_{\mathcal{G}}\max_{\mathcal{D}}\hat{V}(\mathcal{G},\mathcal{D}) =&\mathbb{E}_{p(\boldsymbol{x})}\left[\log\mathcal{D}(c\neq K+1\mid\boldsymbol{x})\right]\\ &+\mathbb{E}_{p_{\mathcal{G}}(x)}\left[\log(1-\mathcal{D}(c\neq K+1\mid\boldsymbol{x}))\right]\\ &+\mathbb{E}_{p(\boldsymbol{x},c)}\left[\log\mathcal{D}(c\mid\boldsymbol{x},c<K+1)\right].\end{aligned}\tag{3}$$

Here c is the label of the model. $\mathcal{D}(c\neq K+1\mid\boldsymbol{x})$ corresponds to $\mathcal{D}(\boldsymbol{x})$ in Equation (2), giving the probability that $\boldsymbol{x}$ is classified as real. $p(\boldsymbol{x},c)$ is the joint distribution of the real data and the model class. $\mathcal{D}(c\mid\boldsymbol{x},c<K+1)$ is the probability that $\boldsymbol{x}$ is classified to the right model c.

2.3 Training Algorithm

The combination of VAE and GAN provides a convenient way to do data interpolation and model selection at the same time, once a set of optimal $\{\mathcal{I},\mathcal{G},\mathcal{D}\}$ is obtained by optimizing Equation (1) and Equation (3). The basic logic of the VAE-GAN network is shown in Figure 1. The observed data $\boldsymbol{x}$ are fed to the encoder $\mathcal{I}$ to find the posterior. Then $\boldsymbol{\xi}$ is sampled from the posterior and fed to the decoder/generator $\mathcal{G}$ to derive the reconstruction (or interpolation) of the input data $\hat{\boldsymbol{x}}$. Finally, the discriminator/classifier extracts the useful features from the reconstruction to derive the probability that $\hat{\boldsymbol{x}}$ belongs to a certain model.

Now the remaining question is that given a set of observations $\{\boldsymbol{x}_i\}_{i=1}^N$ and their covariance $\Sigma_{\rm obs}$ as well as a set of model candidates $\{M_j\}_{j=1}^K$, how to find the optimal $\mathcal{I},\mathcal{G},\mathcal{D}$. Since Equation (1) and Equation (3) set constraints on functions, any flexible functions that have learning abilities can fit in this work. A possible choice is the convolutional neural network (CNN) which is good at representation learning and shift-invariant feature extraction. Suppose $\mathcal{I},\mathcal{G},\mathcal{D}$ are CNNs whose parameters are $\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\psi}$ respectively. One can generate a batch of training samples from the model candidates and train the networks on these fixed data using stochastic gradient descent

1. Select $\{\boldsymbol{x}_i,c_i\}$ from the training samples and retrieve the observed part $\boldsymbol{x}_{i,\rm obs}$, $c_j\in\{1,2,\cdots,K\}$ is the class label. Adding multivariate Gaussian random noise $\boldsymbol{\sigma}_{\rm obs}\sim\mathcal{N}(0,\Sigma_{\rm obs})$ to $\boldsymbol{x}_{i,\rm obs}$ yields $\boldsymbol{x}^*_{i,\rm obs}=\boldsymbol{x}_{i,\rm obs}+\sigma_{\rm obs}$;
2. Feed $\boldsymbol{x}^*_{i,\rm obs}$ to the encoder to get the posterior $\mathcal{I}_{\boldsymbol{\theta}}(\boldsymbol{\xi}\mid\boldsymbol{x}^*_{i,\rm obs})$, then calculate the KL divergence $D_{\rm KL}\left[\mathcal{I}_{\boldsymbol{\theta}}(\boldsymbol{\xi}\mid\boldsymbol{x}^*_{i,\rm obs})\|p(\boldsymbol{\xi})\right]$ (corresponding to the first item in Eq. (1)). Suppose the posterior is a multivariate Gaussian distribution with diagonal covariance, $\mathcal{I}_{\boldsymbol{\theta}}(\boldsymbol{\xi}\mid\boldsymbol{x}^*_{i,\rm obs})=\mathcal{N}(\boldsymbol{\mu}_i,\mathbf{I}\boldsymbol{\sigma}_i^2)$, and the prior is the standard normal distribution, $p(\boldsymbol{\xi})=\mathcal{N}(\mathbf{0},\mathbf{I})$, then the KL divergence can be analytically written as $-\frac{1}{2}\sum(\mathbf{1}+\log\boldsymbol{\sigma}_i^2-\boldsymbol{\mu}_i^2-\boldsymbol{\sigma}_i^2)$, where the square and sum operations are element-wise (Kingma & Welling 2014). Find the gradient of the KL divergence with respect to the parameter of the encoder,
$$\Delta\boldsymbol{\theta}_{{\rm KL},i}=-\nabla_{\boldsymbol{\theta}}D_{\rm KL}\left[\mathcal{I}_{\boldsymbol{\theta}}(\boldsymbol{\xi}\mid\boldsymbol{x}^*_{i,\rm obs})\|p(\boldsymbol{\xi})\right].\tag{4}$$
3. Sample $\boldsymbol{\xi}_i$ from the posterior and feed it to the generator to obtain the reconstruction $\tilde{\boldsymbol{x}}_i=\mathcal{G}_{\boldsymbol{\phi}}(\boldsymbol{\xi}_i)$. The observed part of the reconstruction $\tilde{\boldsymbol{x}}_{i,\rm obs}$ together with $\boldsymbol{x}^*_{i,\rm obs}$ and $\Sigma_{\rm obs}$ gives the negative log likelihood $-\log\mathcal{G}_{\boldsymbol{\phi}}(\boldsymbol{x}^*_{i,\rm obs}\mid\boldsymbol{\xi}_i)=\frac{1}{2}\chi^2+{\rm const.}$ (corresponding to the second item in Eq. (1)), where $\chi^2=(\boldsymbol{x}^*_{i,\rm obs}-\tilde{\boldsymbol{x}}_{i,\rm obs})^T\Sigma^{-1}_{\rm obs}(\boldsymbol{x}^*_{i,\rm obs}-\tilde{\boldsymbol{x}}_{i,\rm obs})$ is the goodness of fit that is broadly used in model regression problems. The const. is the normalization constant of the likelihood, having the value of $\frac{N_{\rm obs}}{2}\log(2\pi)+\frac{1}{2}\log\det\Sigma_{\rm obs}$, where $N_{\rm obs}$ is the dimension of the covariance $\Sigma_{\rm obs}$. Because likelihood depends on both the encoder and generator, its gradient provides an update to $\mathcal{I}_{\boldsymbol{\theta}},\mathcal{G}_{\boldsymbol{\phi}}$,
$$\begin{aligned}\Delta\boldsymbol{\theta}_{\mathcal{L},i}&=\nabla_{\boldsymbol{\theta}}\log\mathcal{G}_{\boldsymbol{\phi}}(\boldsymbol{x}^*_{i,\rm obs}\mid\boldsymbol{\xi}_i),\\ \Delta\boldsymbol{\phi}_{\mathcal{L},i}&=\nabla_{\boldsymbol{\phi}}\log\mathcal{G}_{\boldsymbol{\phi}}(\boldsymbol{x}^*_{i,\rm obs}\mid\boldsymbol{\xi}_i).\end{aligned}\tag{5}$$
4. Sample $\boldsymbol{\xi}_j$ from the prior $p(\boldsymbol{\xi})$ and feed to $\mathcal{G}_{\boldsymbol{\phi}}$ to generate a new sample $\tilde{\boldsymbol{x}}_j$. Feed $\boldsymbol{x}_i,\tilde{\boldsymbol{x}}_i,\boldsymbol{x}_j$ to the discriminator $\mathcal{D}_{\boldsymbol{\psi}}$ to obtain the logits $\boldsymbol{l}_i,\tilde{\boldsymbol{l}}_i,\boldsymbol{l}_j$ which can be interpreted as probabilities, e.g., $\mathcal{D}_{\boldsymbol{\psi}}(k\mid$

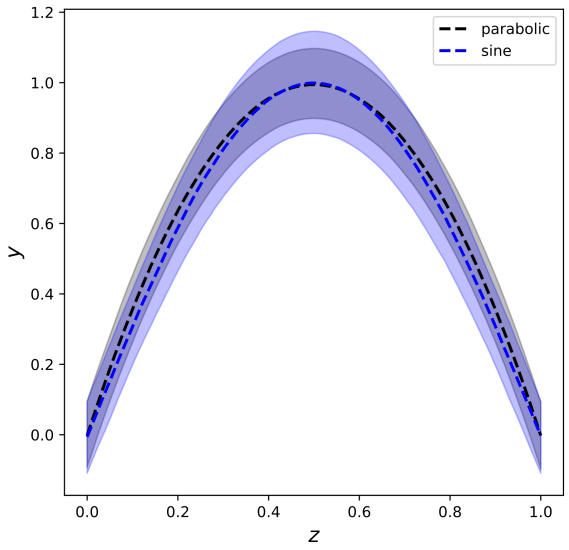

Fig. 2 The distribution of the outputs of the toy models.

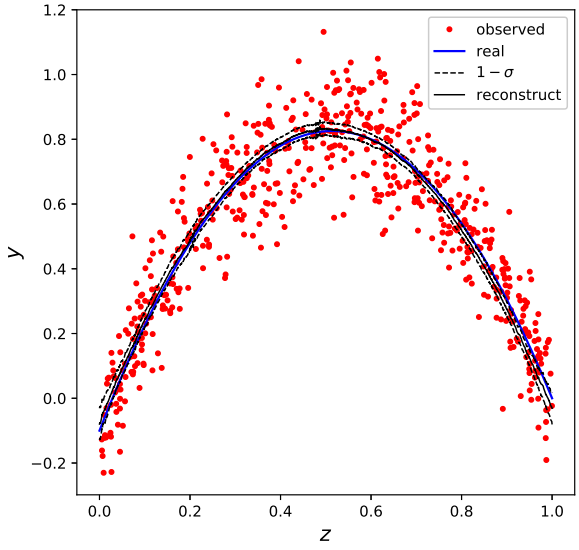

Fig. 3 Reconstruction of the data. The *red dots* represent the observed data with noise characterized by $\Sigma_{\rm obs}$. The *blue line* shows the true model from where the observations are generated. The *black line* is the reconstruction of the data by the network given the observations.

$\boldsymbol{x}_i) = \exp(l_{ik})/\sum_k \exp(l_{ik})$ and l_{ik} is the k-th element of the logit $\boldsymbol{l}_i$. Substituting the probabilities into Equation (3) yields,

$$\begin{aligned}\hat{V}(\mathcal{G}_{\boldsymbol{\phi}}, \mathcal{D}_{\boldsymbol{\psi}}) = &\log \mathcal{D}_{\boldsymbol{\psi}}(c = c_i \mid \boldsymbol{x}_i) \\ &+ \log \mathcal{D}_{\boldsymbol{\psi}}(c = K+1 \mid \tilde{\boldsymbol{x}}_i) \\ &+ \log \mathcal{D}_{\boldsymbol{\psi}}(c = K+1 \mid \tilde{\boldsymbol{x}}_j).\end{aligned} \tag{6}$$

The gradient of $\hat{V}(\mathcal{G}_{\boldsymbol{\phi}}, \mathcal{D}_{\boldsymbol{\psi}})$ provides a modification to $\mathcal{G}_{\boldsymbol{\phi}}, \mathcal{D}_{\boldsymbol{\psi}}$,

$$\begin{aligned}\Delta\boldsymbol{\phi}_{\hat{V},ij} &= -\nabla_\phi \hat{V}(\mathcal{G}_{\boldsymbol{\phi}}, \mathcal{D}_{\boldsymbol{\psi}}), \\ \Delta\boldsymbol{\psi}_{\hat{V},ij} &= +\nabla_\psi \hat{V}(\mathcal{G}_{\boldsymbol{\phi}}, \mathcal{D}_{\boldsymbol{\psi}}).\end{aligned} \tag{7}$$

5. Update the parameters of the encoder, the generator and the discriminator using a learning rate of α,

$$\begin{aligned}\boldsymbol{\theta} &\leftarrow \boldsymbol{\theta} + \alpha(\Delta\boldsymbol{\theta}_{{\rm KL},i} + \Delta\boldsymbol{\theta}_{\mathcal{L},i}), \\ \boldsymbol{\phi} &\leftarrow \boldsymbol{\phi} + \alpha(\Delta\boldsymbol{\phi}_{\mathcal{L},i} + \Delta\boldsymbol{\phi}_{\hat{V},ij}), \\ \boldsymbol{\psi} &\leftarrow \boldsymbol{\psi} + \alpha\Delta\boldsymbol{\psi}_{\hat{V},ij}.\end{aligned} \tag{8}$$

The training process can be easily generalized to mini-batch training to obtain a faster convergence rate. Several training techniques that stabilize or accelerate the training process are also applicable in this problem (Radford et al. 2015; Salimans et al. 2016; Sønderby et al. 2017; Mescheder et al. 2018).

The encoder consists of four 1-D convolutional layers and two dense layers. Each layer is followed by a batch renormalization layer (Ioffe 2017) and an activation layer with Leaky Rectified Linear Unit (a simple variant of ReLU, Nair & Hinton 2010), except the last layer which acts as the output. The input of the encoder has a size of 580 which is the number of distance moduli in the Union2.1 dataset, a compilation of SNeIa, later introduced in Section 4. The dimension of the latent variable should be no less than the number of parameters in the physical models used in the problem, and we set it 20. The size of the convolutional kernel is fixed to 7 and the stride is 4 except for the input layer whose convolutional kernel and stride are of size 69 and 1 respectively. The generator consists of one dense layer and four 1-D fractional convolutional layers. The sizes of the convolutional kernel and stride are the same as the encoder (because it is the inverse process of the encoder), except that the output of the last layer has a dimension of 2048. The discriminator consists of four 1-D convolutional layers and one dense layer. The configuration is similar to the encoder, except that the sizes of the input and output are 2048 and $K+1$ respectively.

3 TESTS ON TOY MODELS

This section creates two toy models to test the data reconstruction and model comparison ability of the network.

Model 1,

$$\begin{aligned}&y = Az^2 + (-A+B)z + C, \\ &\text{where } A \sim \mathcal{N}(-4, 0.1), B \sim \mathcal{N}(0, 0.01), C \sim \mathcal{N}(0, 0.1).\end{aligned} \tag{9}$$

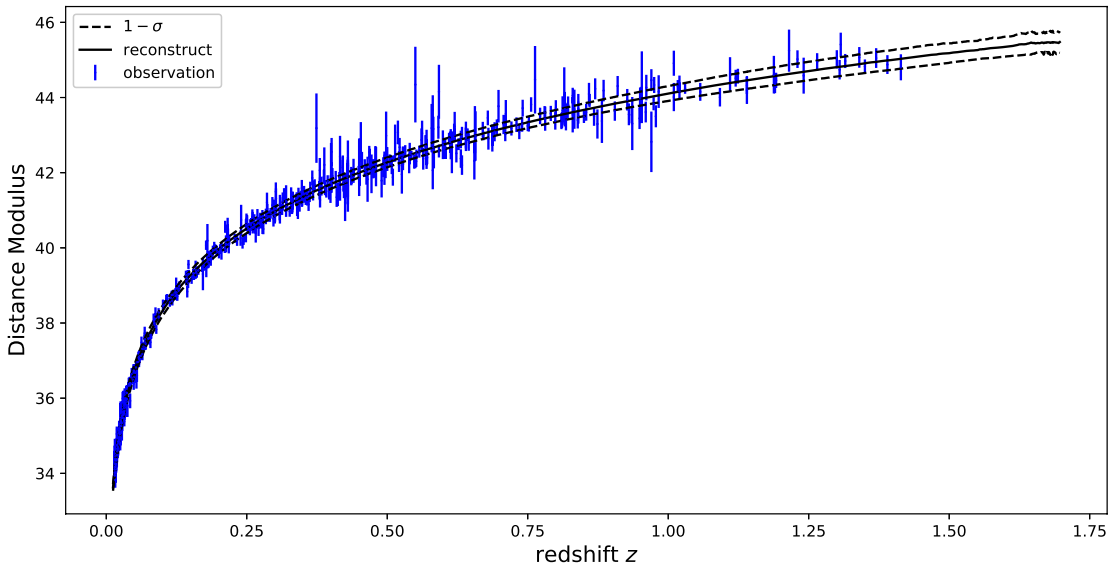

Fig. 4 Reconstruction of the distance modulus by the network.

Model 2,

$$y = A\sin(\omega z) + C,$$
$$\text{where } A \sim \mathcal{N}(1, 0.1), \omega \sim \mathcal{N}(\pi, 0.01), C \sim \mathcal{N}(0, 0.1). \tag{10}$$

Model 1 and Model 2 have similar distributions as shown in Figure 2. Given the observations $\boldsymbol{x}_{\rm obs,real}$ which are generated by the underlying model $y_{\rm true} = -3.5z^2 + 3.6z - 0.1$ on $\boldsymbol{z}_{\rm obs} = \{z_1, z_2, \cdots, z_{580}\}$ with an error matrix $\Sigma_{\rm obs}$, we would like to fit the two toy models to the observations to tell which one is most probable to be the true model, and interpolate the data with the model at $\boldsymbol{z}^* = \{z_1^*, \cdots, z_M^*\}$, for example, $\boldsymbol{z}^*$ even staying in the interval $[0, 1]$ with $M = 1468$.

First we concatenate and sort $\boldsymbol{z}$ and $\boldsymbol{z}^*$, and call the new one $\boldsymbol{z}$. Then sample $\{A_i, B_i, C_i, \omega_i\}$ from the priors of the toy models and generate the training samples $\boldsymbol{x}_i = M_k(\boldsymbol{z} \mid A_i, B_i, C_i, \omega_i)$. (Note that which set of parameters should be used depends on the toy model.) Here 12 800 samples for each model are generated as the training dataset. Finally, the training set $\{\boldsymbol{x}\}_{i=1}^{25\,600}$ together with the observation error $\Sigma_{\rm obs}$ is fed into the network. Once the training converges, one can put the observations $\boldsymbol{x}_{\rm obs,real}$ into the network to tell which toy model is most probable and get the interpolation, see Figure 3. In this task, the discriminator has a classification accuracy of almost 1. It assigns a probability of 97% to the parabolic model (Model 1), which is indeed the case.

4 THE DATASET

4.1 The Observations

The observations are from the Union2.1 compilation (Suzuki et al. 2012) which contains 580 SNeIa. Union2.1 provides the distance moduli with their covariance matrix. Let $\boldsymbol{z}_{\rm obs}$ denote the redshift of the 580 SNeIa and $\boldsymbol{x}_{\rm obs,real}$ signify the measured distance moduli. $\Sigma_{\rm obs}$ represents the covariance of the distance moduli with systematics.

4.2 The Training Set

We study the model comparison problem among three dark energy models: (1) $\omega(z) = -1$ (Λ CDM); (2) $\omega(z) = \omega_{\rm DE}$ (ωCDM); (3) $\omega(z) = \omega_0 + \omega_a \frac{z}{1+z}$ (CPL), given a set of observations of distance moduli at different redshifts. The expansion rate of a spatially flat FRW universe is determined by the matter and dark energy,

$$H^2(z) = H_0^2 \left\{\Omega_{m0}(1+z)^3 + (1-\Omega_{m0}) \exp\left[3\int \frac{1+\omega(z')}{1+z'} dz'\right]\right\}. \tag{11}$$

The luminosity distance is closely related to the Hubble expansion rate (Eq. (12)), and the distance modulus is given by Equation (13).

$$D_L(z) = c(1+z)\int_0^z dz' \frac{1}{H(z')}\,. \tag{12}$$

$$\mu(z) = 5\log_{10} D_L(z) + 25\,. \tag{13}$$

For each dark energy model, 12 800 samples are generated at the redshift $\boldsymbol{z} = \text{sort}\{\boldsymbol{z}_{\rm obs}, \boldsymbol{z}^*\}$, given the priors of the

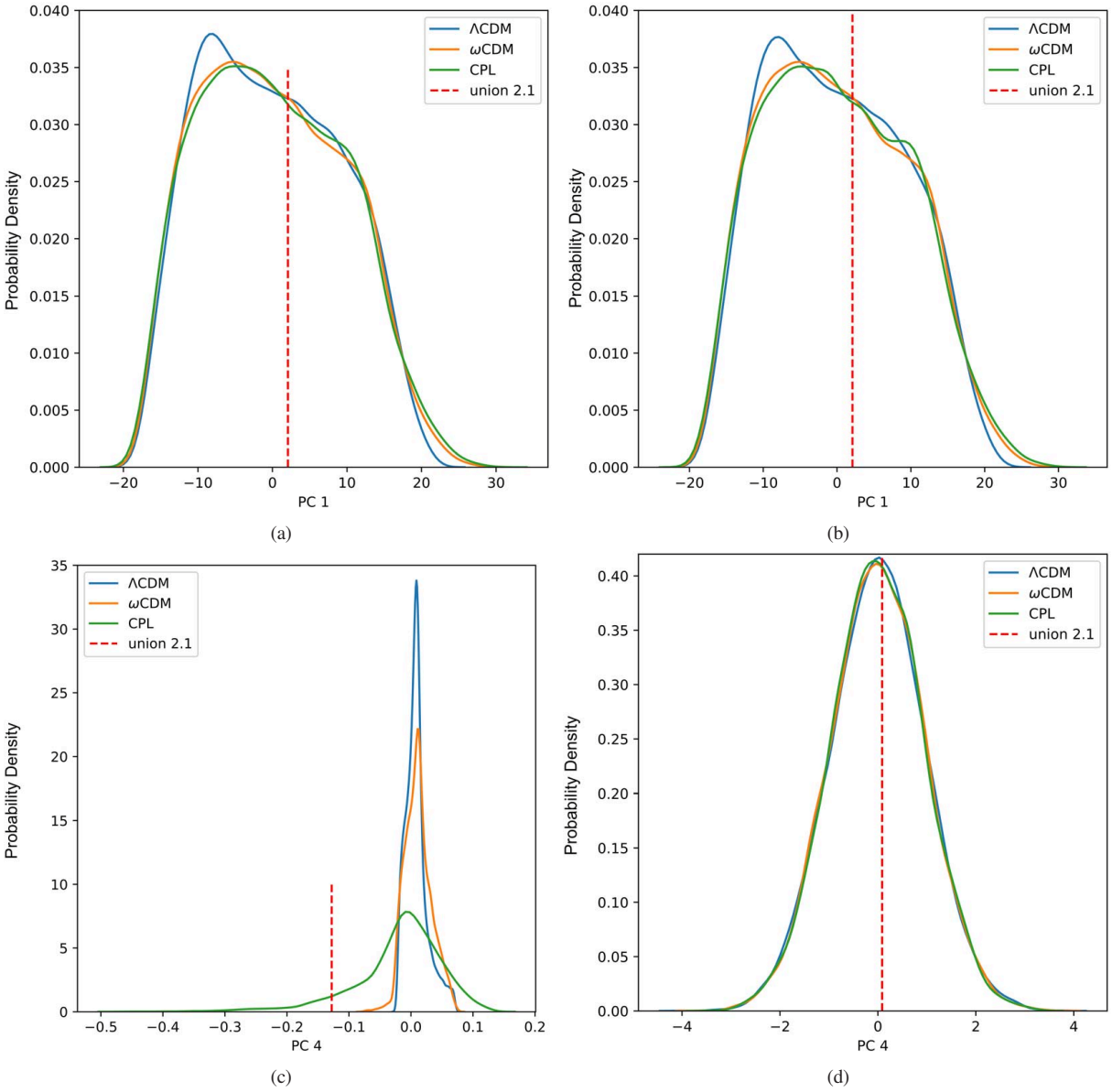

Fig. 5 Normalized distributions of the projections of training data onto the 1st and 4th PCs. The *red dashed line* represents the projection of the Union2.1 data set onto the PCs. (a) The distribution of projection onto the 1st PC with no observation errors; (b) The distribution of the projection onto the 1st PC with the covariance matrix from the Union2.1; (c) The distribution of projection onto the 4th PC with no observation errors; (d) The distribution of the projection onto the 4th PC with the covariance matrix from Union2.1. (a) and (c) share the same set of PCs, while (b) and (d) share another set of PCs.

parameters as,

$$\begin{aligned} \Omega_{m0} &\sim \mathcal{U}(0.1, 0.9)\,, \\ H_0 &\sim \mathcal{U}(50, 90)\,, \\ \omega_{\rm DE} &\sim \mathcal{U}(-1.8, -0.4)\,, \\ \omega_0 &\sim \mathcal{U}(-1.9, -0.4)\,, \\ \omega_a &\sim \mathcal{U}(-4.0, 4.0)\,. \end{aligned} \tag{14}$$

$\boldsymbol{z}^*$ has 1468 elements evenly located in the interval, $[0.8\min(\boldsymbol{z}_{\rm obs}), 1.2\max(\boldsymbol{z}_{\rm obs})]$. The $12\,800\times3$ samples are fixed as the training set.

5 RESULTS AND DISCUSSIONS

Figure 4 shows the reconstruction of the distance modulus produced by the network. Given the observed distance modulus $\boldsymbol{\mu}_{\rm obs}$, the discriminator $\mathcal{D}$ assigns probability to each model with,

$$\begin{aligned} \mathcal{D}(\Lambda{\rm CDM} \mid \boldsymbol{\mu}_{\rm obs}) &= 56.2\%, \\ \mathcal{D}(\omega{\rm CDM} \mid \boldsymbol{\mu}_{\rm obs}) &= 28.6\%, \\ \mathcal{D}({\rm CPL} \mid \boldsymbol{\mu}_{\rm obs}) &= 15.1\%. \end{aligned} \tag{15}$$

We conclude that ΛCDM is slightly more favored than the other two models while CPL is the least favored in light

of the observations. This result is consistent with the one derived by the Bayesian evidence method in Lonappan et al. (2018) that finds the log evidence of each model to be $\log \mathcal{Z}_{\Lambda\mathrm{CDM}} = -68.11, \log \mathcal{Z}_{\omega\mathrm{CDM}} = -69.27$ and $\log \mathcal{Z}_{\mathrm{CPL}} = -69.73$. These evidences can be interpreted into probabilities 66.2%, 20.7% and 13.1%, respectively, given the non-informative prior $p(\Lambda\mathrm{CDM}) = p(\omega\mathrm{CDM}) = p(\mathrm{CPL})$.

The classification accuracy of the three models with the associated observation error is reported as 47.9%. The accuracy is subjectively low, but it is not unexpected. The integration operations in Equations (11) and (12) which act as low-pass filters smooth out the local high frequency features that are useful for model comparison. Thus, the convolutional kernel in the network needs to search for useful low frequency features which are less informative once the output distributions of the models overlap each other. This is the case that we meet in this problem. If one uses $p(\boldsymbol{x} \mid M_i)$ to represent the model's prediction of the distribution of the data (it is the evidence of the model), then the theoretical optimal discriminator assigns each model M_i with a probability of $p(\boldsymbol{x} \mid M_i)/\sum_i p(\boldsymbol{x} \mid M_i)$. Once $\boldsymbol{x}_{\mathrm{obs}}$ drops in the overlapped region where $p(\boldsymbol{x}_{\mathrm{obs}} \mid M_i) \approx p(\boldsymbol{x}_{\mathrm{obs}} \mid M_j), \forall i, j$, the discriminator loses its ability to discriminate the models confidently.

Note that ΛCDM is a special case of ωCDM while the latter is a special case of the CPL model, which means there is always a region overlapped among the data distributions of the three models. If the measurements of the distance moduli are accurate enough, the region covered by $p(\boldsymbol{x} \mid \Lambda\mathrm{CDM})$ is negligible compared to $p(\boldsymbol{x} \mid \omega\mathrm{CDM})$, and the latter is negligible compared to $p(\boldsymbol{x} \mid \mathrm{CPL})$. Thus $\boldsymbol{x}$ randomly generated by ωCDM has an extremely low probability to drop in the region of $p(\boldsymbol{x} \mid \Lambda\mathrm{CDM})$ and conversely a high probability is assigned to $\boldsymbol{x}$ that it comes from ΛCDM if it falls in the region of $p(\boldsymbol{x} \mid \Lambda\mathrm{CDM})$. This situation is also applicable to the comparison between ωCDM and CPL. Then the discriminator has great confidence to tell from which model $\boldsymbol{x}$ comes.

Figure 5 is an illustration of the discussion above. The left column shows the normalized histograms of projections of the training samples to their 1st and 4th principal components (PCs) with no observation errors. The upper left panel reveals that the low frequency part (1st PC) of the model contributes little to the model discrimination, because the projection of the Union2.1 data to the 1st PC drops in the region where all the models have similar probabilities. The lower left panel demonstrates that the high frequency part (4th PC) of the model is useful for model discrimination, because the projection of the Union2.1 data on the 4th PC is located in the region where the model has obviously different probabilities.

The discriminator degrades, however, if a non-zero observation error Σ_{obs} is involved in the problem. The overlapped region expands due to the errors so that it is not negligible anymore. An extreme limit is that the errors become infinity, thus the distribution of the three models becomes the same so that the discriminator can only make a random guess about which model is true. In this situation, the accuracy degrades to $1/3$. Finite observation errors lead to a non-negligible intersection where the discriminator loses the ability to tell confidently from which model the data come. This is illustrated in the bottom row of Figure 5. The lower right panel shows the distribution of 4th PC scores of the training samples with the covariance matrix of Union2.1. The projection of the Union2.1 data onto the 4th PC is now located in the region where the different models have similar probabilities. This explains why the network yields a result that is in good concordance with the standard Bayesian analysis but has a subjectively low classification accuracy – the model classification accuracy is intrinsically determined by the nested structure of the three models as well as the observation noise. The variational network successfully learns the posterior distribution and the likelihood distribution to produce a consistent result.

Although this work uses the distance modulus for model comparison and data reconstruction, it is easy to extend the scenario to Hubble parameters or another dataset. The framework should be further considered to include not only $\boldsymbol{x}_{\mathrm{obs}}$ but also its n-th derivatives $\boldsymbol{x}^{(n)}_{\mathrm{obs}}$, e.g., both the angular diameter distance and the Hubble parameter measured by BAO, to let the encoder and discriminator benefit from different datasets. Another improvement of the framework is to allow the reconstruction of the data to implicitly include a model averaging process which will enhance the generalization of the reconstruction, for example, extend the VAE-GAN to its more powerful variant CVAE-GAN (Bao et al. 2017). These are left to future works.

Acknowledgements This work was funded by the National Natural Science Foundation of China (Grant Nos. 11573006 and 11528306), the National Key R&D Program of China (2017YFA0402600) and the 13th Five-year Informatization Plan of Chinese Academy of Sciences (XXH13505-04).

References

Aghanim, N., Akrami, Y., Ashdown, M., et al. 2018, arXiv:1807.06209

Alam, S., Ata, M., Bailey, S., et al. 2017, Monthly Notices of the Royal Astronomical Society, 470, 2617

Bao, J., Chen, D., Wen, F., Li, H., & Hua, G. 2017, in Proceedings of the IEEE International Conference on Computer Vision, 2745, https://ieeexplore.ieee.org/abstract/document/8237561

Basilakos, S., et al. 2018, The European Physical Journal C, 78, 889

Betoule, M., Kessler, R., Guy, J., et al. 2014, A&A, 568, 22

Caldwell, R. R., Dave, R., & Steinhardt, P. J. 1998, Physical Review Letters, 80, 1582

Caldwell, R. R., Kamionkowski, M., & Weinberg, N. N. 2003, Physical Review Letters, 91, 071301

Chevallier, M., & Polarski, D. 2001, International Journal of Modern Physics D, 10, 213

Delubac, T., Bautista, J. E., Busca, N. G., Rich, J., et al. 2015, A&A, 574, 59

Elizalde, E., Nojiri, S., & Odintsov, S. D. 2004, Physical Review D, 70, 043539

Goodfellow, I., Pouget-Abadie, J., Mirza, M., et al. 2014, Advances in Neural Information Processing Systems , eds. Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, & K. Q. Weinberger (Curran Associates, Inc.), 2672, https://papers.nips.cc/paper/5423-generative-adversarial-nets

Ioffe, S. 2017, Advances in Neural Information Processing Systems 30, 1945

Kingma, D. P., & Welling, M. 2014, Second International Conference on Learning Representations (ICLR), arXiv:1312.6114

Larsen, A. B. L., S?nderby, S. K., Larochelle, H., & Winther, O. 2016, Proceedings of The 33rd International Conference on Machine Learning, eds. M. F. Balcan & K. Q. Weinberger (New York, USA: PMLR), 48, 1558, http://proceedings.mlr.press/v48/larsen16.pdf

Lonappan, A. I., Kumar, S., Dinda, B. R., Sen, A. A., et al. 2018, Physical Review D, 97, 043524

Macaulay, E., Wehus, I. K., & Eriksen, H. K. 2013, Physical Review Letters, 111, 161301

MacKay, D. J. 1992, Neural Computation, 4, 415

Martin, J., Ringeval, C., & Trotta, R. 2011, Physical Review D, 83, 063524

Mescheder, L., Nowozin, S., & Geiger, A. 2018, Proceedings of the conference "Neural Information Processing systems 2017", eds. Guyon, I., Luxburg, U.v., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., & Garnett, R.

Nair, V., & Hinton, G. E. 2010, in Proceedings of the 27th International Conference on Machine Learning (ICML-10), 807

Peebles, P. J. E., & Ratra, B. 2003, Reviews of Modern Physics, 75, 559

Penny, W., Mattout, J., & Trujillo-Barreto, N. 2006, Statistical Parametric Mapping: The Analysis of Functional Brain Images (London: Elsevier), 454

Radford, A., Metz, L., & Chintala, S. 2015, arXiv:1511.06434

Rissanen, J. 1978, Automatica, 14, 465, http://dx.doi.org/10.1016/0005-1098(78)90005-5

Sahni, V. 2002, Classical and Quantum Gravity, 19, 3435

Salimans, T., Goodfellow, I., Zaremba, W., et al. 2016, in Advances in Neural Information Processing Systems 29, eds. Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., & Garnett, R. (Curran Associates, Inc.), 2234

Scherrer, R. J., & Sen, A. 2008, Physical Review D, 78, 067303

Sønderby, C. K., Caballero, J., Theis, L., Shi, W., & Huszár, F. 2017, arXiv:1610.04490

Suzuki, N., Rubin, D., Lidman, C., et al. 2012, ApJ, 746, 85

Thakur, S., Nautiyal, A., Sen, A. A., & Seshadri, T. 2012, MNRAS, 427, 988

Trotta, R. 2008, Contemporary Physics, 49, 71