Nonlinearity-free prediction of the growth-rate $f\sigma_8$ using convolutional neural networks

Koya Murakami,[1,*] Indira Ocampo,[2,†] Savvas Nesseris,[2,‡] Atsushi J. Nishizawa,[3,4,5,§] and Sachiko Kuroyanagi[2,1,∥]

[1]*Department of Physics, Nagoya University, Furocho, Chikusa, Nagoya, 464-8602, Japan*
[2]*Instituto de Física Teórica UAM-CSIC, Universidad Autonóma de Madrid,*
Cantoblanco, 28049 Madrid, Spain
[3]*Gifu Shotoku Gakuen University, Takakuwa-Nishi, Yanaizu, Gifu, 501-6194, Japan*
[4]*Institute for Advanced Research, Nagoya University, Furocho, Chikusa, Nagoya, Aichi, 464-8602, Japan*
[5]*Kobayashi Maskawa Institute, Nagoya University, Furocho, Chikusa, Nagoya, Aichi, 464-8602, Japan*

(Received 3 January 2024; accepted 10 June 2024; published 19 July 2024)

The growth-rate $f\sigma_8(z)$ of the large-scale structure of the Universe is an important dynamic probe of gravity that can be used to test for deviations from general relativity. However, for galaxy surveys to extract this key quantity from cosmological observations, two important assumptions have to be made: (i) a fiducial cosmological model, typically taken to be the cosmological constant and cold dark matter model and (ii) the modeling of the observed power spectrum, especially at nonlinear scales, which is particularly dangerous as most models used in the literature are phenomenological at best. In this work, we propose a novel approach involving convolutional neural networks (CNNs), trained on the Quijote N-body simulations, to predict $f\sigma_8(z)$ directly and without assuming a model for the nonlinear part of the power spectrum, thus avoiding the second of the assumptions above. This could serve as an initial step towards the future development of a method for parameter inference in stage IV surveys. We find that the predictions for the value of $f\sigma_8$ from the CNN are in excellent agreement with the fiducial values since they outperform a maximum likelihood analysis and the CNN trained on the power spectrum. Therefore, we find that the CNN reconstructions provide a viable alternative to avoid the theoretical modeling of the nonlinearities at small scales when extracting the growth rate.

DOI: 10.1103/PhysRevD.110.023525

I. INTRODUCTION

Recent cosmological observations indicate that the Universe is experiencing accelerated expansion, typically attributed to a new form of matter, i.e., dark energy, that is responsible for this phenomenon. Within the context of the Friedmann-Lemaître-Robertson-Walker cosmology, this new dark energy component is frequently attributed to the cosmological constant Λ, which is introduced regardless of the fine-tuning issues that this may cause [1,2]. Together with the assumption of a cold dark matter (CDM) component, these two ingredients comprise the so-called ΛCDM model, which is currently the concordance cosmological model [3].

The large-scale structure (LSS) of the Universe encodes very important details crucial to testing cosmological models, as it provides information about the late-time evolution of the Universe and the structure of the matter density field. Both are useful for constraining the current values of the fractional density parameters for CDM, baryonic matter, and dark energy: $\Omega_{\mathrm{m},0}$, $\Omega_{\mathrm{b},0}$, and $\Omega_{\Lambda,0}$, respectively, and many other parameters like the clustering strength σ_8 [4,5]. They are also important probes to search for possible deviations from general relativity.

An overriding challenge in modern cosmology is understanding the features of dark energy, and here lies the importance of obtaining highly accurate data from forthcoming cosmological missions such as *Euclid* [6], LSST [7], and DESI [8], which aim to provide measurements to a few percent of the various key parameters related to the LSS. In order to make sense of the plethora of currently available cosmological models and observations [9,10], it is necessary to develop valuable statistical tools. In particular, machine learning (ML) has attracted increasing attention as it simplifies the usually computationally expensive procedures for data treatment [11]. Simulations also play an indispensable role in understanding the whole picture of the LSS since some information is particularly difficult to extract due to the nonlinear nature of the system. This is the motivation behind the Quijote N-body simulations,

[*] Contact author: koya.murakami9627@gmail.com
[†] Contact author: indira.ocampo@csic.es
[‡] Contact author: savvas.nesseris@csic.es
[§] Contact author: atsushi.nisizawa@gifu.shotoku.ac.jp
[∥] Contact author: sachiko.kuroyanagi@csic.es

consisting of CDM particle simulations that allow quantifying data on the matter field in the fully nonlinear regime and with different statistics [12].

One way of probing the dynamics of LSS is by measuring the growth of matter density perturbations $\delta_m = \delta\rho_m/\rho_m$, where ρ_m is the background matter density and $\delta\rho_m$ its linear order perturbation, and its growth rate that is usually represented by its logarithmic derivative $f = \frac{d\ln\delta_m}{d\ln a}$. However, when dealing with galaxy surveys, the observed quantity is, in fact, the galaxy density δ_g, which is connected to the matter perturbations through a bias parameter that changes from survey to survey: $\delta_g = b\delta_m$ and depending on the type of galaxy observed [13]. At some redshift bin z_i, the growth rate $f(z_i)$ can be combined with the bias $b(z_i)$ to give rise to the so-called velocity-density coupling parameter $\beta(z_i) = f(z_i)/b(z_i)$. At the same time, the combination of $b(z_i)\sigma_8(z_i)$ can also be independently measured [14], where $\sigma_8(z)$ is the root mean square density fluctuation in a sphere of radius $R = 8\ \mathrm{Mpc}\,h^{-1}$. Therefore, the combination $f\sigma_8(z) \equiv f(z)\sigma(z)$ is independent of bias and can be measured via redshift-space distortions [15]. Additionally, other approaches were developed in recent years: analyses of galaxy clustering have also been done using the effective field theory of large-scale structure [16,17] and simulation-based emulators were implemented [18–20].

In order to extract $f\sigma_8(z)$ with this approach, two main assumptions have to be made: first, that of a cosmological model (typically assumed to be the ΛCDM model), as we need to convert the redshifts of galaxies and coordinates in the sky to distances, for extracting the correlation function [21]. Second, we need to assume a model for the nonlinear part of the power spectrum, typically done with phenomenological models [22–26].

This approach has also created several $f\sigma_8(z)$ dataset compilations; see, for example, Refs. [14,27,28] and references therein, where the data explicitly depend on the cosmology used. This can be somewhat easily corrected via an Alcock-Paczynski type correction [27], but the second issue of the model dependence on the nonlinear power spectrum is more insidious as all models used are phenomenological, and, thus far, there is no way to correct for that *a posteriori*.

One approach that has been recently proposed to extract cosmological parameters in a theory-agnostic manner is via ML techniques, in particular, by training convolutional neural networks (CNNs) on N-body simulations and then extracting the parameters from the LSS statistics [29]. This approach was also further expanded to extract the cosmological density and velocity fields from N-body simulations [30,31]. These ML methods have the main advantage that after the original training has occurred, any subsequent evaluations are practically instantaneous, and more importantly, they allow for extracting the quantities of interest without assuming a specific functional form for the power spectrum.

Therefore, in this work, we aim to extract and compare the growth-rate $f\sigma_8$ by leveraging ML techniques. We implemented and trained a CNN directly on N-body simulations at different redshift bins and also on the power spectrum of these simulations. While we do not steer clear of the cosmological model premise, we refrain from assuming the modeling of the nonlinear power spectrum.

Lastly, let us clarify that, even if one could measure a dark matter (DM) density map, the analysis would not be as direct and useful as the one with $f\sigma_8$ measurements in the end. The reason is that while the DM density map serves as a valuable tool, model comparison, e.g., ΛCDM versus $f(R)$, is typically conducted using $f\sigma_8$, which essentially compresses the same information. Therefore, having predictions for $f\sigma_8$ in a manner as model independent as possible is highly desirable. An approximate analogy is found in the Planck cosmic microwave background maps at the pixel level. While these maps are useful for analysis and contain comprehensive cosmological data, parameter inference and model selection are primarily based on the angular power spectra.

The layout of our paper is as follows: in Sec. II, we briefly summarize the theoretical background of our analysis, while in Sec. III, we present the details for the Quijote simulations used in our analysis. Then, in Sec. IV, we describe the results of our ML analysis using the density field of the simulations. Finally, we summarize our conclusions in Sec. V.

II. THEORETICAL FRAMEWORK

A. The ΛCDM model and the growth rate

In this work, we will consider general relativity and a flat ΛCDM universe with an equation of state of $w = -1$ for dark energy.

At the background level, the Hubble parameter in a flat ΛCDM model is given by the first Friedmann equation as usual:

$$H(a)^2 = H_0^2(\Omega_{m,0}a^{-3} + 1 - \Omega_{m,0}), \qquad (1)$$

where H_0 is the Hubble constant and the matter density $\Omega_{m,0}$ can be related to the scale factor a by

$$\Omega_m(a) = \frac{\Omega_{m,0}a^{-3}}{H(a)^2/H_0^2}. \qquad (2)$$

Also, assuming a flat universe, we can calculate the comoving distance $D(z)$ from us to a redshift z as

$$D(z) = \int_0^z \frac{cdz'}{H(z')}, \qquad (3)$$

where c is the speed of light and $H(z)$ is Hubble parameter at redshift z calculated via Eq. (1), when neglecting radiation and neutrinos at late times.

On the other hand, observations from the LSS and the cosmic microwave background also suggest the existence of small $\sim\mathcal{O}(10^{-5})$ perturbations. Consequently, we have to work in the framework of a perturbed Friedmann-Lemaître-Robertson-Walker metric, as the gravitational instability produced by density fluctuations plays a crucial role in seeding the structures at large scales. In the conformal Newtonian gauge, we consider scalar metric perturbations Ψ and Φ, so the perturbed metric can be written as [32,33]

$$ds^2 = a(\tau)^2\{-[1+2\Psi(\vec{x},\tau)]d\tau^2 + [1-2\Phi(\vec{x},\tau)]d\vec{x}^2\}, \quad (4)$$

where the potentials depend on the space-time point $x^\mu = (\vec{x},\tau)$, with τ being the conformal time, while the scale factor $a(\tau)$ only depends on the conformal time.

In general, we can assume that the matter component behaves as a perfect fluid and is described by the stress-energy tensor:

$$T^\mu_\nu = P\delta^\mu_\nu + (\rho + P)U^\mu U_\nu, \quad (5)$$

where the four-velocity is $U^\mu = \frac{dx^\mu}{\sqrt{-ds^2}}$, the total density is $\rho = \bar{\rho} + \delta\rho$, the total pressure is $P = \bar{P} + \delta P$, also $\delta\rho = \delta\rho(\vec{x},\tau)$ and $\delta P = \delta P(\vec{x},\tau)$ are the density and pressure perturbations, respectively, while $\bar{\rho} = \bar{\rho}(\tau)$ and $\bar{P} = \bar{P}(\tau)$ are the background energy density and pressure quantities. Therefore, the stress-energy tensor components are [34]

$$T^0_0 = -(\bar{\rho} + \delta\rho), \quad (6)$$

$$T^0_i = (\bar{\rho} + \bar{P})u_i, \quad (7)$$

$$T^i_j = (\bar{P} + \delta P)\delta^i_j + \Sigma^i_j, \quad (8)$$

where $\Sigma^i_j \equiv T^i_j - \delta^i_j T^k_k/3$ is the anisotropic stress and $u = \dot{\vec{x}}$. The dot denotes the derivative with respect to τ [35]. Recall that the energy-momentum tensor follows the conservation law $T^{\mu\nu}{}_{;\nu} = 0$, as a consequence of the Bianchi identities [34].

To study the evolution of the perturbed variables, we resort to the perturbed Einstein equations in k space [32,34]:

$$k^2\Phi + 3\frac{\dot{a}}{a}\left(\dot{\Phi} + \frac{\dot{a}}{a}\Psi\right) = 4\pi G_\mathrm{N} a^2 \delta T^0_0, \quad (9)$$

$$k^2\left(\dot{\Phi} + \frac{\dot{a}}{a}\Psi\right) = 4\pi G_\mathrm{N} a^2 (\bar{\rho} + \bar{P})\theta, \quad (10)$$

$$\ddot{\Phi} + \frac{\dot{a}}{a}(\dot{\Psi} + 2\dot{\Phi}) + \left(2\frac{\ddot{a}}{a} - \frac{\dot{a}^2}{a^2}\right)\Psi + \frac{k^2}{3}(\Phi - \Psi)$$
$$= \frac{4\pi}{3} G_\mathrm{N} a^2 \delta T^i_i, \quad (11)$$

$$k^2(\Phi - \Psi) = 12\pi G_\mathrm{N} a^2 (\bar{\rho} + \bar{P})\sigma, \quad (12)$$

where the fluid velocity is defined via $\theta = ik_j u^j$ and k_j is the wave number of the perturbations in Fourier space. We can also rewrite the anisotropic stress as $(\bar{\rho} + \bar{P})\sigma \equiv -(-\hat{k}_i\hat{k}_j - \frac{1}{3}\delta_{ij})\Sigma^{ij}$.

By taking the following approximations: subhorizon (only the modes in the Hubble radius are important) and quasistatic (neglect terms with time derivatives), we can simplify the perturbed Einstein equations. Let us consider the perturbation of the Ricci scalar and see how it simplifies with these approximations:

$$\delta R = -\frac{12(\mathcal{H}^2 + \dot{\mathcal{H}})}{a^2}\Psi - \frac{4k^2}{a^2}\Phi + \frac{2k^2}{a^2}\Psi$$
$$- \frac{18\mathcal{H}}{a^2}\dot{\Phi} - \frac{6\mathcal{H}}{a^2}\dot{\Psi} - \frac{6\ddot{\Phi}}{a^2},$$
$$\delta R \simeq -\frac{4k^2}{a^2}\Phi + \frac{2k^2}{a^2}\Psi, \quad (13)$$

then, we can find the following expressions for the Newtonian potentials, see [34]

$$\Psi(k,a) = -4\pi G_\mathrm{N}\frac{a^2}{k^2}\mu(k,a)\bar{\rho}_\mathrm{m}\delta_\mathrm{m}, \quad (14)$$

$$\Phi(k,a) = -4\pi G_\mathrm{N}\frac{a^2}{k^2}Q_\mathrm{eff}(k,a)\bar{\rho}_\mathrm{m}\delta_\mathrm{m}, \quad (15)$$

where $\mu(k,a) \equiv G_\mathrm{eff}(k,a)/G_\mathrm{N}$ is used to denote an evolving Newton's constant. In GR, the two parameters $\mu(k,a)$ and $Q_\mathrm{eff}(k,a)$ can be shown to be equal to unity, but in modified gravity theories they are, in general, time and scale dependent [34].

From the context developed before and by using the continuity equations that come from the conservation of the energy-momentum tensor (via the Bianchi identities), we arrive at a second-order differential equation that describes the evolution of the matter density perturbations (in the absence of massive neutrinos), which is valid in the context of most modified gravity models [36]:

$$\delta''_\mathrm{m}(a) + \left(\frac{3}{a} + \frac{H'(a)}{H(a)}\right)\delta'_\mathrm{m}(a)$$
$$- \frac{3\Omega_{\mathrm{m},0}\mu(k,a)}{2a^5 H(a)^2/H_0^2}\delta_\mathrm{m}(a) = 0, \quad (16)$$

for which we assume the initial conditions $\delta_\mathrm{m}(a \ll 1) \sim a$ and $\delta'_\mathrm{m}(a \ll 1) \sim 1$ at some initial time in the matter domination era, e.g., $a = 10^{-3}$.

In GR and the ΛCDM model ($\mu = 1$), the analytical solution for the growing mode can be found by directly solving Eq. (16) and is given by [37–40]

$$\delta_{\rm m}(a) = a \cdot {}_2F_1\left(\frac{1}{3}, 1; \frac{11}{6}; a^3\left(1 - \frac{1}{\Omega_{\rm m,0}}\right)\right), \tag{17}$$

where we use the quasistatic and subhorizon approximations [36] and ${}_2F_1$ is the Gauss hypergeometric function expressed as [41]

$${}_2F_1(a,b,c,z) = \sum_{n=0}^{\infty} \frac{(a)_n (b)_n}{(c)_n} \frac{z^n}{n!}, \tag{18}$$

where $(x)_n$ is the rising factorial calculated as

$$(x)_n = x(x+1)(x+2)\cdots(x+n-1). \tag{19}$$

With this in mind, we can define several key quantities: the growth rate f, and σ_8, which is the root mean square normalization of the matter power spectrum as

$$f(a) = \frac{{\rm d}\log\delta_{\rm m}}{{\rm d}\log a}, \tag{20}$$

$$\sigma_8(a) = \sigma_{8,0}\frac{\delta_{\rm m}(a)}{\delta_{\rm m}(1)}, \tag{21}$$

$$\sigma_{8,0}^2 = \langle\delta_{\rm m}(x)^2\rangle. \tag{22}$$

It is important to mention that, in general, we have to correct the data for the Alcock-Paczynski effect (as different surveys use different fiducial cosmologies) [42–44]. The combination of the growth rate $f(z)$ and $\sigma_8(z)$ gives rise to a new bias-independent variable $f\sigma_8$, that is, in fact, what is measured by galaxy surveys from redshift space distortions (RSDs):

$$f\sigma_8(a) = a\frac{\delta'_{\rm m}(a)}{\delta_{\rm m}(1)} \cdot \sigma_{8,0}. \tag{23}$$

This quantity can be directly measured from current and forthcoming galaxy surveys, and several compilations exist in the literature [14,27,28]. Thus, if a growth rate measurement has been obtained via a fiducial cosmology $H'(z)$, then the corresponding $f\sigma_8$ value is obtained for the true cosmology $H(z)$ via an Alcock-Paczynski-like correction [27,44,45]:

$$f\sigma_8(z) \simeq \frac{H(z)D_A(z)}{H'(z)D'_A(z)} f\sigma'_8(z). \tag{24}$$

Nonetheless, one of the main advantages of the $f\sigma_8$ growth rate is that it is a direct dynamic probe of gravity since, as can be seen from Eq. (16), the dependence on the gravitational theory appears explicitly via the normalized evolving Newton's constant $\mu(k,a)$ and indirectly via the Hubble parameter $H(a)$. On the other hand, one of its main weaknesses is also that the measurements, as currently made by the galaxy surveys, suffer from model dependence (typically assumed to be the ΛCDM model) and by the fact that the modeling of the nonlinear scales is phenomenological [22–26].

B. Modeling of the nonlinear scales

While this work aims to predict $f\sigma_8$ without assuming a specific model, for comparison with the machine learning approaches, we performed a maximum likelihood analysis for the effective field theory (EFT) power spectrum in redshift space in a later section. Here, we provide a brief review of this approach for completeness.

The power spectrum analysis needs to adopt a theoretical model to accurately account for the nonlinear effects in galaxy clustering (e.g., [26,46–56]). We introduce the nonlinear power spectrum, incorporating one-loop corrections, through the utilization of the EFT power spectrum in redshift space, described as [57]

$$\begin{aligned} P_s^{\rm EFT}(k,\mu) = {} & P_{\delta\delta}(k) + 2f_0\mu^2 P_{\delta\theta}(k) + f_0^2\mu^4 P_{\theta\theta}(k) \\ & + A^{\rm TNS}(k,\mu) + D(k,\mu) \\ & + (\alpha_0 + \alpha_2\mu^2 + \alpha_4\mu^4 + \alpha_6\mu^6)k^2 P_L(k) \\ & + \tilde{c}(f_0\sigma_v k\mu)^4 P_s^K(k,\mu) \\ & + P_{\rm shot}[\alpha_0^{\rm shot} + \alpha_2^{\rm shot}(k\mu)^2], \end{aligned} \tag{25}$$

where μ is the cosine of the angle between the line of sight, and f_0 is the growth rate at large scales.

The first and second lines of the equation represent the power spectrum in redshift space up to the one-loop order in perturbation theory. Here, $P_{\delta\delta}$ and $P_{\theta\theta}$ refer to the one-loop power spectra for the density and velocity fields in real space, respectively. The term $P_{\delta\theta}$ denotes the cross-power spectrum between these fields. Additionally, $A^{\rm TNS}(k,\mu)$ and $D(k,\mu)$ are correction terms derived from combinations of three and four fields (densities or velocities), respectively [51,54].

In the third line, $P_L(k)$ represents the linear matter power spectrum, and the EFT parameters α_0, α_2, α_4, and α_6 are introduced. This term accounts for small-scale variations in the behavior of large scales and the mapping between real and redshift space in the nonlinear regime. The forth line represents the finger-of-God effects [58], where σ_v is the velocity dispersion and $P_s^K(k,\mu)$ is the Kaiser power spectrum [56].

The last line accounts for noise effects. Here, $P_{\rm shot}$ denotes the Poisson shot noise, which is proportional to $1/n$, where n represents the number density of the tracer. The term $(k\mu)^2$ characterizes deviations from pure white noise [13,59,60].

III. THE QUIJOTE SIMULATIONS

The Quijote simulations, see Ref. [12], are a set of 44100 N-body simulations created via the TreePM code

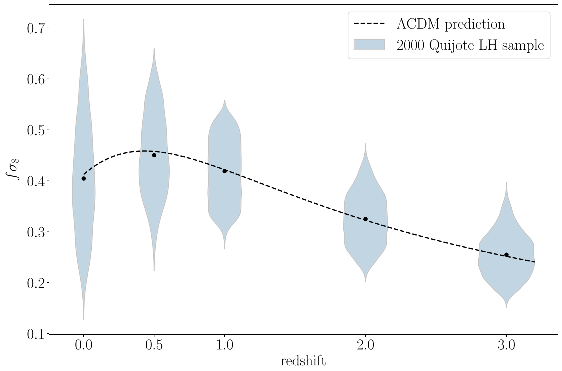

FIG. 1. The distribution of the $f\sigma_8$ parameter values for Latin-hypercube realizations of the Quijote simulations at their corresponding redshifts. Black dots represent the mean value at each redshift, and the dashed curve shows the ΛCDM prediction.

Gadget-III in boxes of sides of 1 Gpc/h, where h is the reduced Hubble constant. This work uses these simulations to train and test our CNN [61].

The Quijote simulations have 2000 realizations with different combinations of cosmological parameters $(\Omega_m, \Omega_b, h, n_s, \sigma_8)$, chosen by the Latin-hypercube sampling within the ranges of

$$\begin{aligned} \Omega_m &\in [0.1, 0.5], \\ \Omega_b &\in [0.03, 0.07], \\ h &\in [0.5, 0.9], \\ n_s &\in [0.8, 1.2], \\ \sigma_8 &\in [0.6, 1.0]. \end{aligned} \tag{26}$$

All those simulations have different seeds of the initial condition. A single simulation contains 512^3 dark matter particles and each realization has five snapshots at redshifts $z = [3.0, 2.0, 1.0, 0.5, 0.0]$. In Fig. 1, we show the $f\sigma_8$ distribution for 2000 realizations.

A. The halo catalog

The Quijote simulations have a halo catalogue built by a friends-of-friends algorithm, where each halo has at least 20 dark matter particles. Therefore, the lower limit of the halo mass is $\mathcal{O}(10^{12})$–$\mathcal{O}(10^{13})[M_\odot/h]$ (the value varies with the values of the cosmological parameters).

We use the snapshots of $z = [0.5, 1.0, 2.0, 3.0]$ from the Quijote simulations. Then, we assume the redshift halo distribution that roughly corresponds to the *Euclid* survey and randomly pick up halos such that the redshift distribution matches the relative numbers in Table I. This table shows the mean of the comoving number density of the halos in each redshift bin over 2000 realizations after after the selection process.

TABLE I. Specifications for the mock simulation. It includes the relative number, which represents the expected number of halos based on an observation akin to the *Euclid* survey, and n_{halo} denotes the total number of halos incorporated within the simulation box, after the random selection process.

z	Relative number	n_{halo} $[(\mathrm{Gpc}h^{-1})^{-3}]$
0.5	2	8.0×10^3
1.0	4	1.6×10^4
2.0	3	1.2×10^4
3.0	2	8.0×10^3

B. Data for machine learning

The architecture of our CNN will be discussed in Sec. IV A. For the training part, we use the dark matter or halo distribution, and we build images for our CNN as follows: first, we define the 40^3 grids in a simulation box, while this size of the grid corresponds to $k = 0.25\ h\,\mathrm{Mpc}^{-1}$ in Fourier space, and then we redistribute dark matter particles or halos to the cells by nearest gridding point. As a result, we get the 3D images whose size is 40^3, and each voxel value is the density of dark matter or halo in each cell. Note that this grid size is larger than the scale of the redshift distortion. In fact, we have tested our CNN using images in redshift space and found that the results do not change from those obtained using images in real space.

And then, for comparison, we train and test a neural network (NN) by the Legendre multipoles of power spectrum which is the two-point statistics in the redshift space defined as

$$P_l(k) = \frac{2l+1}{2} \int_{-1}^{1} P(k,\mu) L_l(\mu) d\mu, \tag{27}$$

where $P(k,\mu)$ is the anisotropic power spectrum in redshift space, and $L_l(\mu)$ is the Legendre polynomial of order l. In this work, we use the expansion coefficient $P_0(k)$, $P_2(k)$, and $P_4(k)$, which are already calculated for the dark matter distribution in the simulations we introduce and are publicly available. As the input to our NN, we use the $P_0(k)$, $P_2(k)$, and $P_4(k)$ for $0.089 < k[h\,\mathrm{Mpc}^{-1}] < 0.25$, linearly separated in 39 bins.

The Quijote simulations have 2000 realizations, so we use 1500 simulations as training data, 100 as validation data, and 400 as test data for both, the CNN and the NN.

IV. MACHINE LEARNING BASED COSMOLOGICAL PARAMETER INFERENCE

In this section, we describe the ML approach to extract cosmological parameters from the images of the density field or the measured power spectra using the Quijote simulation. Furthermore, we employ maximum likelihood analysis to estimate the value of $f\sigma_8$, which we compare

TABLE II. The predicted errors on the $f\sigma_8$ as derived by different methods and tracers. For the method, CNN, PS-ML, and likelihood stand for convolutional neural networks, the power spectrum analyzed by machine learning, and the maximum likelihood analysis, respectively. The values for "halo" are obtained from the halo data, resampled from the parent halo to match the redshift distribution of our simulated observation (see Table I) for CNN, while those in parenthesis are from the parent halo sample as a reference.

Method	CNN		PS-ML	Likelihood
Tracer	DM	Halo	DM	DM
$z = 0.5$	3.8	4.5 (3.9)	2.3	7.9
1.0	2.3	2.5 (2.2)	2.0	5.7
2.0	1.2	1.7 (1.1)	2.7	3.9
3.0	0.74	1.2 (1.2)	2.9	3.3

($\times 10^{-2}$).

with the CNN approach. We first provide a basic structure of the ML architecture, specifically optimized for analysing images and power spectrum, and the method of the maximum likelihood analysis. We then show the results of the comparison of different methodologies. We first compare the result of the CNN on the dark matter density field images to those on the halo density field images (Sec. IV C). This will directly compare the ability to constrain $f\sigma_8$ between dark matter and halo. We then compare the result of the CNN on the dark matter component with the ML-based dark matter power spectrum result and the maximum likelihood analysis (Sec. IV D).

The predicted errors on the $f\sigma_8$ quantity as derived by different methods are summarized in Table II. In what follows, we discuss in detail how the analysis is performed for each method and how we perform the measurements in each case.

A. ML architecture

This work uses the three-dimensional CNN to analyse the images and an NN for the power spectrum. We used the publicly available platform PyTorch [62] to construct our CNN and NN and the architecture based on Ref. [29]; however, note that some hyperparameters are different from those of that work, as the size of input data is different. In Tables III and IV, we show the architecture of our CNN and NN, while it should be noted that for the activation function, we apply the ReLU after each convolution layer and FullyConnected layer except for the last layer. Also, our CNN predicts the value of $f\sigma_8(z)$ from the $40 \times 40 \times 40$ image of dark matter or halo distribution introduced in Sec. III B.

TABLE III. Our CNN architecture. In all convolutional layers, stride = 1 and padding is not applied. Output map size corresponds to (height, width, depth, and channel). After each convolution layer and FullyConnected layer, except for the last layer, we apply the ReLU as the activation function. The total number of trainable parameters is 5,345,341.

	Layer	Output map size
1	Input	$40 \times 40 \times 40 \times 1$
2	$3 \times 3 \times 3$ convolution	$38 \times 38 \times 38 \times 2$
3	BatchNorm3d	$38 \times 38 \times 38 \times 2$
4	$2 \times 2 \times 2$ MaxPool	$19 \times 19 \times 19 \times 2$
5	$2 \times 2 \times 2$ convolution	$18 \times 18 \times 18 \times 64$
6	BatchNorm3D	$18 \times 18 \times 18 \times 64$
7	$2 \times 2 \times 2$ MaxPool	$9 \times 9 \times 9 \times 64$
8	$3 \times 3 \times 3$ convolution	$7 \times 7 \times 7 \times 64$
9	$3 \times 3 \times 3$ convolution	$5 \times 5 \times 5 \times 64$
10	$2 \times 2 \times 2$ convolution	$4 \times 4 \times 4 \times 128$
11	BatchNorm3d	$4 \times 4 \times 4 \times 128$
12	Flatten	$8192 (= 4^3 \times 128)$
13	FullyConnected	512
14	FullyConnected	256
15	FullyConnected	1

TABLE IV. Our NN architecture. After each FullyConnected layer, except for the last layer, we apply the dropout layer with rates of 0.1. The total number of trainable parameters is 848, 897.

	Layer	Output size
1	Input	3×39
2	FullyConnected	512
3	FullyConnected	512
4	FullyConnected	512
5	FullyConnected	512
6	FullyConnected	1

Here, we also use minibatch learning, and when we choose N_b as the batch size, we randomly divide the training data into groups, and each group has N_b training data. This group is called a minibatch, and then the average value of the loss function in the minibatch is used to update the trainable parameters. With trial and error, we determined that the optimal batch size is 16; however, if we make the batch size unity, the loss value does not converge because we use batch normalization in our CNN.

As our main loss function, we use the mean squared error (MSE)

$$\mathcal{L} = \frac{1}{N_b} \sum_{i=1}^{N_b} (y_i - \hat{y}_i)^2, \tag{28}$$

where N_b is the batch size, y_i is the predicted value of $f\sigma_8$ from our CNN for the ith data in the minibatch data, and $\hat{y}_i$ is the ground-true value, i.e., the correct $f\sigma_8$ value.

In updating the trainable parameters, we use the Adam optimizer [63], which is defined as `torch.optim.Adam()` in PyTorch. We use 5×10^{-7} and 0.1 as the value of lr and weight_decay, which are the arguments of torch.optim.Adam(), respectively. The batch normalization [64] is used in our CNN as we have found that this improves the learning efficiency. This is defined as torch.nn.BatchNorm3d() in PyTorch, and we use the default values for all parameters.

Our NN predicts the value of $f\sigma_8$ from the Legendre expanded power spectrum. We use the $P_0(k)$, $P_2(k)$, and $P_4(k)$ for $k \leq 0.25\,h\,\mathrm{Mpc}^{-1}$, so the size of input data to our NN is $3(P_0, P_2, \text{and}\,P_4) \times 39$ (the number of k bins). Furthermore, we apply the dropout layer with the rate of 0.1 after each FullyConnected layer, except for the sixth layer, see Table IV. As the loss function and the optimizer of the trainable parameters, we use the MSE and the Adam optimizer where $\mathrm{lr} = 5 \times 10^{-6}$ and weight_decay $= 0$. In addition, we apply $N_b = 16$ as the batch size in training.

B. Parameter inference by the maximum likelihood

For the comparison between machine learning and conventional approaches, we estimate $f\sigma_8$ using maximum likelihood for the dark matter density field. Here, we consider the likelihood below:

$$-2\ln L = \sum_{i,j}[P_i^{\text{model}} - P_i^{\text{data}}] \times \mathrm{Cov}_{i,j}^{-1} \times [P_j^{\text{model}} - P_j^{\text{data}}]. \tag{29}$$

Here, P_i^X, where X is either “model” or “data,” represents the concatenated Legendre multipoles $[P_0(k), P_2(k), P_4(k)]$ of the power spectrum as defined in Eq. (27). We restrict our analysis to $k < 0.2\,h\,\mathrm{Mpc}^{-1}$, which is justified by the maximum wave number applicable for the EFT power spectrum utilized [57]. Each multipole comprises 31 bins of the wave number, resulting in i ranging from 1 to 93 $(= 3 \times 31)$.

Also, P^{data} denotes the power spectrum derived from the Quijote Latin Hypercube (LH) simulation dataset, where cosmological parameters are sampled using Latin-Hypercube sampling. Cov represents the covariance matrix of P^{data}. To estimate the covariance matrix, we use 15,000 realizations from the “Fid” simulation dataset of Quijote, where the cosmological parameters are $[\Omega_m, \Omega_b, h, n_s, \sigma_8] = [0.3175, 0.049, 0.6711, 0.9624, 0.834]$. Each realization is generated with a different random seed for its initial condition, thus accounting for cosmic variance in the covariance estimation.

Finally, P_i^{model} corresponds to the power spectrum calculated following Eqs. (25) and (27). To compute it, we use FOLPS-nu [65], which computes the Legendre multipoles of the EFT power spectrum [57], and CLASS [66] for the linear power spectrum. Here, we focus on an unbiased tracer (dark matter). Specifically, we consider the five cosmological parameters [Ω_m, Ω_b, h, n_s, σ_8] and four nuisance parameters [α_0, α_2, α_0^{shot}, α_2^{shot}] in Eq. (25), setting both α_4 and α_6 to zero following [57]. We randomly select 400 realizations from the LH dataset, compute P^{data} for each realization, and fit the aforementioned $5+4$ parameters to minimize $-\ln L$. For the minimization of $-\ln L$ (equivalent to maximizing the likelihood), we use scipy.optimize.minimize(), implemented in SciPy [67]. Subsequently, we calculate $f\sigma_8$ from the fitted parameters in each realization following Eqs. (17)–(24).

C. CNN results on dark matter and halo

In this subsection, we compare the results from our CNN for the images of dark matter distribution and the halo distribution.

During the training phase, our CNN undergoes 100 epochs, each representing a complete iteration of the CNN learning process using the entire training dataset. Then, in the test phase, we use the CNN model, which minimizes the value of the loss function for the validation data.

In this work, the errors are estimated by the standard deviation of the vector of $\Delta f\sigma_{8,i} = (y_{\text{pred},i} - y_{\text{true},i})$, where $y_{\text{pred},i}$ and $y_{\text{true},i}$ are the CNN prediction and the ground truth for the ith test data, respectively.

To probe the validity of this estimation, we perform the following test:

(1) We make a 30×30 grid of parameters for $\Omega_m \in [0.1, 0.5]$ and $\sigma_8 \in [0.6, 1.0]$.
(2) For each redshift, we evaluate the $f\sigma_8$ in the whole grid, and then, we pick 400 grid points (the same number of the test data for our CNN) from them.
(3) We draw predictions for $f\sigma_8$ based on a normal distribution with mean given by the value at each point and with error σ. This is justified by considering a large number of simulations and the central limit theorem.
(4) We compare the predicted error from the standard deviation of $\Delta f\sigma_{8,i}$ vector with the fiducial error assumed in the previous step.

Figure 2 shows the relation between true and predicted $f\sigma_8$ for our CNN results, respectively for dark matter (blue) and halo (orange) test images. Figure 3 displays histograms of $\Delta f\sigma_8$ for the test data, allowing us to compare errors between the dark matter and halo images. As shown in Fig. 3, all histograms are centered around zero, indicating the effectiveness of our approach in constraining $f\sigma_8$.

Here, for comparison purposes, we use all the halos when building the images (without disregarding any halo). By comparing the results for dark matter and halo images, we find that the errors are mostly comparable across all redshift bins, while the results for the dark matter images are better at $z = 3.0$. The error decreases as the redshift increases for both dark matter and halo images. One possibility is that nonlinearities make it more challenging to extract information from the matter distribution at low

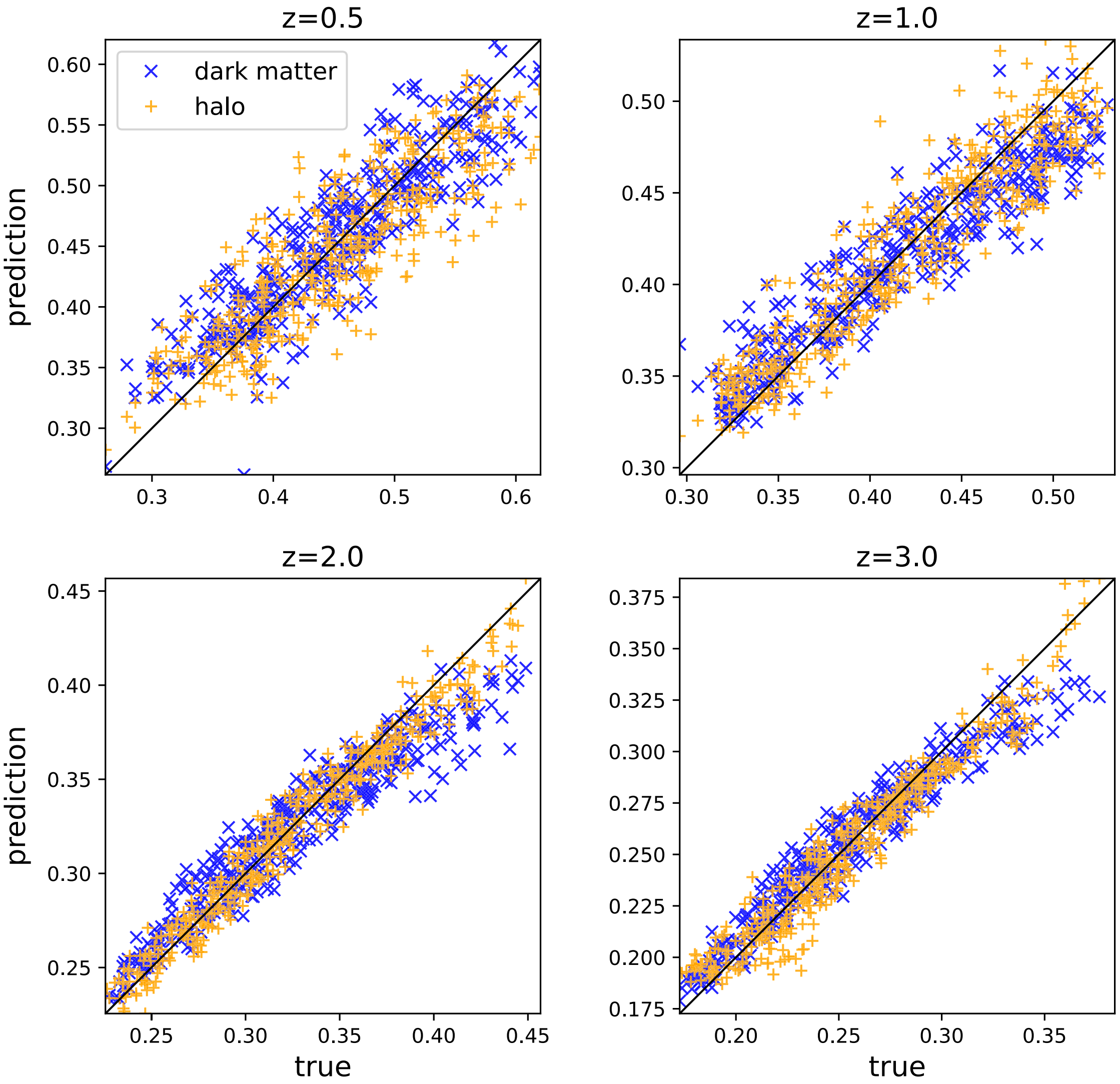

FIG. 2. The results of our CNN for dark matter images (blue) and halo images, including all halo (orange) for each redshift. The horizontal axis shows the true value of $f\sigma_8$, and the vertical axis shows our CNN's predicted value of $f\sigma_8$.

redshifts. At $z = 3.0$, the error for the halo images is larger than that for the dark matter and comparable to the error for $z = 2.0$ halo images. We can attribute this to shot noise because the number of halos in the simulation at $z = 3.0$ is less than one-tenth of the number at other redshifts when considering all halos.

As a side note, when we test our CNN with the dark matter images of 64^3 grids, corresponding to $k = 0.4\ h\,\mathrm{Mpc}^{-1}$, we observe slightly improved errors by approximately 15% across all the redshifts we consider. Given the configuration of the Quijote simulation we utilize, the Poisson shot noise dominates the dark matter power spectrum at $k = \mathcal{O}(1)\ h\,\mathrm{Mpc}^{-1}$. It is probable that the CNN performs well in analyzing these scales.

Next, we investigate the effect of the random selection of the halos. Instead of the images including all halos, we reduce the number of halos so that its redshift distribution follows a realistic observation (the ratio is shown in Table I) and use them to train and test our CNN. The results are shown in Fig. 4. Even when we use only randomly selected halos, we can see the redshift dependence of the error as in the all-halo case. The errors of the selected halo case are more significant than the ones for all the halos, but this is reasonable because the images of the selected halos lose the information compared to the images including all halos.

It is important to mention that, by assuming a redshift-halo distribution that matches the one of surveys like *Euclid*, our intention is to make an analysis with stage IV-like (realistic) data. However, to make a direct comparison with a real catalog, additional criteria must be addressed. An initial strategy could involve training a CNN to analyze observable features within a galaxy survey, such as a luminosity-limited galaxy sample, which inherently contains noise. Subsequently, the CNN could extract $f\sigma_8$ from these observations.

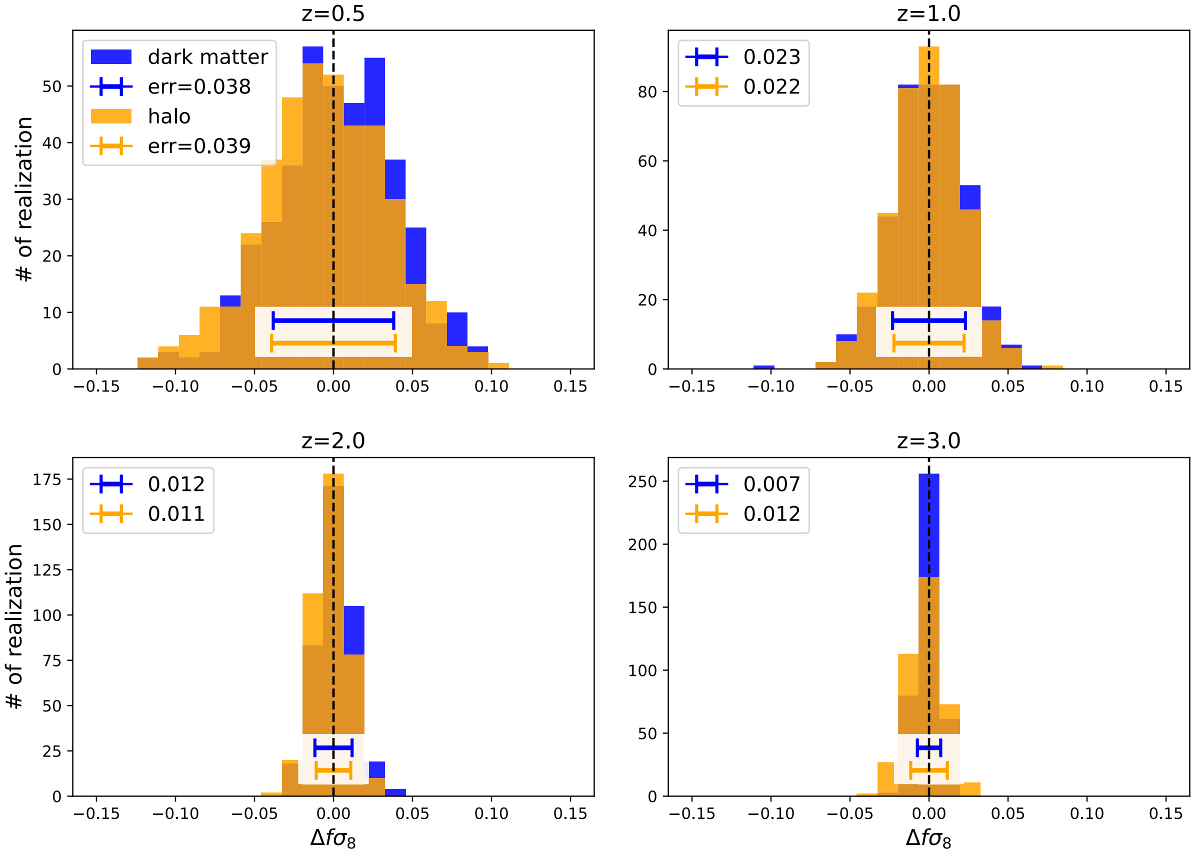

FIG. 3. The histograms of the CNN prediction for dark matter images (blue) and halo images (orange) for each redshift bin. The horizontal axis corresponds to the value of $\Delta f\sigma_8$ predicted by our CNN, and the vertical axis shows the number of test images for each $\Delta f\sigma_8$. The error bars show the standard deviation of $\Delta f\sigma_8$; its value is shown in the legend. It can be noticed that all histograms are centred around zero, indicating that our approach successfully constrains $f\sigma_8$.

In a simulation suite like Quijote, where the matter density varies, the minimum mass of halos fluctuates according to the cosmological parameters. Another approach could be to train on a halo catalog with a minimum luminosity matching the one expected from *Euclid*. These could be interesting ways of complementing the realistic-like catalog analysis, but this is beyond the scope of this work.

We additionally performed some tests in the CNN architecture by exploring different loss functions to corroborate our results. We studied the CNN behavior with MSE loss, MAE (mean absolute error) loss, Hubber loss, and the moments network loss (LFI—likelihood free inference) [68,69]. We observe a slight improvement in error with the LFI loss at low redshift. However, when evaluating the loss curve, we find that the MSE loss is more stable, so we chose to implement it for our final results. See the Appendix for more details.

D. ML-based power spectrum analysis on dark matter

Now, we compare the CNN results with the ones from the machine-learning-based power spectrum analysis (PS-ML) for dark matter. In this subsection, we use the NN whose architecture is defined in Table IV for the power spectrum analysis. Training and error evaluation processes are exactly the same for our CNN in Sec. IV C.

In Fig. 5, we compare the results obtained by dark matter images (blue, same as the one in Fig. 3) and PS-ML (green). We see that the constraints on $f\sigma_8$ from PS-ML are weaker at higher redshifts ($z \geq 2.0$) while it goes opposite at low redshift $z \leq 1.0$. The results look puzzling, but the constraining power may depend on the architecture of the CNN/NN, as we discussed in Sec. IV C. Therefore, it is not straightforward to interpret the results, but one possibility is that in the highly nonlinear regime, image recognition tends to fail to extract the characteristics of the data.

In Fig. 5, we also show the results of the maximum likelihood analysis for dark matter images (red). The standard deviation of $\Delta f\sigma_8$ is represented by the red error bar. We estimate the $f\sigma_8$ values for 400 realizations from the Quijote LH dataset following Sec. IV B. For each realization, we obtain the predicted $f\sigma_8$ value and evaluate $\Delta f\sigma_8$ in a manner similar to the ML analysis. It is evident that CNN and PS-ML outperform the likelihood analysis, indicating their ability to extract more information from the

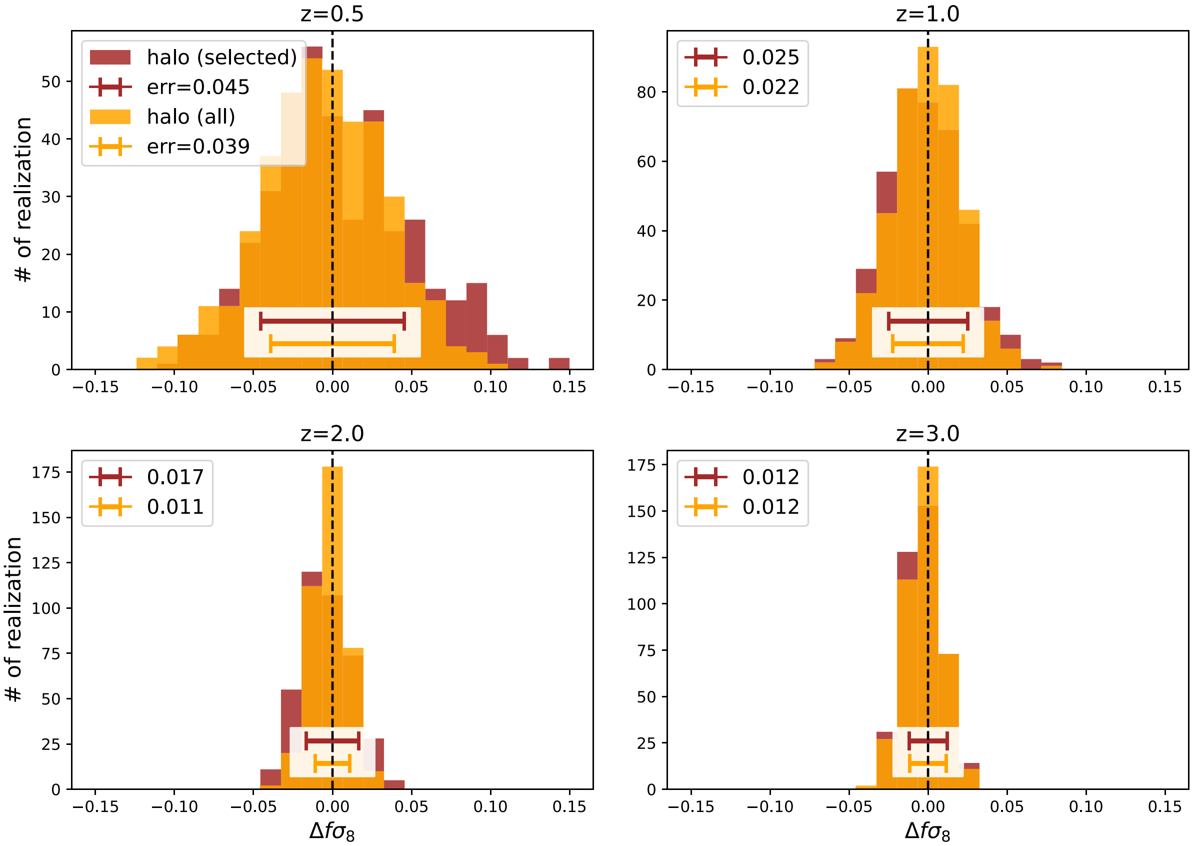

FIG. 4. The results of our CNN for images including all halos (orange) and randomly selected halos (brown) for each redshift bin. The horizontal axis corresponds to the value of $\Delta f\sigma_8$ predicted by our CNN, and the vertical axis shows the number of the test images for each $\Delta f\sigma_8$. The error bars show the standard deviation of $\Delta f\sigma_8$, and its value is shown in the legend.

dark matter density field than the modeled power spectrum in Eq. (25).

Additionally, we can see two features in the results obtained from the maximum likelihood. First, the maximum likelihood analysis shows redshift dependence similar to CNN. This dependence is likely caused by nonlinearities at lower redshifts. At $z = 3$, the maximum likelihood and the PS-ML show comparable results because the nonlinearity is weaker at higher redshifts. The PS-ML directly fits the observed power spectra for the simulation data, including the nonlinear effects, to derive the $f\sigma_8$ value, and the nonlinearity has less effect on the PS-ML compared to the maximum likelihood method. Second, the maximum likelihood analysis shows a bias of $\Delta f\sigma_8$ at lower redshifts. This bias may be caused by the model-dependent property of the analysis. On the other hand, CNN and PS-ML do not show such bias, and it is one of the advantages of the ML-based analysis.

E. Effect of the random seed

Finally, we discuss the effect of the random initial conditions. So far, we have used the Latin-hypercube Quijote simulations, which have different random seeds for the initial conditions. Therefore, the difference in these simulations is caused by the difference in its cosmological parameters and initial conditions. To discuss the effect of the second one, we use another dataset of Quijote simulations. The Quijote simulations have 15,000 realizations as “Fid” simulations for the same cosmological parameter set and different random seeds for their initial conditions.

First, we pick up 7000 realizations from the “Fid” simulation dataset, accessible on the San Diego Cluster hosting the Quijote simulations data. Then, we build the 3D images of randomly selected halo distributions from these data following Sec. III B. Then, we use our CNN model, which is already trained by the randomly selected halo images in Sec. IV C, and test it by the images from the “Fid” simulations. By doing this, we can evaluate the error only from random initial conditions. As a result, we find that the error for “Fid” realizations is $\mathcal{O}(10^{-3})$.

This is about one-fifth of the error in the results in the previous subsections, which should include both effects from the variations of cosmological parameters and the initial conditions. Therefore, we can conclude that the error in our CNN analysis is dominated by the variance of cosmological parameters arising from different cosmologies in each realization.

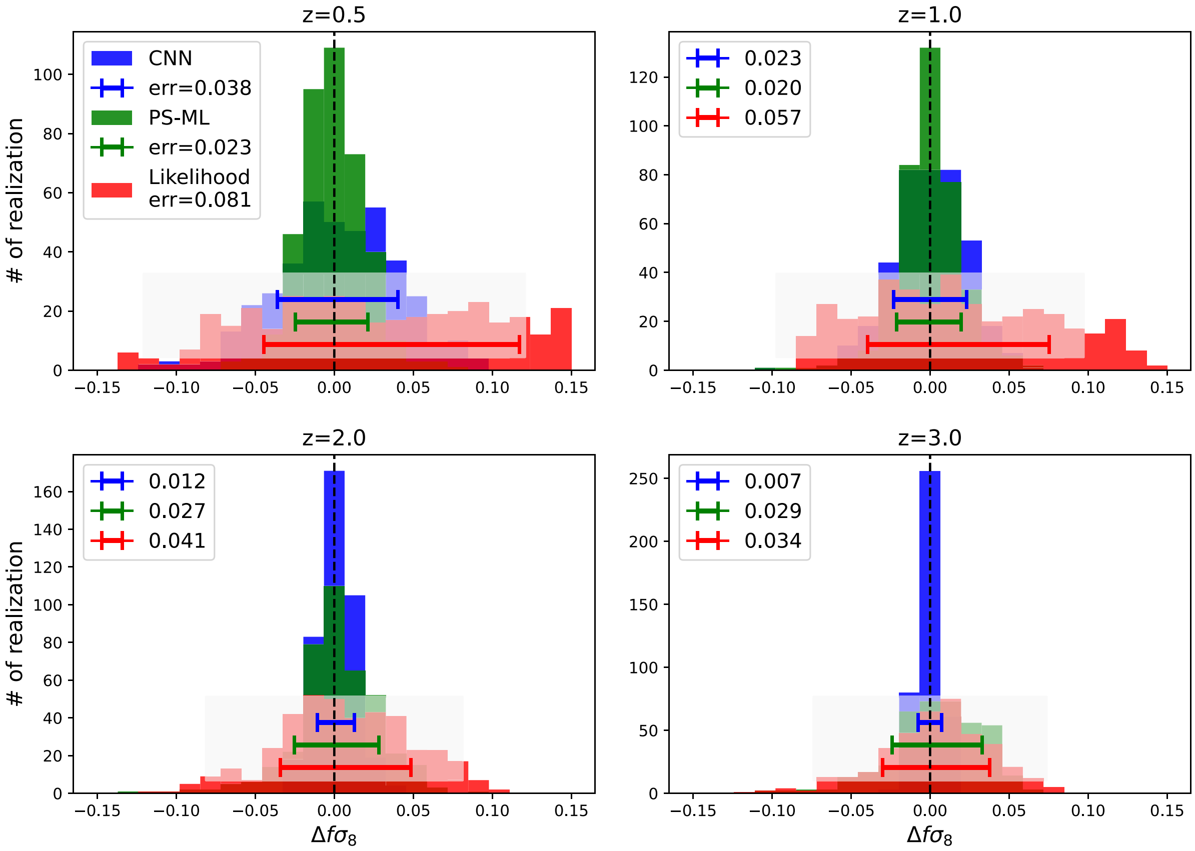

FIG. 5. The results of our CNN for dark matter images (blue), PS-ML (green), and likelihood (red) for each redshift. The horizontal axis corresponds to the value of $\Delta f\sigma_8$ predicted by our CNN, PS-ML, and likelihood, and the vertical axis shows the number of test images for each $\Delta f\sigma_8$. The error bars show the standard deviation of $\Delta f\sigma_8$ for each method. The values of these errors are shown in the legend.

V. DISCUSSIONS AND CONCLUSIONS

The growth rate of matter density perturbations, described by the bias-independent $f\sigma_8(z)$ quantity, is an important quantity in studying the large-scale structure of the Universe, as it allows us to probe the dynamic features of gravity. However, to extract $f\sigma_8(z)$ from observations made by galaxy surveys, two key assumptions have to be made: (i) a fiducial cosmological model and (ii) the modeling of the nonlinear part of the power spectrum. From these two assumptions, the latter is particularly insidious. Currently, only purely phenomenological models exist in the literature, thus potentially biasing the measurements of $f\sigma_8(z)$ in ways that are not easy to correct *a posteriori*.

The primary objective of this work, is to implement a ML approach based on training CNNs on N-body simulations to extract $f\sigma_8(z)$, without the reliance on any specific modeling of the nonlinear power spectrum. This could serve as an initial step towards the future development of a method for parameter inference in stage IV surveys, but for this, some considerations, like systematic and baryonic effects, need to be taken into account. In a sense, we perform a likelihood-free extraction of the growth rate, as the CNN requires neither a likelihood nor any modeling of the power spectrum, by leveraging the ability of the neural network architecture to predict the growth $f\sigma_8(z)$ from images of the dark matter or halo distributions build from the N-body simulations, as discussed in Sec. IV A. In addition, we compare the results of the ML approaches with those obtained by NN trained on the multipoles of the power spectrum and by the maximum likelihood analysis with the modeled power spectrum.

Specifically, our work followed a multipronged approach to find the optimal way to extract the growth via an ML architecture. The principal results and comparisons between the various approaches are shown in Figs. 2–5, where we showed scatter plots and histograms comparing the CNN dark matter and halo reconstructions, and the CNN dark matter and PS-ML reconstructions Table II shows the predicted errors from all approaches.

Overall, as can be seen in the figures above and Table II, we find that the ML architecture can predict quite accurately the value of the growth rate $f\sigma_8(z)$ (as seen in Fig. 3 where all histograms are centered around zero). At the same time, the reconstructed errors from the CNN are comparable within a factor of order unity between the different

ML approaches based on the DM or halo catalogs. Additionally, the ML approaches show smaller errors compared to likelihood analysis using modeled power spectra. However, we observe the redshift dependence of the CNN results, while the errors calculated by the ML-based power spectrum analysis do not largely depend on the redshift. To probe for the cause of this redshift behavior of the error from the CNN, we tested several possibilities, which, while they might not individually constitute the main reason, each may contribute to some extent.

First, one plausible reason for the redshift trend of the CNN results might be that at low redshifts, the enhanced nonlinearities might introduce more scatter in the observed values of $f\sigma_8$ by the CNN, something which might not be captured correctly by the PS-ML.

Second, another possible reason for the decreasing error of the CNN estimate might be because while the Quijote simulations are sampled well on the $(\Omega_m, \Omega_b, h, n_s, \sigma_8)$ from the Latin-Hypercube, this is not the case for $f\sigma_8(z)$, as already seen in Fig. 1. This would, in effect, introduce a prior on $f\sigma_8(z)$ that decreases/tightens with redshift. See the blue "blobs" in Fig. 1 that indicate the scatter of $f\sigma_8(z)$, thus artificially training the CNN to expect a reducing value on the error with redshift (since the possible scatter of the parameters is limited by the prior). However, we should note that the width of the prior is generally greater than the error of the CNN. We conducted a test by artificially narrowing the range of the prior (making the $f\sigma_8(z)$ width, equal for all the considered redshift bins). After training and testing our architecture, no significant impact on the error estimates was found.

To investigate the correct sampling of $f\sigma_8(z)$ in the Quijote simulations compared to $(\Omega_m, \Omega_b, h, n_s, \sigma_8)$, two approaches could be considered. First, increasing the number of simulations used in training could broaden the prior and address the observed redshift dependence on the error in Fig. 1. Alternatively, directly sampling within the $f\sigma_8(z)$ space, possibly facilitated by an N-body emulator, could be explored. This would require additional N-body simulations to potentially enhance resolution. However, due to the limited availability of Quijote simulations, leveraging an emulator to artificially augment the dataset is proposed as a means to address this limitation and potentially mitigate any observed trends in $f\sigma_8$. Nonetheless, conducting these additional tests is deemed beyond the scope of this study as we primarily aim to highlight our methodology.

Third, the training/testing approach of the CNN also has some stochasticity/randomness as different runs produce different errors. Thus, we tested this by running the testing phase with different random seeds and calculated the mean and standard deviation of the CNN error. In this case, we have found that the mean error is consistent with the values in Table II and the standard deviation of the error (i.e., the "error of the error") is much smaller. Thus, this reason could only contribute to a small extent to the redshift behavior of the CNN error.

Finally, we also performed several more tests; for example, we examined the effect of the random selection of the halos by making images that have a part of halos to reproduce the relative number shown in Table I and then used them to train and test our CNN. Doing that, we found the same redshift dependence of the error as in the previous case. Second, we also explored different loss functions, including the MSE, MAE, Hubber Loss, and LFI loss functions. Overall, we found a similar decrease in the error with redshift. Third, we investigated the effect of the random initial conditions by using another dataset of Quijote simulations. Doing so, we find that the error contribution is only one-fifth of the total error. Thus, we conclude that the difference in the cosmological parameters dominates the error in our analysis.

We successfully trained and validated CNNs to predict $f\sigma_8(z)$, using various input data sources, such as dark matter density fields and halo catalogs. But the applicability of the trained CNNs is somehow limited when talking about its implementation to real data. Therefore, this could be taken only as a first step for developing ML-based methods for cosmological parameter inference from observations. This approach may require additional factors into consideration, including the systematic and baryonic effects, as well as the fact that in real galaxy surveys, the positions of halos are known within a certain degree of uncertainty, so the measurements of $f\sigma_8$ in these surveys carry uncertainties proportional to their inherent noise characteristics. There were other attempts to perform parameter inference from simulations, for instance [70], where Ω_m and σ_8 are inferred directly from the power spectrum multipoles, whereas in this study we employ the raw image of the density field. Certainly, if the Legendre polynomial power spectrum decomposition were available without the bias term, one could compute σ_8 directly. Furthermore, in the presence of a dark matter density field, it is feasible to integrate the variance of the matter density within 8 Mpc shells to derive σ_8. For example see [71,72] where parameter inference is carried out at the density field level, thus requiring forward modeling to predict the field value, followed by a pixel-by-pixel comparison to determine the likelihood. Our approach does not demand a pixel-by-pixel comparison, since it operates within the latent space. This is a more generic approach compared to the field-level inference, but it needs a more comprehensive study to understand how various parameters (such as minimum halo mass, friend-of-friends method), affect the results and potentially bias the constraints on $f\sigma_8$, therefore this represents a limitation for our current methodology.

For comparison, we also carried out an EFT approach and we found that the CNN performs better when inferring $f\sigma_8$. This outperformance may be attributed to the fact that the

EFT is limited to wave numbers below $k < 0.2\ h\,\mathrm{Mpc}^{-1}$, whereas the CNN extends its scale to values up to $k < 0.25\ h\,\mathrm{Mpc}^{-1}$. The broader range of wave numbers accessible to the CNN likely provides it with more information, enhancing its ability to infer $f\sigma_8$.

We leave for future work the extension of this analysis to models beyond the ΛCDM model, i.e., to eliminate the first assumption of current growth measurements as mentioned earlier since that requires N-body simulations for modified gravity. Also, we leave a more detailed comparison between the CNN and the conventional approach such as the Fisher matrix approach for future work, as it would require significant theoretical modifications on the nonlinear part for the latter and more tests on the CNN architecture to investigate the effect on low redshifts. We also leave for future work the possible development of ML-based methods for parameter inference with real data.

Finally and to conclude, given that in a few years, the forthcoming galaxy surveys will provide a plethora of high-quality data related to the large-scale structure of the Universe, novel ways to analyze these data will be required to minimize the theoretical errors emanating from assumptions, just as the nonlinear modeling at small scales. Here, we provided the first step in this direction, but more work will be required to bridge the gap between the theory and the actual data.

The numerical codes used in this analysis will be available upon publication [73].

ACKNOWLEDGMENTS

We acknowledge the i-LINK 2021 Grant No. LINKA20416 by the Spanish National Research Council (CSIC), which initiated and facilitated this collaboration. We are also particularly indebted first and foremost to D. Sapone, but also to C. Baugh, C. Carbonne, S. Casas, J. García-Bellido, K. Ichiki, E. Komatsu, E. Majerotto, V. Pettorino, A. Pourtsidou, F. Villaescusa-Navarro, and T. S. Yamamoto for useful discussions. I. O., S. K., and S. N. acknowledge support from the research Project No. PID2021-123012NB-C43 and the Spanish Research Agency (Agencia Estatal de Investigación) via the Grant IFT Centro de Excelencia Severo Ochoa No. CEX2020-001007-S, funded by MCIN/AEI/10.13039/501100011033. A. J. N. acknowledges JSPS KAKENHI, Grant-in-Aid for Transformative Research Areas, 21H05454 and Grant-in-Aid for Scientific Research (C), 21K03625. S. K. is supported by the Spanish Atracción de Talento Contract No. 2019-T1/TIC-13177 granted by Comunidad de Madrid, the I + D Grant No. PID2020-118159GA-C42 funded by MCIN/AEI/10.13039/501100011033, and Japan Society for the Promotion of Science (JSPS) KAKENHI Grants No. 20H01899, No. 20H05853, and No. 23H00110. I. O. is supported by the fellowship LCF/BQ/DI22/11940033 from "la Caixa" Foundation (ID 100010434) and by a Graduate Fellowship at Residencia de Estudiantes supported by Madrid City Council (Spain), 2022–2023. K. M. would like to thank the "Nagoya University Interdisciplinary Frontier Fellowship" supported by Nagoya University and JST, the establishment of university fellowships towards the creation of science technology innovation, Grant No. JPMJFS2120. Finally, the authors acknowledge use of the Boltzmann code CLASS [66,74] and the Python packages SciPy [67], NumPy [75] and Matplotlib [76].

APPENDIX: LOSS FUNCTIONS TESTS

The MSE loss calculates the mean squared error between the CNN estimates and the true values and is set up to find an approximation of the marginal posterior mean. In general, the marginal posterior describes the probability that a simulation (and its array of clustering measurements) were created with a particular combination of parameters θ_i.

The advantage of LFI with respect to the other loss functions is that it achieves a better convergence between the CNN output and both the mean μ_i and standard deviation σ_i of the marginal posterior parameters $\theta_{i,j}$. LFI eliminates the influence of the overall scale in the spread of a parameter by incorporating logarithms. This approach assigns weights based on inverse variance to the gradients of the various terms, in contrast to the MSE loss. Consequently, it optimizes the combination of gradients to account for each term's importance. According to the mentioned arguments, LFI loss is defined by [77]

$$\mathcal{L}_{\mathrm{LFI}} = \log\left(\sum_{j\in\mathrm{batch}}(\theta_j-\mu_j)^2\right) + \log\left(\sum_{j\in\mathrm{batch}}((\theta_j-\mu_j)^2-\sigma_j^2)^2\right). \tag{A1}$$

The input parameter of each training and test procedure was $f\sigma_8$. This particular loss function takes the logarithm of the squared difference between the true and predicted values of this parameter within a batch j of the simulation, taking into account the standard deviation in the second

TABLE V. Predicted errors for each redshift snapshot using MSE, MAE, Hubber loss, and the moments network (LFI) loss functions in the CNN architecture.

Redshift	MSE loss	MAE loss	Huber loss	LFI loss
0.5	3.8	4.1	3.9	3.3
1.0	2.3	3.1	2.9	2.3
2.0	1.2	1.5	1.8	1.7
3.0	0.74	1.0	1.1	1.3

($\times 10^{-2}$).

term. As a matter of interest, we found a similar outcome with all the considered loss functions—an error decrease for larger redshifts. It is worth mentioning that LFI performs better, as expected. In Table V, we show the results only for the MSE and LFI loss functions for simplicity, where we see that the LFI improves slightly the error estimates compared to the previously considered MSE loss function.

It is worth mentioning that the slight improvement found at low redshift, achieved by the LFI loss, had a cost in stability, as mentioned before. This is why we chose the most stable method, MSE loss, for the final results.

[1] T. Padmanabhan, Phys. Rep. **380**, 235 (2003).

[2] L. Amendola, V. Marra, and M. Quartin, Mon. Not. R. Astron. Soc. **430**, 1867 (2013).

[3] N. Aghanim *et al.* (Planck Collaboration), Astron. Astrophys. **641**, A6 (2020); **652**, C4(E) (2021).

[4] T. M. C. Abbott *et al.* (DES Collaboration), Phys. Rev. D **105**, 023520 (2022).

[5] T. Kacprzak and J. Fluri, arXiv:2203.09616.

[6] R. Laureijs *et al.* (EUCLID Collaboration), arXiv:1110.3193.

[7] P. A. Abell *et al.* (LSST Science, LSST Project Collaborations), arXiv:0912.0201.

[8] A. Aghamousa *et al.* (DESI Collaboration), arXiv:1611.00036.

[9] A. Joyce, L. Lombriser, and F. Schmidt, Annu. Rev. Nucl. Part. Sci. **66**, 95 (2016).

[10] V. Motta, M. A. García-Aspeitia, A. Hernández-Almada, J. Magaña, and T. Verdugo, Universe **7**, 163 (2021).

[11] A. Lazanu, J. Cosmol. Astropart. Phys. 09 (2021) 039.

[12] F. Villaescusa-Navarro, C. Hahn, E. Massara, A. Banerjee, A. M. Delgado, D. K. Ramanah, T. Charnock, E. Giusarma, Y. Li, E. Allys *et al.*, Astrophys. J. Suppl. Ser. **250**, 2 (2020).

[13] V. Desjacques, D. Jeong, and F. Schmidt, Phys. Rep. **733**, 1 (2018).

[14] B. Sagredo, S. Nesseris, and D. Sapone, Phys. Rev. D **98**, 083543 (2018).

[15] Y.-S. Song and W. J. Percival, J. Cosmol. Astropart. Phys. 10 (2009) 004.

[16] J. J. M. Carrasco, S. Foreman, D. Green, and L. Senatore, J. Cosmol. Astropart. Phys. 07 (2014) 057.

[17] P. Carrilho, C. Moretti, and A. Pourtsidou, J. Cosmol. Astropart. Phys. 01 (2023) 028.

[18] H. A. Winther, S. Casas, M. Baldi, K. Koyama, B. Li, L. Lombriser, and G.-B. Zhao, Phys. Rev. D **100**, 123540 (2019).

[19] N. Ramachandra, G. Valogiannis, M. Ishak, and K. Heitmann (LSST Dark Energy Science Collaboration), Phys. Rev. D **103**, 123525 (2021).

[20] G. Brando, B. Fiorini, K. Koyama, and H. A. Winther, J. Cosmol. Astropart. Phys. 09 (2022) 051.

[21] C. Contreras *et al.* (WiggleZ Collaboration), Mon. Not. R. Astron. Soc. **430**, 924 (2013).

[22] R. Scoccimarro and J. Frieman, Astrophys. J. Suppl. Ser. **105**, 37 (1996).

[23] F. Bernardeau, N. Van de Rijt, and F. Vernizzi, Phys. Rev. D **85**, 063509 (2012).

[24] T. Baldauf, M. Mirbabayi, M. Simonović, and M. Zaldarriaga, Phys. Rev. D **92**, 043514 (2015).

[25] S. Casas, M. Kunz, M. Martinelli, and V. Pettorino, Phys. Dark Universe **18**, 73 (2017).

[26] A. Blanchard *et al.* (Euclid Collaboration), Astron. Astrophys. **642**, A191 (2020).

[27] S. Nesseris, G. Pantazis, and L. Perivolaropoulos, Phys. Rev. D **96**, 023542 (2017).

[28] L. Kazantzidis, L. Perivolaropoulos, and F. Skara, Phys. Rev. D **99**, 063537 (2019).

[29] A. Lazanu, J. Cosmol. Astropart. Phys. 09 (2021) 039.

[30] Z. Wu, L. Xiao, X. Xiao, J. Wang, X. Kang, Y. Wang, X. Wang, L. Zhang, and X.-D. Li, arXiv:2301.04586.

[31] F. Qin, D. Parkinson, S. E. Hong, and C. G. Sabiu, arXiv:2302.02087.

[32] C.-P. Ma and E. Bertschinger, arXiv:astro-ph/9506072.

[33] F. Bernardeau, S. Colombi, E. Gaztanaga, and R. Scoccimarro, Phys. Rep. **367**, 1 (2002).

[34] S. Nesseris, Universe **9**, 13 (2022).

[35] H. Kurki-Suonio, unpublished lecture notes available: https://www.mv.helsinki.fi/home/hkurkisu/cpt/ (referenced 23 October 2005) (2005).

[36] S. Tsujikawa, Phys. Rev. D **76**, 023514 (2007).

[37] T. Padmanabhan, Phys. Rep. **380**, 235 (2003).

[38] V. Silveira and I. Waga, Phys. Rev. D **50**, 4890 (1994).

[39] W. J. Percival, Astron. Astrophys. **443**, 819 (2005).

[40] A. Bueno belloso, J. Garcia-Bellido, and D. Sapone, J. Cosmol. Astropart. Phys. 10 (2011) 010.

[41] In the SciPy Python module, the Gauss hypergeometric function ${}_2F_1$ is implemented as scipy.special.hyp2fi(a,b,c,z).

[42] X.-D. Li, H. Miao, X. Wang, X. Zhang, F. Fang, X. Luo, Q.-G. Huang, and M. Li, Astrophys. J. **875**, 92 (2019).

[43] X.-D. Li, C. G. Sabiu, C. Park, Y. Wang, G.-b. Zhao, H. Park, A. Shafieloo, J. Kim, and S. E. Hong, Astrophys. J. **856**, 88 (2018).

[44] L. Kazantzidis and L. Perivolaropoulos, Phys. Rev. D **97**, 103503 (2018).

[45] E. Macaulay, I. K. Wehus, and H. K. Eriksen, Phys. Rev. Lett. **111**, 161301 (2013).

[46] M. Tegmark, Phys. Rev. Lett. **79**, 3806 (1997).

[47] H.-J. Seo and D. J. Eisenstein, Astrophys. J. **598**, 720 (2003).

[48] S. Yahia-Cherif, A. Blanchard, S. Camera, S. Casas, S. Ilić, K. Markovic, A. Pourtsidou, Z. Sakr, D. Sapone, and I. Tutusaus, Astron. Astrophys. **649**, A52 (2021).

[49] S.-F. Chen, Z. Vlah, E. Castorina, and M. White, J. Cosmol. Astropart. Phys. 03 (2021) 100.

[50] A. Aviles, G. Valogiannis, M. A. Rodriguez-Meza, J. L. Cervantes-Cota, B. Li, and R. Bean, J. Cosmol. Astropart. Phys. 04 (2021) 039.

[51] A. Aviles, A. Banerjee, G. Niz, and Z. Slepian, J. Cosmol. Astropart. Phys. 11 (2021) 028.

[52] Z. Vlah and M. White, J. Cosmol. Astropart. Phys. 03 (2019) 007.

[53] A. Perko, L. Senatore, E. Jennings, and R. H. Wechsler, arXiv:1610.09321.

[54] A. Taruya, T. Nishimichi, and S. Saito, Phys. Rev. D **82**, 063522 (2010).

[55] R. Scoccimarro, Phys. Rev. D **70**, 083007 (2004).

[56] N. Kaiser, Astrophys. J. Lett. **284**, L9 (1984).

[57] H. E. Noriega, A. Aviles, S. Fromenteau, and M. Vargas-Magaña, J. Cosmol. Astropart. Phys. 11 (2022) 038.

[58] M. M. Ivanov, M. Simonović, and M. Zaldarriaga, J. Cosmol. Astropart. Phys. 05 (2020) 042.

[59] P. McDonald and A. Roy, J. Cosmol. Astropart. Phys. 08 (2009) 020.

[60] S.-F. Chen, Z. Vlah, and M. White, J. Cosmol. Astropart. Phys. 07 (2020) 062.

[61] The detailed document of the Quijote simulations can be accessed here. The simulations data can be accessed through `binder`, which can be accessed from here.

[62] A. Paszke *et al.*, arXiv:1912.01703.

[63] D. P. Kingma and J. Ba, arXiv:1412.6980.

[64] S. Ioffe and C. Szegedy, arXiv:1502.03167.

[65] https://github.com/henoriega/FOLPS-nu.

[66] J. Lesgourgues, arXiv:1104.2932.

[67] P. Virtanen *et al.* (SciPy 1.0 Contributors), Nat. Methods **17**, 261 (2020).

[68] F. Villaescusa-Navarro, S. Genel, D. Angles-Alcazar, L. Thiele, R. Dave, D. Narayanan, A. Nicola, Y. Li, P. Villanueva-Domingo, B. Wandelt *et al.*, Astrophys. J. Suppl. Ser. **259**, 61 (2022).

[69] N. Jeffrey and B. D. Wandelt, arXiv:2011.05991.

[70] C. Modi, S. Pandey, M. Ho, C. Hahn, B. R. Blancard, and B. Wandelt, arXiv:2309.15071.

[71] S. Stopyra, H. V. Peiris, A. Pontzen, J. Jasche, and G. Lavaux, Mon. Not. R. Astron. Soc. **527**, 1244 (2024).

[72] A. Junzhe Zhou, X. Li, S. Dodelson, and R. Mandelbaum, arXiv:2312.08934.

[73] K. Murakami, fsigma8_ML 1.0 (2023) https://github.com/murakoya/fsigma8_ML

[74] D. Blas, J. Lesgourgues, and T. Tram, J. Cosmol. Astropart. Phys. 07 (2011) 034.

[75] C. R. Harris *et al.*, Nature (London) **585**, 357 (2020).

[76] J. D. Hunter, Comput. Sci. Eng. **9**, 90 (2007).

[77] L. A. Perez, S. Genel, F. Villaescusa-Navarro, R. S. Somerville, A. Gabrielpillai, D. Anglés-Alcázar, B. D. Wandelt, and L. A. Yung, Astrophys. J. **954**, 11 (2023).