Monthly Notices of the ROYAL ASTRONOMICAL SOCIETY

MNRAS **531,** 4203–4211 (2024) https://doi.org/10.1093/mnras/stae1261
Advance Access publication 2024 May 15

Fast emulation of two-point angular statistics for photometric galaxy surveys

Marco Bonici,[1]★ Luca Biggio,[2] Carmelita Carbone[1] and Luigi Guzzo[3,4,5]

[1] *INAF – IASF Milano, Via Alfonso Corti 12, I-20133 Milano, Italy*
[2] *Data Analytics Lab, Institute of Machine Learning, Department of Computer Science, Universitätstrasse 6, 8006 Zürich, Switzerland*
[3] *Dipartimento di Fisica 'Aldo Pontremoli', Università degli Studi di Milano, Sez. di Milano, Via Celoria 16, I-20133 Milano, Italy*
[4] *INFN, Sez. di Milano, Via Celoria 16, I-20133 Milano, Italy*
[5] *INAF, Osservatorio Astronomico di Brera, Via Brera, 28, I-20121 Milano, Italy*

Accepted 2024 April 22. Received 2024 April 21; in original form 2022 July 5

ABSTRACT

We develop a set of machine-learning-based cosmological emulators, to obtain fast model predictions for the $C(\ell)$ angular power spectrum coefficients, characterizing tomographic observations of galaxy clustering and weak gravitational lensing from multiband photometric surveys (and their cross-correlation). A set of neural networks are trained to map cosmological parameters into the coefficients, achieving, with respect to standard Boltzmann solvers, a speed-up of $\mathcal{O}(10^3)$ in computing the required statistics for a given set of cosmological parameters, with an accuracy better than 0.175 per cent (<0.1 per cent for the weak lensing case). This corresponds to $\lesssim 2$ per cent of the statistical error bars expected from a typical Stage IV photometric surveys. Such overall improvement in speed and accuracy is obtained through (i) a specific pre-processing optimization, ahead of the training phase, and (ii) an effective neural network architecture. Compared to previous implementations in the literature, we achieve an improvement of a factor of 5 in terms of accuracy, while training a considerably lower amount of neural networks. This results in a cheaper training procedure and a higher computational performance. Finally, we show that our emulators can recover unbiased posteriors when analysing synthetic Stage-IV galaxy survey data sets.

Key words: methods: data analysis – methods: statistical – cosmological parameters – large-scale structure of Universe.

1 INTRODUCTION

The golden age of cosmology is continuing. The discovery of the accelerated expansion of the universe (Riess et al. 1998; Perlmutter et al. 1999) led to the consolidation of the lambda-cold dark matter (ΛCDM) model, which can now be convincingly defined as the standard model of cosmology.

We are currently witnessing a period of fast developments, from the theoretical and observational points of view, aimed at understanding its details and ingredients. In fact, the standard model is capable to successfully predict virtually all early and late universe observations to high accuracy using a restricted number of parameters, although some residual, perhaps telltale, tensions should not be overlooked (e.g. Bernal, Verde & Riess 2016). However, the very nature of this model opens a number of fundamental questions which await a satisfactory answer, as in particular the nature of its dark sector components, Dark Matter and Dark Energy.

Together with cosmic microwave background (CMB) anisotropy observations (Aghanim et al. 2020), one of the key pillars supporting the standard model is represented by measurements derived from galaxy surveys. On the one hand, statistical measurements of galaxy clustering (GC hereafter) in 3D (when spectroscopic redshifts are available) or 2D (i.e. in projection on the sky) provide us with an indirect probe (through luminous tracers) of the distribution of matter density fluctuations on different scales and at different epochs (Feldman, Kaiser & Peacock 1994; Yamamoto et al. 2006). On the other hand, photometric imaging surveys allow us, under certain conditions, to measure galaxy shapes and use their shear, due to weak gravitational lensing (WL hereafter) by foreground galaxies, to assess directly the matter density field, providing us with the ability to perform a tomography of the large-scale matter distribution in the Universe (Hu 1999). Combinations of photometric 2D-GC angular correlation function, WL correlation function, and GC-WL cross-correlation between galaxy positions and shapes (3×2pt statistic hereafter), together with the spectroscopic 3D-GC, will enable unprecedented precision and accuracy to be achieved in the estimate of cosmological parameters.

We are on the verge of a new era for galaxy surveys, with the next generation of experiments that aim at solving the mysteries of the standard cosmological model. This is the case, in particular, of the Dark Energy Spectroscopic Instrument (DESI) ground-based spectroscopic survey (Abbott et al. 2018b), recently started at Kitt Peak National Observatory (KPNO), and the European Space Agency (ESA) *Euclid* mission (Laureijs et al. 2011), a combined imaging and spectroscopic experiment, currently scheduled for launch in 2023. The multiband, multi-epoch photometric Large Synoptic Survey Telescope (LSST) survey by the Vera C. Rubin telescope

★ E-mail: marco.bonici@inaf.it

(Weltman et al. 2020), and the future *Nancy Roman Space Telescope* (Dore et al. 2019), will provide further unprecedented photometric and spectroscopic galaxy data, contributing to the accumulation of an enormous amount of information. The *Euclid* mission, in particular, will play a special role in this respect, with its ability to simultaneously deliver measurements of both spectroscopic and photometric GC together with WL and (further additional probes, like e.g. galaxy clusters – see e.g. Blanchard et al. 2020).

Such an avalanche of new experimental data, naturally calls for a comparable quantum leap in the efficiency of our analysis tools. Each of these new data sets will be massive on its own. Additionally, the combination and cross-correlation of different probes (GC and WL, but also CMB anisotropies, void statistics, clusters, etc.) will leverage their information content, while providing firmer control over systematic errors. Besides the theoretical challenge of accurately modelling the data down to small, non-linear scales (Carrasco, Hertzberg & Senatore 2012; Cataneo et al. 2019; Angulo et al. 2021; Knabenhans et al. 2021), we will be facing a serious computational challenge: cosmological analyses are based on Markov Chain Monte Carlo (MCMC) techniques, which require theoretical predictions to be computed as many as 10^4 to even 10^6 times, with the execution of so-called Boltzmann solvers at each step of the chains. The most widely used of such codes are CAMB (Lewis, Challinor & Lasenby 2000) and CLASS (Lesgourgues 2011), which compute the evolution of linear perturbations for a w_0w_aCDM model in about ≈ 0.5 s (7 s when massive neutrinos are included) on a standard laptop.[1] Such basic figures clearly show that performing a full analysis of cosmological data sets requires either a very long time or the access to high-performance computing facilities.

To overcome this issue, codes called emulators, capable of reproducing the results of Boltzmann solvers, but interpolating from a coarser grid of 'pivot' models, have been recently developed, achieving significant speed-up. The first of such codes, based on polynomial regression, were used in the analysis of CMB data (Jimenez et al. 2004; Fendt & Wandelt 2006). A different approach was taken in Albers et al. (2019), where a neural network was trained to replace the most expensive parts of the CLASS Boltzmann solver.

More recently, emulators were developed to produce model predictions for large-scale structure two-point statistics (Manrique-Yus & Sellentin 2020; Mootoovaloo et al. 2020, 2022; Aricò, Angulo & Zennaro 2021; Spurio Mancini et al. 2022). These codes allow us to emulate either the spatial 3D matter power spectrum, $P_{\rm mm}(k)$, in real space, or the angular power spectrum, $C(\ell)$, that is, the coefficients of the spherical harmonic decomposition of the 2D correlation function, which quantifies density fluctuations as a function of the angular scale alone. The latter is customary when describing CMB anisotropies, weak lensing measurements or any statistics measured in projection on the sky, as also 2D GC within 'tomographic slices'.

In this paper, we describe the development of a set of three new emulators to derive fast model predictions for the $C(\ell)$ associated to three different observables. In particular, we focus on the 3×2pt statistics associated to two of the main cosmological probes of next generation surveys, GC and WL, together with their cross-correlation. Specifically, the cross-correlation of the tiny and coherent image distortions of background galaxies with the angular 2D galaxy distribution is of special interest and will be among the key observables of, for instance, the *Euclid* and LSST galaxy surveys.

We adopt a machine-learning approach, building and training a set of neural networks to map cosmological parameters into the 3×2pt angular correlation coefficients $C(\ell)$. Neural networks are becoming more and more popular in many areas of science and engineering. Much of their success stems from their universal approximation properties, that is, their ability to approximate any continuous function, provided enough data and an appropriate number of hidden layers (Cybenko 1989; Hornik, Stinchcombe & White 1989) are combined. Recent years have witnessed the emergence of powerful neural networks in the form of different architectures tailored to the specific task being performed. Such models are typically built by stacking multiple layers of neurons, the so-called deep neural networks.

The main advantage of a machine-learning approach to cosmological inference is the significant speed-up of the full analysis. In fact, concerning the computation of $P_{\rm mm}(k)$, in the standard approach the hierarchy of coupled Einstein–Boltzmann equations needs to be solved, and the solution is numerically integrated to obtain the desired quantities. Emulators skip these steps, with a gain of orders of magnitude in speed. The gain is even more pronounced when emulators are applied directly to cosmological observables and tomographic analyses, which are our main interest here (Hu 1999). For example, in an analysis involving both GC, WL, and their cross-correlation, with 10 tomographic redshift bins the value of 210 different $C(\ell)$ needs to be computed, for each multipole ℓ. This is computationally very expensive if performed via standard approaches, especially within MCMC techniques for cosmological parameter inference. Thus, a direct emulation of the angular power spectra would provide a huge speed-up of the full analysis.

As discussed in the following, our implementation allows us to emulate the 3×2pt angular power spectrum coefficients $C(\ell)$ for a typical Stage-IV galaxy survey, with a speed-up $\mathcal{O}(10^3)$ with respect to the standard approach, while preserving an accuracy of about 0.175 per cent over the whole emulated range of angular scales. Besides their speed and accuracy, our emulators can be easily extended to larger parameter spaces. These figures compare very favourably to similar works in the literature.

Of course, the gain in terms of time needed for cosmological parameter inference comes at the price of data generation and training processes, which can be also computationally expensive. The overall balance, however, remains positive, given the massive computational costs needed to generate standard MC chains and the property of deep neural networks to be easily generalized to high-dimensional parameter spaces. In this work, emulators are treated as black-box functions and do not incorporate any explicit prior knowledge of the underlying physics. They learn the mapping between input parameters and output correlation coefficients in an entirely data-driven fashion.

This paper is structured as follows. In Section 2, we explain how we produced and pre-processed the training data set, the neural network architecture and the training procedure. In Section 3, we test the precision and speed of our emulators. In Section 4, we show that our emulator can recover unbiased posteriors when analysing a simulated Stage-IV survey. In Section 5, we make a comparison with the emulators present in literature. We then summarize our results and conclude in Section 6.

2 METHODOLOGY

2.1 Theoretical background

We give here a brief overview of the theory underlying two-point statistics used in the analyses of Large Scale Structure (LSS) surveys, using a notation similar to Blanchard et al. (2020). Throughout

[1] Test performed using a laptop with a i7-10510U CPU.

the paper, we consider as reference background model the simplest generalization of the standard ΛCDM model, in which dark energy is assumed to be a dynamical quantity with equation of state described by the Chevalier–Polarski–Linder parametrization: $w(a) = w_0 - w_a(1 - a)$ (Chevallier & Polarski 2001; Linder 2003). We refer to this background model as a flat 'w_0w_aCDM' model.

As mentioned above, our aim is to emulate three specific sets of two-point angular statistics involving GC and weak lensing, accounting also for the correlation between different tomographic redshift bins: (i) the autocorrelation of angular galaxy positions in a photometric sample; (*ii*) the cross-correlation of the lens galaxy positions with the estimated shear, and (iii) the autocorrelation of the shear field itself. We describe these quantities in terms of $C(\ell)$ angular power spectra, that is, the coefficients of the spherical harmonic decomposition of the above correlation functions, with the multipole order, ℓ, related to the inverse of the angular scale in configuration space.

In the standard approach, each $C(\ell)$ corresponds to a projection of the nonlinear matter power spectrum, $P_{\rm mm}(k, z)$, weighted with the window functions $W^A(z)$ and $W^B(z)$, where A and B correspond to different contributions to the measured power (as, specifically, $\gamma \equiv$ WL, $I \equiv$ intrinsic alignment, $\epsilon \equiv$ cosmic shear, and$g \equiv$ GC). Under the Limber approximation (Limber 1954; LoVerde & Afshordi 2008), these coefficients can be expressed as

$$C_{ij}^{AB}(\ell) = \frac{c}{H_0} \int_{z_{\rm min}}^{z_{\rm max}} {\rm d}z \frac{W_i^A(z) W_j^B(z)}{E(z) r^2(z)} P_{\rm mm}\left(k_\ell(z), z\right), \tag{1}$$

where $E(z) = H(z)/H_0$ is the dimensionless *Hubble* parameter, $z_{\rm min}$ and $z_{\rm max}$ are the minimum and maximum redshifts of the survey, $r(z)$ is the comoving distance, i and j label the tomographic redshift bins, and k_ℓ is the wavenumber, which is connected to the multipole order, ℓ, as this

$$k_\ell(z) = \frac{\ell + 1/2}{r(z)}. \tag{2}$$

In the adopted flat w_0w_aCDM cosmology, $E(z)$ reads

$$E(z) = \sqrt{\Omega_{\rm m,0}(1+z)^3 + \Omega_{\rm DE,0}(1+z)^{3(1+w_0+w_a)} e^{-3w_a \frac{z}{1+z}}}, \tag{3}$$

and the comoving distance is

$$r(z) = \frac{c}{H_0} \int_0^z \frac{{\rm d}z}{E(z)}. \tag{4}$$

Still following Blanchard et al. (2020), the WL window function is given by

$$W_i^\gamma(z) = \frac{3}{2} \left(\frac{H_0}{c}\right)^2 \Omega_M (1+z) r(z) \tilde{W}_i^\gamma(z), \tag{5}$$

where $\tilde{W}_i^\gamma(z)$ is the so-called lensing efficiency

$$\tilde{W}_i^\gamma(z) = \int_z^{z_{\rm max}} {\rm d}z' n_i^g\left(z'\right) \left(1 - \frac{r(z)}{r\left(z'\right)}\right), \tag{6}$$

with $n_i^g(z)$ being the normalized surface galaxy number density in the considered ith tomographic bin (see below for details).

Another important contribution that we include in our model is that of galaxy intrinsic alignments (IA): during galaxy formation, tidal interactions can affect galaxy shapes, by inducing intrinsically correlated orientations (Joachimi et al. 2015). This is an astrophysical contribution to the observed signal, and can be included via the window function

$$W_i^{\rm I}(\chi) = -\mathcal{A}_{\rm IA} C_{\rm IA} \Omega_{\rm m,0} \frac{(1+z)^{\eta_{\rm IA}}}{D(\chi)} n_i(\chi), \tag{7}$$

Table 1. Parameters used in the photometric redshift distribution $p(z|z_p)$ of equation (12) and Fig. 1.

c_b	z_b	σ_b	c_o	z_0	σ_o	$f_{\rm out}$
1.0	0.0	0.05	1.0	0.1	0.05	0.1

where $D(z)$ is the linear growth factor, $\rho_{\rm cr}$ is the critical density, $C_{\rm IA}$ is a constant (whose value is fixed to 0.0134), while $A_{\rm IA}$ and $\eta_{\rm IA}$ are two free parameters whose effect is to scale, respectively, in amplitude and redshift the IA contribution (Blanchard et al. 2020). The shear window function $W_i^\epsilon(z)$ corresponds to the sum of the lensing and IA window functions; as a consequence, the observed shear power spectrum is evaluated as

$$C_{ij}^{\epsilon\epsilon}(\ell) = C_{ij}^{\gamma\gamma}(\ell) + C_{ij}^{{\rm I}\gamma}(\ell) + C_{ij}^{\gamma{\rm I}}(\ell) + C_{ij}^{\rm II}(\ell). \tag{8}$$

Unlike WL, the GC selection function needs to account also for the galaxy bias $b_i^g(z)$ (see Desjacques, Jeong & Schmidt 2018, for a review),

$$W_i^g(z) = \frac{H(z)}{c} n_i^g(z) b_i^g(z). \tag{9}$$

We use the phenomenological galaxy bias model introduced in Tutusaus et al. (2020),

$$b(z) = A + \frac{B}{1 + \exp[-(z-D)C]}, \tag{10}$$

with $A = 1.0$, $B = 2.5$, $C = 2.8$, and $D = 1.6$, as obtained through a fit to the *Euclid* flagship simulation.[2]

Here, we model the true galaxy distribution as

$$n^g(z) \propto \left(\frac{z}{z_0}\right)^2 \exp\left[-\left(\frac{z}{z_0}\right)^{3/2}\right], \tag{11}$$

where $z_0 = z_m/\sqrt{2}$ is the median redshift of the distribution normalized as to obtain an integrated galaxy surface density $\bar{n}^g$. Here, we assume the same specifications as for the *Euclid* photometric survey (Laureijs et al. 2011), that is, $z_m = 0.9$ and $\bar{n}^g = 30\,{\rm arcmin}^{-2}$.

To account for photometric redshift errors, we convolve the true galaxy surface density, $n^g(z)$, with the following double Gaussian kernel describing the probability that a galaxy with redshift z has a measured redshift z_p (Kitching et al. 2009)

$$\begin{aligned} p(z|z_p) \;=\; & \frac{1-f_{\rm out}}{\sqrt{2\pi}\sigma_{\rm b}(1+z)} \exp\left\{-\frac{1}{2}\left[\frac{z - c_{\rm b}z_{\rm p} - z_{\rm b}}{\sigma_{\rm b}(1+z)}\right]^2\right\} \\ & + \frac{f_{\rm out}}{\sqrt{2\pi}\sigma_{\rm o}(1+z)} \exp\left\{-\frac{1}{2}\left[\frac{z - c_{\rm o}z_{\rm p} - z_{\rm o}}{\sigma_{\rm o}(1+z)}\right]^2\right\}. \end{aligned} \tag{12}$$

This is quite a flexible parametrization, which includes both multiplicative and additive bias in the redshift determination, accounting for a fraction $f_{\rm out}$ of galaxies with severely incorrect estimate of redshift. c_b and c_0 represent multiplicative bias, z_b and z_0 are an additive bias, and σ_b and σ_o are the width of the two Gaussians (to be scaled with the redshift considered); the values of the coefficients are given in Table 1.

Finally, the number density in the ith bin, entering equations (9)–(6), is

$$n_i^g(z) = \frac{\int_{z_i^-}^{z_i^+} {\rm d}z_{\rm p} n^g(z) p_{\rm ph}\left(z_{\rm p} \mid z\right)}{\int_{z_{\rm min}}^{z_{\rm max}} {\rm d}z \int_{z_i^-}^{z_i^+} {\rm d}z_{\rm p} n^g(z) p_{\rm ph}\left(z_{\rm p} \mid z\right)}, \tag{13}$$

[2]Euclid Collaboration (in preparation).

Table 2. Ranges of parameters used for the training data set generation. The cosmological parameters range is chosen as to be similar to that of EuclidEmulator2 (Knabenhans et al. 2021).

Parameter	Range	Centre
Ω_M	[0.29, 0.35]	0.32
Ω_B	[0.04, 0.06]	0.05
h	[0.61, 0.73]	0.67
n_s	[0.92, 1.0]	0.96
σ_8	[0.6, 1.0]	0.8
w_0	[−1.3, −0.7]	−1.0
w_a	[−0.7, 0.7]	0.0
η_{IA}	[−6, 6]	0
ℓ	[10, 3000]	–

where z_i^+ and z_i^- are the redshift values that define the ith tomographic bin. Here. we use 10 equipopulated redshift bins, so that the number density in each bin is simply $\bar{n}_i^g = \bar{n}^g/10$ and the boundaries of the bins are:

$$z_i = \{0.001, 0.42, 0.56, 0.68, 0.79, 0.9, 1.02, 1.15, 1.32, 1.58, 2.5\}.$$

The previous specifications lead to the galaxy densities shown in Fig.1. In a similar fashion to equation (8), we evaluate the cross-correlation power spectrum $C_{ij}^{\epsilon g}$ as the sum of two different terms

$$C_{ij}^{\epsilon g}(\ell) = C_{ij}^{\gamma g}(\ell) + C_{ij}^{Ig}(\ell). \tag{14}$$

2.2 Data set generation

In this subsection, we describe both the data set used in our analysis and the operations performed before the training phase. The $C(\ell)$ were evaluated using our code named `CosmoCentral.jl`,[3] a cosmological analysis package developed in `Julia`.[4] The linear matter power spectrum was obtained from CLASS and corrected for non-linear evolution using the revised `halofit` recipe from Takahashi et al. (2012). The angular spectra obtained for the central cosmology of Table 2 are shown in Fig. 2.

The parameter hyperspace described in Table 2 was then sampled using 200 logarithmic-spaced multipoles ℓ, while 10 000 combinations of cosmological parameters were generated according to the Latin Hypercube Sampling (LHS) algorithm (McKay, Beckman & Conover 1979). This prior was chosen to closely mimic that of `EuclidEmulator2` (Knabenhans et al. 2021).

2.3 Neural network emulator

Previous emulators (Mootoovaloo et al. 2020, 2022) were based on Gaussian processes, which have a neat probabilistic interpretation and allow an easy estimate of the uncertainty associated with their predictions. However, they are not ideal for handling big data sets. Although specialized techniques can be employed to scale Gaussian processes to millions of data (Hensman, Fusi & Lawrence 2013; Liu et al. 2018; Wang et al. 2019), they usually come at the cost of a lower precision or a higher computational cost. Neural networks, instead, can in a more straightforward way scale to regimes with huge data sets. This is a crucial point since, when feasible, by increasing the size of the training data set we can improve the precision and accuracy of the emulator.

[3]The code is publicly available on GitHub (Bonici & Carbone, in preparation).
[4]`Julia` is a just-in-time compiled language, developed for high-performance numerical computations. Here, `Julia` version 1.6.3 was used.

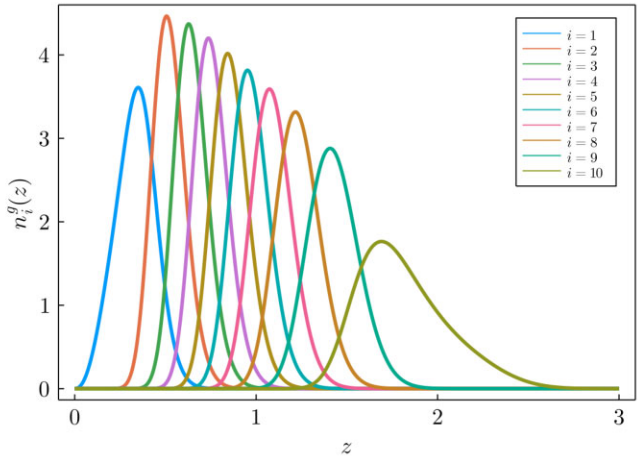

Figure 1. Normalized galaxy density distributions in tomographic redshift bins, accounting for the photometric errors given in Table 1, used in this work.

We address our problem as a classic regression task: given the value of the input parameters (the cosmological parameters and the chosen multipole ℓ), a neural network is trained to output the set of angular correlation coefficients, $C_{ij}^{AB}(\ell)$, for each of the three probes described in Section 2.1. In particular, we train a single emulator for GC autocorrelation, while for WL–GC cross-correlation and cosmic shear autocorrelation we train, respectively, two and three emulators[5] (equations 8 and 14). This approach presents several advantages. In fact, the trained emulators are going to be either independent, linear or quadratic in the IA parameter $\mathcal{A}_{\mathrm{IA}}$. This means that the latter can be treated analytically, with no need of emulation.

The neural networks are developed using `Flux.jl`, a `Julia` machine-learning library (Innes et al. 2018). The resulting `Julia` emulators can be easily wrapped and used in established PYTHON data-analysis frameworks (Brinckmann & Lesgourgues 2019; Torrado & Lewis 2021). The chosen architecture consists of four hidden layers with 100 neurons each, and a rectified linear unit activation function (Nwankpa et al. 2018), as presented in Fig. 3. The input layer takes eight features (corresponding to the cosmological parameters and the multipole ℓ). The output layer differs from that of similar emulators in the literature (Manrique-Yus & Sellentin 2020; Mootoovaloo et al. 2020), which typically compute the $C(\ell)$ for a specific tomographic combination, ij, and in a given multipole range (or even at a specific multipole value). In our case, instead we output 55 coefficients for the auto-power spectra, $C_{ij}^{AA}(\ell)$, and 100 coefficients for the cross-power spectra, $C_{ij}^{AB}(\ell)$, that is, we output at once all combinations, (i, j), of tomographic indices, with $i, j = 1, \ldots, 10$, for the required multipoles.[6] This approach has several advantages:

(i) Both the training and the predictions are fast, since we use only three emulators, rather than the several hundreds required, for example, in Manrique-Yus & Sellentin (2020) to emulate the photometric 3×2pt.

[5]Although there are four terms in equation (8), we trained only three emulators since we group together the $C_{ij}^{\gamma \mathrm{I}}$ and the $C_{ij}^{\mathrm{I}\gamma}$ contributions.
[6]Since our emulators can take a single multipole ℓ value as input, evaluating the C_ℓ's over a grid with several ℓ's requires running the emulators for each specific value of ℓ. More on this point can be found in Appendix A.

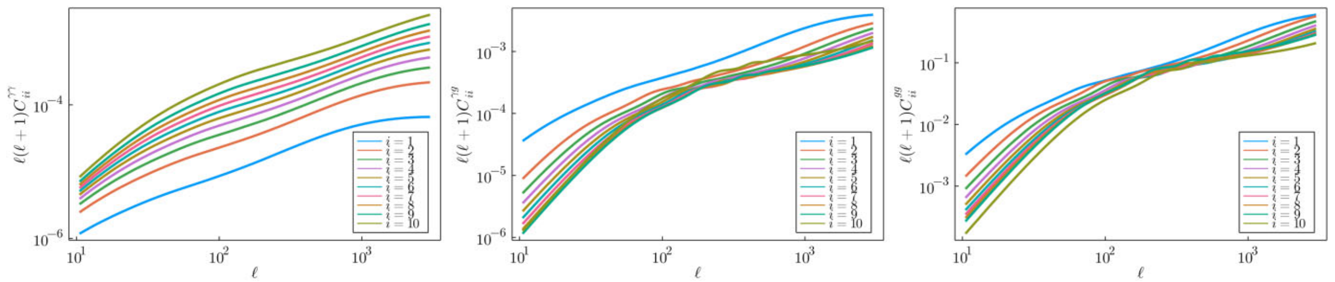

Figure 2. Theoretical predictions for, respectively, lensing autocorrelation, galaxy–galaxy lensing cross-correlation, and GC autocorrelation. The power spectrum was generated using CLASS in the central cosmology of Table 2, while the $C(\ell)$'s have been evaluated using `CosmoCentral.jl`.

Figure 3. Architecture definition of our neural network emulators. `n_output` changes accordingly to the emulator considered (55 and 100, respectively, for the autocorrelation and cross-correlation functions).

(ii) $C_{ij}^{AB}(\ell)$ are not uncorrelated between different tomographic bins. We take advantage of this correlation by emulating all tomographic combinations at once.

(iii) Our emulator suite is lightweight, occupying less than 1 MB of disc storage.

The architecture of the emulators is presented in Fig. 3.

A step of the utmost importance before training is the pre-processing of the training data set, which helps the neural network to better capture the features it has to learn, both reducing the training time and increasing the neural network performance (Ioffe & Szegedy 2015). Specifically, we first perform the following transformation

$$C_{ij}^{AB}(\ell) \longrightarrow \log(\ell(\ell+1))C_{ij}^{AB}(\ell), \tag{15}$$

and then normalize the resulting values within the [0,1] interval, performing a min–max normalization. We empirically verified the importance of this step: compared to a preliminary attempt, in which we tried to directly emulate the $C_{ij}^{AB}(\ell)$, this pre-processing yields a considerable improvement. Regarding the training phase, we chose the ADAptive Moment (ADAM) optimizer (Kingma & Ba 2014), with a learning rate of 3×10^{-4} and a batch size of 256; the training lasted for 1000 epochs. 80 per cent of the data set was used for training and 20 per cent for the validation. We used a mean-square-error loss function, that is, an L2 loss function, which was evaluated after each epoch both for the training and the validation data sets; when the test loss function decreases, the stored network coefficients are updated. The training of each emulator lasted for around one hour using an NVIDIA A100 GPU.

3 EMULATOR PERFORMANCE

We assess the speed of our emulators using `BenchmarkTool.jl`, which runs them several times to provide an accurate estimate of execution times.

We simultaneously emulated 30 multipoles ℓ, which realistically matches a real analysis; furthermore, if one were to be interested in a higher number of multipoles, we empirically found the scaling to be linear in the number of considered multipoles and we developed an efficient strategy to interpolate on the multipole grid, as described in Appendix A. We performed the test using the lensing autocorrelation emulator, obtaining a mean execution time of about 300 µs.

In Fig. 4, we show the relative emulation error for the specific set of cosmological parameters corresponding to the centre of the emulated space, as reported in Table 2. Remarkably, the accuracy is very good over all multipoles considered: the residuals for the lensing autocorrelation are almost always below 0.1 per cent, while for the other two angular statistics they almost never exceed the 0.2 per cent threshold. We also built a test data set of 4000 cosmologies distributed, according to the LHS, over the same range as the training data set reported in Table 2. To quantify the goodness of the emulators, we use the mean absolute relative difference (MARD) for each combination $\boldsymbol{\theta}$ of cosmological parameters present in the validation data set, defined as

$$\mathrm{MARD}^{\mathrm{AB}}(\boldsymbol{\theta}) = \frac{100}{N_{C(\ell)}} \sum_{\ell=\ell_{\min}}^{\ell_{\max}} \sum_{i,j} \left| 1 - \frac{C_{ij,\mathrm{NN}}^{\mathrm{AB}}(\boldsymbol{\theta})}{C_{ij,\mathrm{GT}}^{\mathrm{AB}}(\boldsymbol{\theta})} \right|, \tag{16}$$

where $C_{ij,\mathrm{NN}}^{\mathrm{AB}}$ is the neural network prediction, $C_{ij,\mathrm{GT}}^{\mathrm{AB}}$ is the ground truth, computed in the standard way (equation 1), and $N_{C(\ell)}$ represents the number of C_ℓ's computed for each combination of cosmological parameters. This metric quantifies the relative accuracy of the emulator and is the same employed in Manrique-Yus & Sellentin (2020). The resulting distribution is shown in Fig. 5. The important result is that the relative error of the predictions is well within 0.175 per cent, with the lensing emulator, in particular, yielding an accuracy of 0.1 per cent. This higher accuracy, compared to the other two cases, is due to the negligible effect of baryon acoustic oscillations (BAO) on $C_{ij}^{\gamma\gamma}$, which make the weak lensing angular power spectra smoother and hence easier to emulate than C_{ij}^{gg} and $C_{ij}^{\gamma g}$. In all cases, these errors are well below the typical uncertainties of non-linear corrections to the matter power spectrum: for example, `EuclidEmulator2` (Knabenhans et al. 2021) and `BaccoEmulator` (Angulo et al. 2021), have an accuracy of

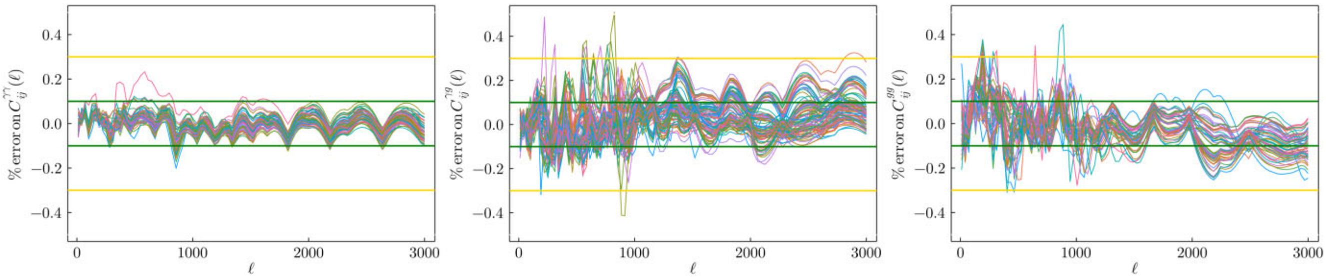

Figure 4. The relative error in the $C(\ell)$ evaluation for the central cosmology defined in Table 2. The outer and external couple of lines represent, respectively, a relative difference of 0.3 per cent and 0.1 per cent. The lensing emulator is the most precise, with an error smaller than 0.1 per cent for almost all evaluated coefficients. The superior performance of the WL emulator is due to the absence of the BAO wiggles, as can be seen in Fig. 2.

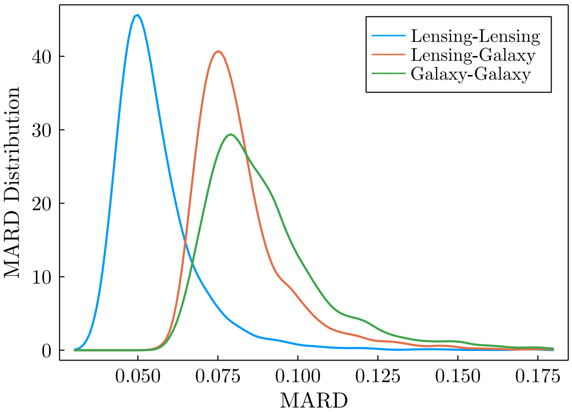

Figure 5. The distribution of the relative systematic errors of the three emulators, defined in equation (16), using a validation data set of 4000 cosmologies.

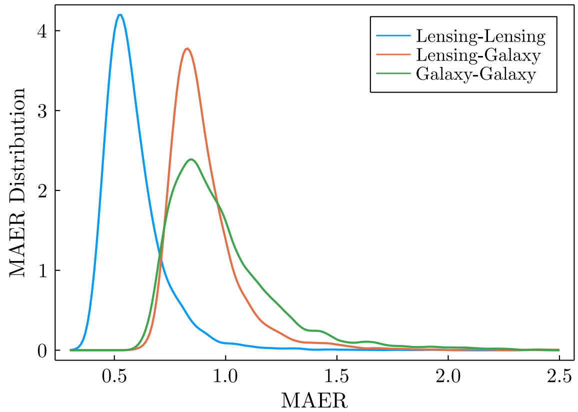

Figure 6. The distribution of the relative importance of systematic errors, with respect to the intrinsic cosmic variance, as defined in equation (18).

about 1 per cent – 2 per cent up to $k = 5\,h\mathrm{Mpc}^{-1}$. To estimate the significance of this level of systematic error with respect to the typical measurement uncertainty, we compare it to the expected cosmic variance $\sigma(\ell)$

$$\sigma_{ij}^{\mathrm{AB}}(\ell) = \sqrt{\frac{C_{ii}^{\mathrm{AA}}(\ell)C_{jj}^{\mathrm{BB}}(\ell) + \left(C_{ij}^{\mathrm{AB}}(\ell)\right)^2}{f_{\mathrm{sky}}(2\ell+1)}}\,, \tag{17}$$

where f_{sky} is the fraction of surveyed sky. We define the mean absolute error ratio (MAER) as

$$\mathrm{MAER}^{\mathrm{AB}}(\boldsymbol{\theta}) = \frac{100}{N_{C(\ell)}} \sum_{\ell=\ell_{\min}}^{\ell_{\max}} \sum_{i,j} \frac{\left|C_{ij,\mathrm{NN}}^{\mathrm{AB}} - C_{ij,\mathrm{GT}}^{\mathrm{AB}}\right|}{\sigma_{ij}^{\mathrm{AB}}(\ell)}, \tag{18}$$

where for clarity, we omitted to write explicitly the dependence on the cosmological parameters $\boldsymbol{\theta}$.

To estimate $\sigma(\ell)$, f_{sky} was set to 0.4 (a value compatible with the ∼15 000 deg^2 expected from the *Euclid* and DESI surveys (Laureijs et al. 2011). Equation (18) does not include the shot-noise contribution, which is a survey-dependent quantity and cannot be rescaled as easily as the (multiplicative) f_{sky} term. Neglecting shot noise, however, is a conservative approach: any such term would yield a higher relative performance for our emulators. As can be seen from Fig. 6, the MAER is always smaller than 2 per cent. For comparison, the cosmological library developed by the LSST collaboration, `CCL` (Chisari et al. 2019), set a threshold of 10 per cent for this metric.

4 APPLICATION TO SYNTHETIC DATA

As a final test, we run a bayesian analysis on a synthetic 3 × 2 pt data set, comparing the results obtained with the CLASS Boltzmann solver and our emulators. The synthetic data vector has been produced using the central values of the parameter ranges in the third column of Table 2, with 30 logarithmically spaced multipole bins for $\ell \in [10, 3000]$. Regarding the likelihood, we considered a multivariate normal distribution, accounting only for the Gaussian contribution to the covariance matrix (Barreira, Krause & Schmidt 2018)

$$\begin{aligned} &\mathrm{Cov}\left[C_{ij}^{AB}(\ell), C_{kl}^{A'B'}(\ell')\right] = \frac{1}{(2\ell+1) f_{\mathrm{sky}}\,\Delta\ell}\cdot \\ &\left[C_{ik}^{AA'}(\ell) + N_{ik}^{AA'}(\ell)\right]\left[C_{jl}^{BB'}(\ell') + N_{jl}^{BB'}(\ell')\right] + \\ &+\left[C_{il}^{AB'}(\ell) + N_{il}^{AB'}(\ell)\right]\left[C_{jk}^{BA'}(\ell') + N_{jk}^{BA'}(\ell')\right]\delta_{\ell\ell'}^{K}. \end{aligned} \tag{19}$$

where A, B, A', B' are the probe indices, i, j, j, k are the tomographic indices, f_{sky} the sky fraction (which we set to 0.4), ℓ and ℓ' the multipoles, $\Delta\ell$ the multipole bin width, and N_{ij}^{AB} the noise, given by

$$N_{ij}^{\mathrm{GG}} = \frac{1}{\bar{n}_i^g}\delta_{ij}, \quad N_{ij}^{LL} = \frac{\sigma_\epsilon^2}{\bar{n}_i^g}\delta_{ij}, \quad N_{ij}^{LG} = N_{ij}^{\mathrm{GL}} = 0. \tag{20}$$

Regarding the galaxy shape noise and surface density, we respectively consider $\sigma_\epsilon^2 = 0.3$ and $\bar{n}_i^g = 30\,\mathrm{arcmin}^{-2}$ in each redshift bin; the latter number represents a stringent test on our emulator precision, as we expect Stage-IV surveys, such as *Euclid*, to have a surface galaxy density smaller by an order of magnitude and hence a larger measurement error.

The reference chains have been produced using CLASS and `CosmoCentral.jl` for the computation of the theory vector, using the ensamble sampler EMCEE (Foreman-Mackey et al. 2013) to explore the likelihood. Regarding the chains produced using our emulators, in order to fully take advantage of their differentiability, we also employed the No-U-Turn-Sampler (NUTS) version of the Hamiltonian Monte Carlo (HMC) sampler as coded in the `Turing.jl` library (Ge, Xu & Ghahramani 2018, here we simply describe the most basic features of this algorithm, more details can be found in Hoffman & Gelman 2014; Betancourt 2018). HMC is a gradient-based sampler that can be used to efficiently explore high-dimensional distributions with complex geometries. HMC draws samples from the target space by solving Hamiltonian trajectories, where the potential energy of a particle $V(q)$ is given by the negative log-likelihood, $-\log P$. Furthermore, conjugate momenta p to the position variables are introduced such that

$$-\log P(q) = V(q)\,, \quad H(q, p) = V(q) + K(p)\,, \tag{21}$$

where $K(p)$ is the kinetic energy term. At each step of the MCMC, a random initial momentum p is drawn and the Hamilton equations are numerically solved by a leapfrog integrator. After some steps a Metropolis–Hastings acceptance criterion is applied: the better the energy conservation along the numerically simulated trajectory, the higher the acceptance rate. Employing such a sampling scheme can deliver efficient exploration even of challenging posteriors.

While traditionally this sampling algorithm has been devoted to field-level reconstructions (Bayer, Seljak & Modi 2023; Sellentin et al. 2023), only recently the cosmological community started employing it in analysis involving summary statistics (Bonici, Bianchini & Ruiz-Zapatero 2023; Campagne et al. 2023; Piras & Spurio Mancini 2023; Ruiz-Zapatero et al. 2024). There are mainly two reasons for the late adoption of this algorithm. First and foremost, several gradient-free method have been developed and refined by the community and these methods proved to be powerful enough to analyse current and past surveys; the situation is likely to change with next generation LSS surveys, when we might expect likelihood analysis with $\mathcal{O}(100)$ parameters (see Piras & Spurio Mancini 2023, for a concrete example). The second reason is technical, as the leapfrog integrator requires the gradient of the log-likelihood as an input. With standard approaches based on Boltzmann solvers such as CAMB or CLASS (Lewis et al. 2000; Lesgourgues 2011), evaluating the likelihood gradient requires computing many finite-difference derivatives, which are both expensive and may suffer poor accuracy and precision. With an NN, the situation is completely different, since modern computer science techniques such as automatic differentiation enable for a precise and efficient evaluation of derivatives.

For the reference CLASS case, the EMCEE chains have been computed using 20 walkers, with 2000 burn-in and 20 000 accepted steps, and required around 1500 CPU hours. Regarding the emulators chains, in the EMCEE case, the chains have been computed using the same settings employed for the CLASS chains, and required around 20 CPU hours. Instead, when using the NUTS HMC sampler, we run 6 chains in parallel, with 1000 burn-in and 4000 accepted steps, using around 16 CPU hours; the speed-up provided by the HMC sampler is only of 20 per cent, since the posterior we are studying is extremely simple (see also Ruiz-Zapatero et al. 2024) for a similar result). In both cases, the agreement between the two analyses is excellent: the difference on the marginalized cosmological parameter means are below $0.1\,\sigma$, as can be seen in Table 3, and the 2D marginalized posteriors are almost perfectly overlapping as can be seen in Fig. 7 where we show the comparison between the CLASS--EMCEE results and our emulator-HMC ones. Thus, when exploiting their differentiability, our emulators deliver a speed-up of almost two orders of magnitude, while keeping the posterior almost unchanged.

Table 3. Means and 95 per cent limits of the 1D marginal posteriors of the cosmological parameters obtained from our simulated data set. The second and third columns show the results obtained, respectively, using CLASS and our emulators.

	CLASS + EMCEE	`CosmoCentral.jl`+NUTS
	Parameter means and 95 per cent limits	
Ω_M	0.312001 ± 0.00184	0.31998 ± 0.00178
Ω_B	0.0499 ± 0.0020	0.0501 ± 0.0021
h	0.669 ± 0.01156	0.670 ± 0.01159
n_s	0.960 ± 0.0031	0.959 ± 0.0031
σ_8	0.8000 ± 0.0017	0.8001 ± 0.0016
w_0	-0.998 ± 0.023	-1.001 ± 0.022
w_a	-0.001 ± 0.079	0.002 ± 0.076

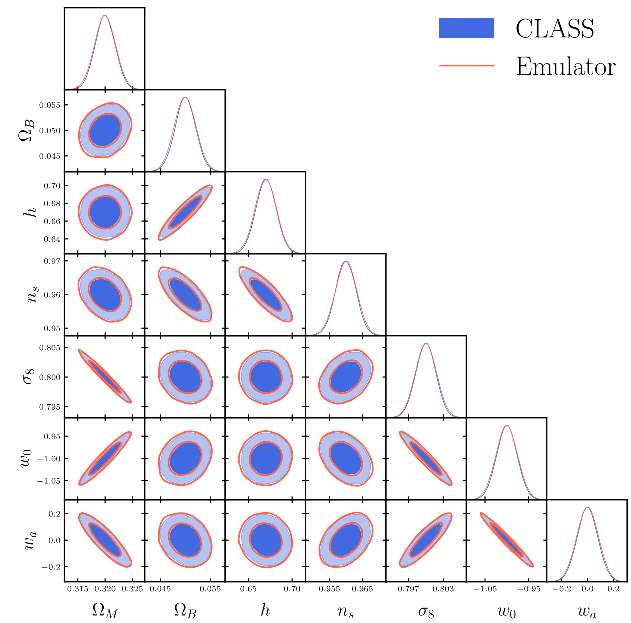

Figure 7. Triangle plot, showing the standard CLASS chains and the ones obtained using our emulators and the NUTS–HMC sampler, for a 3×2 pt data set containing weak lensing, GC and tangential shear synthetic measurements over an ℓ range from 10 to 3000. The differences on the marginalized cosmological parameters is lower than 0.1σ.

5 COMPARISON WITH EXISTING EMULATORS

As we briefly described in the Introduction, there are two categories of LSS emulators in the literature. The first category emulates the matter power spectrum in real space, $P_{\mathrm{mm}}(k)$, which is then fed to a code that computes the theoretical $C_{ij}^{AB}(\ell)$. The second category (Manrique-Yus & Sellentin 2020; Mootoovaloo et al. 2020), instead, emulates directly the latter. Our code belongs to this second class, with our implementation introducing considerable improvement over previous ones.

A direct comparison can be performed with the emulators presented in Manrique-Yus & Sellentin (2020), where the authors trained a neural network for each tomographic combination and multipole bin, as they divided the multipole range in six equispaced bins. Had we followed a similar approach, we would have had to train a total of 1260 emulators, largely increasing the training cost. In terms of accuracy, Fig. 5 shows how our emulation errors are always below 0.2 per cent, compared to values between 0.4 per cent and 1 per cent in fig. A1 of Manrique-Yus & Sellentin (2020).

On the other hand, comparison with the first category is more delicate. For instance, one advantage of $P_{\rm mm}(k)$ emulators is that they are more flexible, since they are agnostic about the galaxy redshift distribution (Spurio Mancini et al. 2022). $C_{ij}^{AB}(\ell)$ emulators, instead, are inevitably tailored for a specific survey: if the survey changes, the training process needs to be repeated. However, the most resource-demanding step of our approach is the estimation of the power spectra from Boltzmann codes. Once computed for a large range of redshifts, as done for this work, these could be stored and re-used to evaluate the $C_{ij}^{AB}(\ell)$ with faster codes than current Boltzmann solvers.[7]

The training procedure can also be accelerated, through transfer learning (Zhuang et al. 2019). For instance, we expect that the final *Euclid* galaxy redshift distribution will be different from the one employed in this implementation. However, it will still probe a similar redshift range, resulting in a small effect on the $C_{ij}^{AB}(\ell)$. We can thus train a new, emulator customized for the specific case, but using as initial weights those from the current one. This will make the training procedure start from an optimal situation, close to a minimum of the loss function, resulting in a faster training.

Another advantage of $P_{\rm mm}$ emulators is their insensitivity to probe-specific nuisance parameters, which enter the $C_{ij}^{AB}(\ell)$ evaluation through the weight functions $W_i^{\rm A}(z)$. Therefore, the training set in the case of $C_{ij}^{AB}(\ell)$ emulators needs to cover a larger parameter hyperspace. However, considering that for this work we needed 10 000 cosmologies, the resulting cost is still acceptable and the gain is still positive. Furthermore, some parameters, as the magnification bias, can be added analytically as multiplicative factors, thus at no cost for the $C_{ij}^{AB}(\ell)$ emulators (Abbott et al. 2018a).

For their part, the $C_{ij}^{AB}(\ell)$ emulators present several advantages. First, they are blazingly fast, since they do not need to compute intermediate quantities like weight functions, background variables and numerical integrals. The net result is a gain of about a factor of 1000 in speed, compared to standard Boltzmann integrators.

Emulating directly the $C_{ij}^{AB}(\ell)$, rather than $P_{\rm mm}(k)$, becomes even more advantageous when one needs to drop the Limber approximation, when projecting from 3D to 2D statistics. This assumption can in fact become a source of systematic error in high-precision measurements, as expected from next-generation surveys (Fang et al. 2020; Martinelli et al. 2022). Although efficient algorithms to estimate the $C_{ij}^{AB}(\ell)$ from $P_{\rm mm}(k)$ without the Limber approximation have been recently developed (Schöneberg et al. 2018; Fang et al. 2020), the advantage of bypassing completely this computational phase through a direct $C_{ij}^{AB}(\ell)$ emulation, is evident: once the training set of $C_{ij}^{AB}(\ell)$ is built properly, using or not the Limber approximation, the emulator will learn accordingly without any change in the architecture or procedure.

6 CONCLUSIONS

In this work, we have extended the `CosmoCentral]` package with a set of fast cosmological emulators, which are capable to predict a specific set of typical 2D statistics from galaxy surveys, the so-called 3×2pt statistics from WL and photometric GC, in less than a millisecond. We have shown that, compared to previous works, they are more precise and flexible, with significant saving of computational resources. Precision is also improved, using an appropriate pre-processing step and possibly benefiting of a different architecture: since we are training a single neural network to learn all tomographic combinations, (i, j), at once, the emulator can take advantage of the existing physical correlations between different combinations to deliver the correct result. In order to assess the accuracy of our emulators, we performed several tests. First, we used a validation data set to assess the emulator precision finding that the emulation error is 50 times smaller than the expected measurement error for a Stage-IV galaxy survey. Second, we also performed a full bayesian analysis on a mock dataset, using both CLASS and our emulators. Although, thanks to the exploitation of their differentiability, our emulators chains required around 100 times less computational resources than the ones obtained with CLASS, the agreement on the marginalized errors is better than $0.1\,\sigma$, showing that improved computational performance does not come at the cost of a reduced precision.

Our current results could be further improved by including more astrophysical effects and nuisance parameters as the galaxy bias. A full beyond-Limber version of the $C(\ell)$ emulators is also in our plans. As discussed in the previous section, with our approach this is particularly advantageous in terms of computational cost. We could also further improve in terms of hyperparameters of the neural network: in the current implementation we have used general-purpose dense neural networks, all with the same number of layers and neurons, without further specialization. However, using different architectures for each emulator could increase the precision further.

[7]For instance, once the power spectrum $P_{\rm mm}$ for a given cosmology is provided, `CosmoCentral.jl` can evaluate the $C_{ij}^{AB}(\ell)$ in about 0.05 s, requiring less than 10 min for 10 000 cosmologies.

ACKNOWLEDGEMENTS

MB warmly thanks Luca Paganin, Alessio Spurio Mancini, and Silvio Traversaro for useful discussions. MB acknowledges financial support from the ASI agreement no. I/023/12/0 'Euclid attivitá relativa alla fase B2/C'.

DATA AVAILABILITY

The data underlying this article will be shared on reasonable request to the corresponding author.

REFERENCES

Abbott T. M. C. et al., 2018a, Phys. Rev. D, 98, 043526
Abbott T. M. C. et al., 2018b, ApJS, 239, 18
Aghanim N. et al., 2020, A&A, 641, A6
Albers J., Fidler C., Lesgourgues J., Schöneberg N., Torrado J., 2019, J. Cosmol. Astropart. Phys., 2019, 028
Angulo R. E., Zennaro M., Contreras S., Aricò G., Pellejero-Ibañez M., Stücker J., 2021, MNRAS, 507, 5869
Aricò G., Angulo R. E., Zennaro M., 2021, Open Res Europe, 1, 152
Barreira A., Krause E., Schmidt F., 2018, J. Cosmol. Astropart. Phys., 2018, 053
Bayer A. E., Seljak U., Modi C., 2023, in 40th International Conference on Machine Learning, preprint (arXiv:2307.09504) https://icml.cc/virtual/2023/28200
Bernal J. L., Verde L., Riess A. G., 2016, J. Cosmol. Astropart. Phys., 2016, 019
Betancourt M., 2018, A Conceptual Introduction to Hamiltonian Monte Carlo, preprint (arXiv:1701.02434)
Blanchard A. et al., 2020, A&A, 642, A191

Bonici M., Bianchini F., Ruiz-Zapatero J., 2024, Open J. Astrophys., 7, 10
Brinckmann T., Lesgourgues J., 2019, Phys. Dark Univ., 24, 100260
Campagne J.-E. et al., 2023, Open J. Astrophys., 6, 1
Carrasco J. J. M., Hertzberg M. P., Senatore L., 2012, J. High Energy Phys., 2012, 082
Cataneo M., Lombriser L., Heymans C., Mead A., Barreira A., Bose S., Li B., 2019, MNRAS, 488, 2121
Chevallier M., Polarski D., 2001, Int. J. Mod. Phys. D, 10, 213
Chisari N. E. et al., 2019, ApJS, 242, 2
Cybenko G., 1989, Math. Control Signal Syst., 2, 303
Desjacques V., Jeong D., Schmidt F., 2018, Phys. Rep., 733, 1
Dore O. et al., 2019, BAAS, 51, 341
Fang X., Krause E., Eifler T., MacCrann N., 2020, J. Cosmol. Astropart. Phys., 2020, 010
Feldman H. A., Kaiser N., Peacock J. A., 1994, ApJ, 426, 23
Fendt W. A., Wandelt B. D., 2006, ApJ, 654, 2
Foreman-Mackey D., Hogg D. W., Lang D., Goodman J., 2013, PASP, 125, 306
Ge H., Xu K., Ghahramani Z., 2018, in Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics,1682
Hensman J., Fusi N., Lawrence N. D., 2013, Proc. Int. Conf. Uncertainty Artif. Intell., 272
Hoffman M. D., Gelman A., 2014, J. Mach. Learn. Res., 15, 1593
Hornik K., Stinchcombe M., White H., 1989, Neural Netw., 2, 359
Hu W., 1999, ApJ, 522, L21
Innes M. et al., 2018, CoRR
Ioffe S., Szegedy C., 2015, preprint (arXiv:1502.03167) https://www.nature.com/articles/s41597-023-02096-0#Bib1
Jimenez R., Verde L., Peiris H., Kosowsky A., 2004, Phys. Rev. D, 70, 023005
Joachimi B. et al., 2015, Space Sci. Rev., 193, 1
Kingma D. P., Ba J., 2015, In ICLR
Kitching T. D., Amara A., Abdalla F. B., Joachimi B., Refregier A., 2009, MNRAS, 399, 2107
Knabenhans M. et al., 2021, MNRAS, 505, 2840
Laureijs R. et al., 2011a, preprint (arXiv:1110.3193) https://academic.oup.com/mnras/article/491/2/2655/5612207
Lesgourgues J., 2011, https://arxiv.org/abs/1104.2932
Lewis A., Challinor A., Lasenby A., 2000, ApJ, 538, 473
Limber D. N., 1954, ApJ, 119, 655
Linder E. V., 2003, Phys. Rev. Lett., 90, 091301
Liu H., Cai J., Wang Y., Ong Y., 2018, in Proc. ICML
LoVerde M., Afshordi N., 2008, Phys. Rev. D, 78, 123506
Manrique-Yus A., Sellentin E., 2020, MNRAS, 491, 2655
Martinelli M., Dalal R., Majidi F., Akrami Y., Camera S., Sellentin E., 2022, MNRAS, 510, 1964
McKay M. D., Beckman R. J., Conover W. J., 1979, Technometrics, 21, 239
Mootoovaloo A., Heavens A. F., Jaffe A. H., Leclercq F., 2020, MNRAS, 497, 2213
Mootoovaloo A., Jaffe A. H., Heavens A. F., Leclercq F., 2022, Astron. Comput., 38, 100508
Nwankpa C., Ijomah W., Gachagan A., Marshall S., 2018, preprint (arXiv:1811.03378)
Perlmutter S. et al., 1999, ApJ, 517, 565
Piras D., Spurio Mancini A., 2023, Open J. Astrophys., 6, 20
Riess A. G. et al., 1998, AJ, 116, 1009
Ruiz-Zapatero J., Alonso D., Garcia Garcia C., Nicola A., Mootoovaloo A., Sullivan J. M., Bonici M., Ferreira P. G., 2024, Open J. Astrophys., 7, 11
Schöneberg N., Simonović M., Lesgourgues J., Zaldarriaga M., 2018, J. Cosmol. Astropart. Phys., 2018, 047
Sellentin E., Loureiro A., Whiteway L., Lafaurie J. S., Balan S. T., Olamaie M., Jaffe A. H., Heavens A. F., 2023, Open J. Astrophys., 6, 31
Spurio Mancini A., Piras D., Alsing J., Joachimi B., Hobson M. P., 2022, MNRAS, 511, 1771
Takahashi R., Sato M., Nishimichi T., Taruya A., Oguri M., 2012, ApJ, 761, 152
Torrado J., Lewis A., 2021, J. Cosmol. Astropart. Phys., 2021, 057
Tutusaus I. et al., 2020, A&A, 643, A70
Upham R. E. et al., 2022, A&A, 660, A114
Wang K. A., Pleiss G., Gardner J. R., Tyree S., Weinberger K. Q., Wilson A. G., 2019, in NeurIPS
Weltman A. et al., 2020, Publ. Astron. Soc. Aust., 37, e002
Yamamoto K., Nakamichi M., Kamino A., Bassett B. A., Nishioka H., 2006, PASJ, 58, 93
Zhuang F., Qi Z., Duan K., Xi D., Zhu Y., Zhu H., Xiong H., He Q., 2019, preprint (arXiv:1911.02685)

APPENDIX A: USING A FINER GRID

In this appendix, we introduce an approach which allows us to evaluate C_ℓ's on a grid much finer than the one presented in the main text, containing all integer multipoles or band powers. We show that this allows us to increase by two orders of magnitudes the number of evaluated multipoles, without affecting the precision or speed of the method.

In the main text of the paper, we have discussed the performances of our emulators for a grid of 30 multipoles. The analysis of forthcoming surveys, however, will require evaluating the C_ℓ's over all integer values of the multipoles ℓ, as to compute some kind of binned power spectra (Upham et al. 2022). The most naive approach would require calling the emulator repeatedly over all integer multipoles: if, for example, $\ell \in (10, 3000)$, the impact on the overall performance would be severe. We show here, however, that it is still possible to adopt a strategy to apply the emulators on a subset of multipoles without degrading the computational performance. It requires the following steps:

(i) evaluate the C_ℓ's using our emulators on a log-spaced grid with 59 multipoles, between the minimum and maximum multipoles employed in the analysis;

(ii) use the emulated C_ℓ's to perform a quadratic interpolation over 30 multipole bins;

(iii) use the polynomial interpolants to evaluate C_ℓ's over a grid with all integers multipoles.

With this strategy, the computation time to evaluate all multipoles increases to about 1.7 ms, to be compared to the original estimate of ~ 1 ms. We stress that the interpolations performed in this way add a negligible error to the C_ℓ's evaluation (which is found to remain always below 0.01 per cent).

We also find that if we are working with interpolants and we want to compute band powers, it is possible to make a further step. The expression of the band powers of a quadratic interpolant is

$$P_b = \sum_\ell \frac{\ell(\ell+1)}{2\pi}\left[\ell_{\rm min}^{b+1} - \ell_{\rm min}^{b}\right]^{-1} C_\ell = \\ = \sum_\ell \frac{\ell(\ell+1)}{2\pi}\left[\ell_{\rm min}^{b+1} - \ell_{\rm min}^{b}\right]^{-1} (a\ell^2 + b\ell + c) \qquad \text{(A1)}$$

where a, b, and c are the three coefficients of the quadratic interpolant. Equation (A1) shows that the band powers can be written as a sum of three (cosmology independent) sums, that are multiplied by the interpolant coefficients. If we store in memory the (cosmology independent) sums, this enables to evaluate them just once in a single MCMC analysis, with a mean execution time of 1.5 ms in this scenario.

This paper has been typeset from a TEX/LATEX file prepared by the author.