CosmoPower: emulating cosmological power spectra for accelerated Bayesian inference from next-generation surveys

Alessio Spurio Mancini[1,2]★, Davide Piras[2], Justin Alsing[3], Benjamin Joachimi[2], Michael P. Hobson[4]
[1]*Mullard Space Science Laboratory, University College London, Holmbury St. Mary, Dorking, Surrey, RH5 6NT, UK*
[2]*Department of Physics and Astronomy, University College London, Gower Street, London, WC1E 6BT, UK*
[3]*Oskar Klein Centre for Cosmoparticle Physics, Department of Physics, Stockholm University, Stockholm SE-106 91, Sweden*
[4]*Astrophysics Group, Cavendish Laboratory, J. J. Thomson Avenue, Cambridge, CB3 0HE, UK*

Accepted XXX. Received YYY; in original form ZZZ

ABSTRACT

We present CosmoPower, a suite of neural cosmological power spectrum emulators providing orders-of-magnitude acceleration for parameter estimation from two-point statistics analyses of Large-Scale Structure (LSS) and Cosmic Microwave Background (CMB) surveys. The emulators replace the computation of matter and CMB power spectra from Boltzmann codes; thus, they do not need to be re-trained for different choices of astrophysical nuisance parameters or redshift distributions. The matter power spectrum emulation error is less than 0.4% in the wavenumber range $k \in [10^{-5}, 10]\,\mathrm{Mpc}^{-1}$, for redshift $z \in [0, 5]$. CosmoPower emulates CMB temperature, polarisation and lensing potential power spectra in the 5σ region of parameter space around the *Planck* best fit values with an error $\lesssim 10\%$ of the expected shot noise for the forthcoming Simons Observatory. CosmoPower is showcased on a joint cosmic shear and galaxy clustering analysis from the Kilo-Degree Survey, as well as on a Stage IV *Euclid*-like simulated cosmic shear analysis. For the CMB case, CosmoPower is tested on a *Planck* 2018 CMB temperature and polarisation analysis. The emulators always recover the fiducial cosmological constraints with differences in the posteriors smaller than sampling noise, while providing a speed-up factor up to $O(10^4)$ to the complete inference pipeline. This acceleration allows posterior distributions to be recovered in just a few seconds, as we demonstrate in the *Planck* likelihood case. CosmoPower is written entirely in Python, can be interfaced with all commonly used cosmological samplers and is publicly available ⌂.

Key words: large-scale structure of Universe – cosmic background radiation – methods: statistical – methods: data analysis

1 INTRODUCTION

Analysis of the two-point statistics of cosmological fields is one of the cornerstones of modern observational cosmology. For parameter inference pipelines involving two-point statistics (i.e. power spectra, or their derived real-space counterparts, correlation functions), the computational bottleneck is running Boltzmann solvers like Camb (Lewis et al. 2000) or Class (Lesgourgues 2011; Blas et al. 2011) to compute theoretical power spectra for a given cosmology. However, cosmological power spectra are generally smooth functions of their input cosmological parameters, and hence lend themselves well to *emulation*: finding compact, accurate, and fast-to-evaluate surrogate functions that map cosmological parameters to the corresponding predicted power spectra.

Emulation offers the promise of reducing the computational overhead of evaluating cosmological power spectra by many orders of magnitude, with negligible loss of accuracy in the final parameter inference. This surrogate modelling approach has recently seen numerous applications to the Bayesian solution of the *inverse problem* in different scientific fields, ranging from geophysical seismic waves (Das et al. 2018; Spurio Mancini et al. 2021; Piras et al. 2021) to stellar and galaxy spectra (Czekala et al. 2015; Alsing et al. 2020), from chemical mechanisms (de Mijolla et al. 2019; Kasim et al. 2021) to applied engineering (Thiagarajan et al. 2020; Buffington et al. 2020).

Emulation of cosmological power spectra is not a new idea either. Early examples of emulators include CMBwarp (Jimenez et al. 2004) and PICO (Fendt & Wandelt 2007), both performing polynomial regression of power spectra (represented in some compact basis). The matter power spectrum emulators built from the Coyote Universe simulations (Heitmann et al. 2009, 2010, 2013; Lawrence et al. 2010, 2017) are based on Gaussian Process regression (Rasmussen & Williams 2005), and were extended by Ramachandra et al. (2021) to $f(R)$ cosmologies. Recently, the Euclid Emulator was proposed as a surrogate model for the nonlinear matter power spectrum (Knabenhans et al. 2019; Euclid Collaboration et al. 2021), Mootoovaloo et al. (2020) developed a Gaussian Process emulator of cosmic shear band powers, while Mootoovaloo et al. (2022) and Ho et al. (2021) used Gaussian Processes to emulate the matter power spectrum. Bird et al.

★ a.spuriomancini@ucl.ac.uk

© The Author(s) 2022. Published by Oxford University Press on behalf of Royal Astronomical Society. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

(2019) and Rogers et al. (2019) developed Gaussian Process emulators for the Lyman-α forest flux power spectrum.

CosmoNet (Auld et al. 2007; Auld et al. 2008) is the first application of neural networks to cosmological power spectra emulation, followed by Agarwal et al. (2012, 2014). More recently, Manrique-Yus & Sellentin (2019) developed neural network interpolators for angular power spectra of LSS observables, while Albers et al. (2019) used neural networks to accelerate parts of power spectra computations within the Boltzmann code Class; moreover, the Bacco project (Angulo et al. 2021) recently included a neural network interpolator for the linear matter power spectrum (Aricò et al. 2021). Kern et al. (2017), Schmit & Pritchard (2017) and Bevins et al. (2021) developed neural network emulators for the 21-cm global signal or power spectrum.

In this paper, we introduce a suite of neural cosmological power spectrum emulators covering both CMB (temperature, polarisation and lensing) and (linear and nonlinear) matter power spectra. These emulators provide orders-of-magnitude speed-up over direct Boltzmann solvers, whilst being comfortably accurate for upcoming surveys such as the Simons Observatory (Ade et al. 2019)[1], *Euclid* (Laureijs et al. 2011)[2], the Vera Rubin Observatory (Ivezić et al. 2019)[3] and the Nancy Grace Roman Space Telescope (Spergel et al. 2015)[4]. For LSS observables, we demonstrate the accuracy and acceleration provided by our emulators on data from the Kilo-Degree Survey (KiDS) as well as on a simulated cosmic shear analysis of a *Euclid*-like survey. For the CMB, we validate CosmoPower on a *Planck* 2018 CMB temperature and polarisation analysis.

Our emulators are trained to provide accurate emulation of cosmological power spectra on a very wide range of cosmological parameters, and they easily allow for different configurations of input and derived cosmological parameters. In addition, they are fully differentiable, which makes them amenable to gradient-based inference, and can be run on Graphics Processing Units to gain further acceleration.

The structure of this paper is as follows. In Sect. 2 we introduce the neural network emulation framework tailored to power spectrum emulation. Application to the matter power spectrum emulation is presented in Sect. 3, including validation on the KiDS and *Euclid*-like analyses. Application to the CMB case is presented in Sect. 4, including validation on the *Planck* analysis. We summarise the properties of CosmoPower and conclude in Sect. 5.

2 EMULATING COSMOLOGICAL POWER SPECTRA

We consider two methods for power spectra emulation.

- The first one is a direct mapping between cosmological parameters and power spectra by means of a neural network (NN), schematically represented in the left-hand panel of Fig. 2 (see e.g. Bishop 2006 for an introduction to NNs; here, we follow a similar notation to Alsing et al. 2020). A NN is a set of stacked layers, each composed of multiple nodes; we label each of the n nodes with index i. To each of them a weight W_i and bias parameter b_i are associated, whose linear combination is passed through a nonlinear activation function (see Appendix A1 for details on the activation function employed in this work); we denote the ensemble of weights and biases as $\boldsymbol{w}$. The nonlinear function parameterised by $\boldsymbol{w}$ maps the input cosmological parameters $\boldsymbol{\theta}$ to the (log-)power spectra $\boldsymbol{P}_\lambda = \boldsymbol{P}_\lambda(\boldsymbol{\theta}; \boldsymbol{w})$, where λ is either the wavenumber k for the matter power spectrum, or the multipole ℓ in the CMB case. The spectra used for training the deep learning emulators are preprocessed by calculating their logarithm, followed by standardisation of the logarithmic features, i.e. dividing each logarithmic feature by its standard deviation after subtracting its mean. Taking the logarithm of the spectra reduces the dynamic range in the training data, and ensures that minimising the mean-square-error loss optimises for fractional (rather than absolute) accuracy on the emulated power spectrum. Standardisation ensures more rapid training convergence (Wan 2019).
- In the second method we train a NN to learn the mapping between cosmological parameters and Principal Components of the power spectra, as shown in the right-hand panel of Fig. 2. Principal Component Analysis (PCA) is a linear dimensionality reduction technique which performs a singular value decomposition of the input signal and retains the $N_{\rm PCA}$ components with the highest variance. We perform a PCA decomposition of the spectra in our training dataset, which produces a set of basis functions $q_{\lambda,i}$, with $i \in 1, \ldots, N_{\rm PCA}$. In other words, we assume we can decompose the spectra as:

$$P_\lambda(\boldsymbol{\theta}; \boldsymbol{w}) = \sum_{i=1}^{N_{\rm PCA}} \alpha_i(\boldsymbol{\theta}; \boldsymbol{w}) q_{\lambda,i} \,, \tag{1}$$

where the coefficients α_i in the new basis $q_{\lambda,i}$ are the Principal Components. We train a NN to output estimates $\hat{\alpha}_i$ of the Principal Components α_i, given cosmological parameters $\boldsymbol{\theta}$ as input. Similarly to the power spectra components in the direct NN case, the PCA components are also standardised.

We report the implementation details of the neural networks and PCA in Appendix A, including details on the training procedure. We tested both emulation approaches on the cosmological power spectra of interest and found the former to be in general more accurate. Thus, we decided to use it for all power spectra with the exception of the cross temperature-polarisation $\left(C_\ell^{\rm TE}\right)$ and lensing potential $\left(C_\ell^{\phi\phi}\right)$ CMB power spectrum, which were emulated using the second method. The use of the PCA decomposition is indeed particularly convenient when, as in the CMB TE spectrum case, it is not possible to take the logarithm of the training power spectra, due to some negative values. We also observed a better performance of the emulator for the $\phi\phi$ spectrum when applying the PCA compression first; we argue this is due to the smaller values of the logarithmic spectra, which range from -6 to -20 and might therefore cause numerical issues if fed directly into the NN.

One of the key advantages of CosmoPower is that the emulators are trained on very broad parameter ranges, which we report in Table 1. The choice of such large parameter ranges is motivated by the desire to provide a tool that is as general as possible (see Sect. 5 for further comments on this point).

A common problem in existing emulators is that they are trained to provide good performance on a *fixed* choice of cosmological parameterisation, while Boltzmann solvers maintain the flexibility to choose among different input parameterisations. In addition, Boltzmann solvers allow for derived parameters to be computed *a posteriori*, so that one can explore degeneracies between different parameters without restrictions. For example, one cannot directly sample in both $\ln 10^{10} A_{\rm s}$ and σ_8, as these two parameters are related; choosing to sample one or the other may in fact even have some effect on the posterior distribution (see e.g. Joachimi et al.

[1] https://simonsobservatory.org/
[2] https://www.euclid-ec.org/
[3] https://www.lsst.org/
[4] https://roman.gsfc.nasa.gov/

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

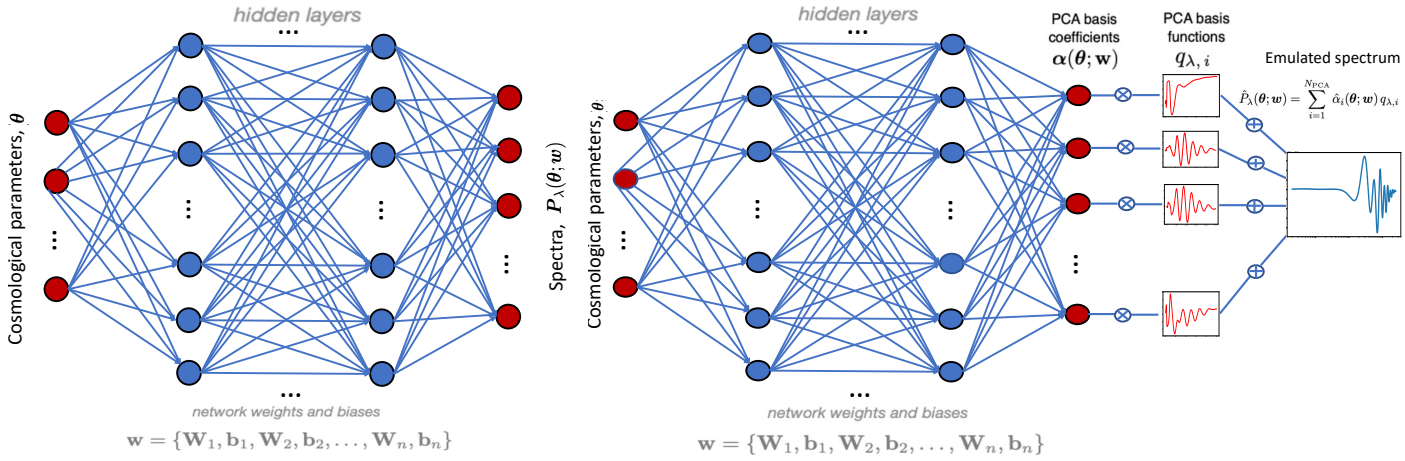

Figure 2. Schematic of the neural network emulation implemented in CosmoPower. A neural network is trained to learn the mapping between cosmological parameters and a) power spectra, b) coefficients in a Principal Component Analysis of the power spectra.

2021). The common strategy when performing cosmological inference is to choose one of these parameters for sampling, and let the Boltzmann solver compute the corresponding other one at each point in the posterior chain. In CosmoPower this is also possible, *without* re-training emulators from scratch. As we show in Sect. 3 and Appendix A3 for the case of $\ln 10^{10} A_s$ and σ_8, one can choose to sample the former or the latter, and obtain the other one with a GP. We refer the reader to Appendix A3 for details on the implementation of such a GP (see also Mootoovaloo et al. 2020 for a similar application of GPs to obtain derived values of σ_8).

3 LARGE-SCALE STRUCTURE

3.1 Theory

Here we briefly summarise the theory underlying two-point statistics analyses of LSS surveys, following a notation similar to that of Asgari et al. (2021); Heymans et al. (2021); Joachimi et al. (2021). A flat ΛCDM model is assumed throughout the paper. Extensions of our emulators to beyond-ΛCDM cosmologies will be explored in future work (see Sect. 5 for details).

LSS analyses target the shear and clustering signal of the observed galaxies, including the shear-clustering cross-correlation, in what is typically referred to as '3x2pt' analysis (Joachimi & Bridle 2010). Angular power spectra of shear and clustering statistics can be expressed as integrals of the matter power spectrum $P_{\delta\delta}(k, z)$ along the line of sight, weighted by kernel functions:

$$C_{ij}^{\rm ab}(\ell) = \int_0^{\chi_{\rm H}} {\rm d}\chi \, \frac{W_i^{\rm a}(\chi)\, W_j^{\rm b}(\chi)}{\chi^2} P_{\delta\delta}\left(\frac{\ell + 1/2}{\chi}, z\right) , \tag{2}$$

where the indices {a, b} can be assigned the labels {γ, I, n}, denoting contributions from cosmic shear, galaxy intrinsic alignment and galaxy positions, respectively. The integral in Eq. (2) is performed along the line of sight up to the Hubble radius $\chi_{\rm H} = c/H_0$, with c the speed of light and H_0 the Hubble constant. χ denotes the comoving distance, which is a function of the redshift z (a dependence implicitly assumed in Eq. 2 for ease of notation). The Limber projection (Kaiser 1992; LoVerde & Afshordi 2008) connects the wavenumber k and redshift z at which the matter power spectrum $P_{\delta\delta}(k, z)$ is evaluated in that integral, so that given a multipole ℓ and a redshift z (corresponding to a comoving distance χ) the matter power spectrum is evaluated at wavenumber $k = (\ell+1/2)/\chi(z)$. We note that, while we will restrict ourselves to Limber-approximated spectra in this paper, the emulation framework in CosmoPower can be equally applied to accelerate the computation of non-Limber projected spectra, which we plan to investigate in future work (see Sect. 5 for a discussion).

The weighting functions W can be written as

$$W_i^{\gamma}(\chi) = \frac{3\, H_0^2\, \Omega_{\rm m}}{2\, c^2} \frac{\chi}{a} \int_{\chi}^{\chi_{\rm H}} {\rm d}\chi' \; n_i(\chi') \; \frac{\chi' - \chi}{\chi'} \; , \tag{3}$$

$$W_i^{\rm I}(\chi) = -A_{\rm IA} \left(\frac{1+z}{1+z_{\rm pivot}}\right)^{\eta_{\rm IA}} \frac{C_1\, \rho_{\rm cr}\, \Omega_{\rm m}}{D(\chi)} \; n_i(\chi) \; , \tag{4}$$

$$W_i^{\rm n}(\chi) = b_i \, n_i(\chi) \, , \tag{5}$$

where $\Omega_{\rm m}$ is the total matter density parameter, a is the scale factor, $n_i(\chi)$ denotes the redshift distribution for redshift bin i, $D(\chi)$ is the linear growth factor, $\rho_{\rm cr}$ is the critical density, C_1 is a constant, $z_{\rm pivot}$ is an arbitrary redshift set to 0.3, while $A_{\rm IA}$ and $\eta_{\rm IA}$ are two free parameters, allowing for a scaling in amplitude and redshift, respectively, of the intrinsic alignment contribution. The linear galaxy bias coefficients b_i are also left free to vary in the inference pipeline.

The observed angular two-point statistics of galaxy ellipticities ϵ for tomographic redshift bins i and j, as a function of the angular multipole ℓ, can be written as

$$C_{ij}^{\epsilon\epsilon}(\ell) = C_{ij}^{\gamma\gamma}(\ell) + C_{ij}^{\gamma{\rm I}}(\ell) + C_{ij}^{{\rm I}\gamma}(\ell) + C_{ij}^{\rm II}(\ell) \, , \tag{6}$$

i.e. as a sum of a pure cosmic shear contribution and contaminants resulting from the intrinsic alignment of galaxies, which also affects the angular power spectrum of the cross-correlation between galaxy ellipticities ϵ and positions n (usually referred to as 'galaxy-galaxy lensing'):

$$C_{ij}^{{\rm n}\epsilon}(\ell) = C_{ij}^{{\rm n}\gamma}(\ell) + C_{ij}^{n{\rm I}}(\ell) \, , \tag{7}$$

while the galaxy clustering power spectrum $C_{ij}^{\rm nn}(\ell)$ is not affected by intrinsic alignments and, assuming no *magnification bias* (Duncan et al. 2013; von Wietersheim-Kramsta et al. 2021), can be explicitly written as

$$C_{ij}^{\rm nn}(\ell) = b_i b_j \int_0^{\chi_{\rm H}} {\rm d}\chi \, \frac{n_i(\chi)\, n_j(\chi)}{\chi^2} \, P_{\delta\delta}\left(\frac{\ell + 1/2}{\chi}, z\right) . \tag{8}$$

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

Parameter	LSS emulator range	CMB emulator range
ω_b	[0.01875, 0.02625]	[0.005, 0.04]
ω_{cdm}	[0.05, 0.255]	[0.001, 0.99]
h	[0.64, 0.82]	[0.2, 1.0]
τ_{reio}	fixed	[0.01, 0.8]
n_s	[0.84, 1.1]	[0.7, 1.3]
$\ln 10^{10} A_s$	[1.61, 3.91]	[1.61, 5]
c_{min}	[2, 4]	fixed
η_0	[0.5, 1]	fixed

Table 1. Ranges of validity of our emulators. The uniform prior distributions that we use at inference time to test our emulators share the same ranges. The LSS and CMB ranges are also the prior ranges assumed by the KiDS-1000 and Planck 2018 analysis, respectively. Parameters denoted as *fixed* in the LSS or CMB column are not considered as inputs to the neural networks that emulate the relevant power spectra, since the power spectra dependence on those parameters is negligible.

Two-point statistics measured by LSS surveys are typically linear transformations of the theoretical power spectra given by Eqs. 6, 7, 8, such as band-power estimates (Schneider et al. 2002; van Uitert et al. 2018). We refer the reader to Asgari et al. (2021) for a review of different measured two-point statistics.

3.2 Emulating the matter power spectrum

Eq. (2) clearly indicates that the prediction of the matter power spectrum $P(k, z)$ is central in the computation of the two-point statistics of cosmic shear, galaxy-galaxy lensing and galaxy clustering. Boltzmann codes can perform practically exact computations (up to numerical accuracy) of the matter power spectrum predicted by the linear theory of gravitational instability, which we denote as $P^{L}_{\delta\delta}(k, z)$. As nonlinearities become more important, linear theory breaks down. We will write the full, nonlinear spectrum as the product of the linear one $P^{L}_{\delta\delta}(k, z)$ and a nonlinear correction, which we label $P^{NL-CORR}_{\delta\delta}(k, z)$:

$$P_{\delta\delta}(k, z) = P^{L}_{\delta\delta}(k, z)\, P^{NL-CORR}_{\delta\delta}(k, z)\,. \tag{9}$$

Nonlinear corrections become important at small scales, corresponding to large wavenumbers k. Their modelling as a function of cosmological parameters is uncertain and further complicated by baryonic effects, whose impact on the nonlinear matter power spectrum induces important, yet not fully understood modifications to the nonlinear power on small scales (see e.g. Chisari et al. 2019, for a review). In recent years, the HMcode software developed by Mead et al. (2015, 2016, 2021) has found widespread use in LSS analyses, as a way to predict the nonlinear power spectrum while taking into account baryonic effects. HMcode is a modified halo model which includes baryonic contributions as opposed to, for example, the fitting function HaloFit (Smith et al. 2003; Takahashi et al. 2012; Bird et al. 2012). In this work we consider the latest version of the HMcode software (Mead et al. 2021). We focus on two of its free parameters, c_{min} and η_0, denoting the minimum halo concentration and the halo bloating, respectively. We train one emulator for the linear power spectrum $P^{L}_{\delta\delta}(k, z)$ and one for the nonlinear correction $P^{NL-CORR}_{\delta\delta}(k, z)$. We also experimented emulating directly the full power spectrum, but noticed increased performance when separating the linear contribution from the nonlinear correction. We observe this split helps the emulator isolate and learn more efficiently the effect of the nonlinear parameters c_{min} and η_0. For both linear and nonlinear contribution, we emulate the power spectrum at 420 k values in the range $[10^{-5}, 10]\,\mathrm{Mpc}^{-1}$. The redshift z is varied over the range $[0, 5]$ and treated as an additional input parameter for the emulator. We use $\sim 1.8 \cdot 10^5$ spectra for our training set and leave $\sim 2 \cdot 10^4$ spectra for our testing set. We verified that this fraction of parameter samples randomly selected for testing purposes from the training set still represents a homogeneous and uniformly distributed sample for each parameter.

The left- and right-hand panels of Fig. 4 report the emulation accuracy on the testing set for the linear power spectrum and nonlinear correction, respectively. We use percentile plots to show the statistical behaviour of the emulator accuracy throughout the testing set, as a function of the wavenumber k. For the linear power, 99% of the testing power spectra are emulated with differences smaller than 0.1% of their real value across the entire wavenumber range considered. For the nonlinear correction, 99% of the spectra are emulated with less than 0.4% error. As expected, the percentage difference in the nonlinear correction is typically about an order of magnitude larger than the linear one, thus dominating the total error. We can see how the error on the nonlinear correction increases for $k \gtrsim 1\,\mathrm{Mpc}^{-1}$, i.e. on highly non-linear scales. This was expected, as numerical computations performed by Boltzmann codes at these scales are more uncertain in the first place. One way to ensure increased numerical stability in these computations is to ask for a maximum k value k_{max} at which nonlinearities are computed that is well above the actual maximum k at which the power spectrum is required. In our analysis we set $k_{max} = 100\,\mathrm{Mpc}^{-1}$, while we only use the matter power spectrum up to $k = 10\,\mathrm{Mpc}^{-1}$.

These accuracy plots already show excellent performance of the emulators in obtaining high-fidelity predictions for the matter power spectrum. However, a full inference analysis is required to thoroughly test our emulators. As further discussed in Sect. 5, we remark here that this is the only way to completely test the validity of an emulation approach such as the one developed in CosmoPower. Crucially, different accuracy emulation levels are required for different types of analyses for which the emulators are being developed. CosmoPower is a tool designed to be adequate for Stage IV surveys: as such, we need to test the performance of CosmoPower on a simulated Stage IV inference pipeline. This is what we show in Sect. 3.4 with the simulated analysis of a *Euclid*-like survey.

Before showing those results, we present in Sect. 3.3 an application to a Stage III analysis, namely a joint weak lensing and galaxy clustering analysis from 450 deg^2 of the KiDS survey. We refer the reader to Appendix B for another, more recent Stage III application, namely a cosmic shear analysis of 1000 deg^2 from the KiDS survey. The goal of showing these Stage III results is to demonstrate that while CosmoPower is explicitly designed to aid inference analyses from Stage IV surveys, it can clearly benefit those already ongoing from Stage III experiments too. In the latter, CosmoPower can indeed provide contours in a matter of a few minutes without any Graphic Processing Unit acceleration (see Sect. 5 for a discussion on this point). In addition, in showing results for Stage III analyses we would like to emphasise the 'train-once-use-repeatedly' nature of CosmoPower. Our emulators are designed to be trained only once, so that the end users do not need to re-train any of the models, as long as the emulators are employed assuming the cosmological model and range indicated in Table 1. For example, in the KiDS and *Euclid*-like analyses shown in Sect. 3.3 and 3.4, the emulators for

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

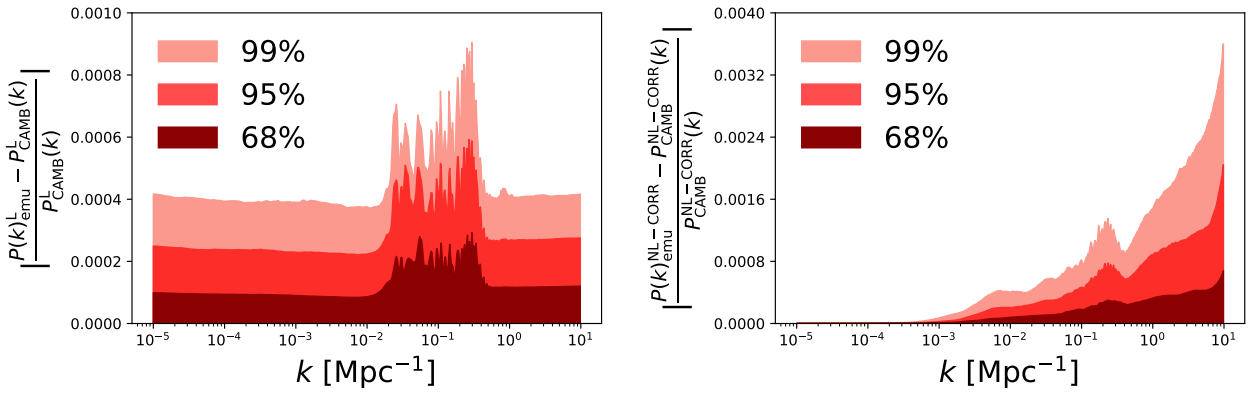

Figure 4. Matter power spectrum emulation accuracy for a) the linear power spectrum and b) the nonlinear correction, as measured on the $\sim 2 \cdot 10^4$ spectra of the testing set for our emulators. *Dark red*, *red* and *salmon* areas enclose the 68, 95 and 99 percentiles of the fractional absolute emulator error, respectively, as a function of wavenumber k. By construction, the redshift z is an input parameter for the emulators. Therefore, these percentile curves are computed with spectra evaluated at redshifts $z \in [0, 5]$, i.e. spanning the whole range of validity of our emulators.

Parameter	Prior range
ω_b	[0.01875, 0.02625]
ω_{cdm}	[0.05, 0.255]
h	[0.64, 0.82]
n_s	[0.84, 1.1]
$\ln 10^{10} A_s$	[1.61, 3.91]
c_{min}	[2, 4]
A_{IA}	[-6, 6]
b_{z_1}	[0.1, 5]
b_{z_2}	[0.1, 5]

Table 2. Prior ranges for the 3x2pt analysis of the KiDS-450 and GAMA surveys. Prior distributions are all taken to be uniform across these ranges. For the cosmological parameters and the baryonic feedback parameter c_{min} the prior range corresponds to the range of validity of our emulators (cf. Table 1).

the linear power spectrum and nonlinear correction are the *same* for both analyses.

3.3 Validation on the KiDS-450+GAMA 3x2pt likelihood

van Uitert et al. (2018) performed a 3x2pt analysis of power spectra from 450 deg^2 of the KiDS survey (KiDS-450) with overlapping galaxy clustering spectroscopic measurements from the Galaxy And Mass Assembly survey (GAMA, Driver et al. 2009, 2011; Liske et al. 2015). Here, we repeat their analysis using an inference pipeline entirely embedded within the cosmological sampler MontePython (Brinckmann & Lesgourgues 2019). This pipeline is an adaptation to ΛCDM of the likelihood developed in Spurio Mancini et al. (2019)[5] for the same survey configuration, but for a Horndeski gravity scenario. In Spurio Mancini et al. (2019) that likelihood, run in ΛCDM, was already shown to provide excellent agreement with the fiducial pipeline developed by van Uitert et al. (2018) for the sampler CosmoMC. In addition to the data vector, the redshift distributions and analytical covariance matrix are also exactly the same as those used in van Uitert et al. (2018) and Spurio Mancini et al. (2019).

The modelling of the power spectra for each cosmological probe follows the same prescription as in van Uitert et al. (2018); Spurio Mancini et al. (2019) and described in Eqs. 6, 7, 8. In particular, we consider one intrinsic alignment amplitude A_{IA}, we set $\eta_{IA} = 0$ and we include one linear galaxy bias coefficient b_{z_i} for each of the two spectroscopic bins of the GAMA survey. The prior ranges are reported in Table 2. They are the same ones used in van Uitert et al. (2018); Spurio Mancini et al. (2019), except for those cosmological parameters ($\omega_{cdm}, \omega_b, n_s$), whose prior range was restricted in the more recent KiDS-1000 analysis (Asgari et al. 2021; Heymans et al. 2021; Joachimi et al. 2021). As explained in Joachimi et al. (2021), the KiDS collaboration decided to restrict those prior ranges in their analyses, to make sure that the parameters are more physically motivated; we adhere to this choice. For the cosmological parameters in Table 2 and the baryonic feedback parameter c_{min}, the prior ranges coincide with the range of validity of our emulators (cf. Table 1). The halo bloating parameter η_0 is fixed to $\eta_0 = 0.98 - 0.12 c_{min}$, as implemented in Class following Mead et al. (2015).

The recent KiDS-1000 analyses sample the cosmological parameter $S_8 = \sigma_8 \sqrt{\Omega_m/0.3}$, instead of $\ln 10^{10} A_s$ used in van Uitert et al. (2018) and other past KiDS analyses (Hildebrandt et al. 2016; KÃűhlinger et al. 2017; Hildebrandt et al. 2020). Here, we stick to the choice of $\ln 10^{10} A_s$ as in van Uitert et al. (2018), for a more direct comparison. In Appendix B, instead, we show results for KiDS-1000 with S_8 among the sampled parameters. In all cases, we always show plots for all of $\ln 10^{10} A_s$, σ_8 and S_8: the parameters that have not been used directly in sampling have been obtained as derived parameters with a postprocessing GP, as explained in Sect. 2 and Appendix A3.

Fig. 5 shows the comparison between contour plots obtained sourcing power spectra from Class and those obtained running CosmoPower to replace the Boltzmann code. In both analyses,

[5] available at https://github.com/alessiospuriomancini/KiDSHorndeski

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

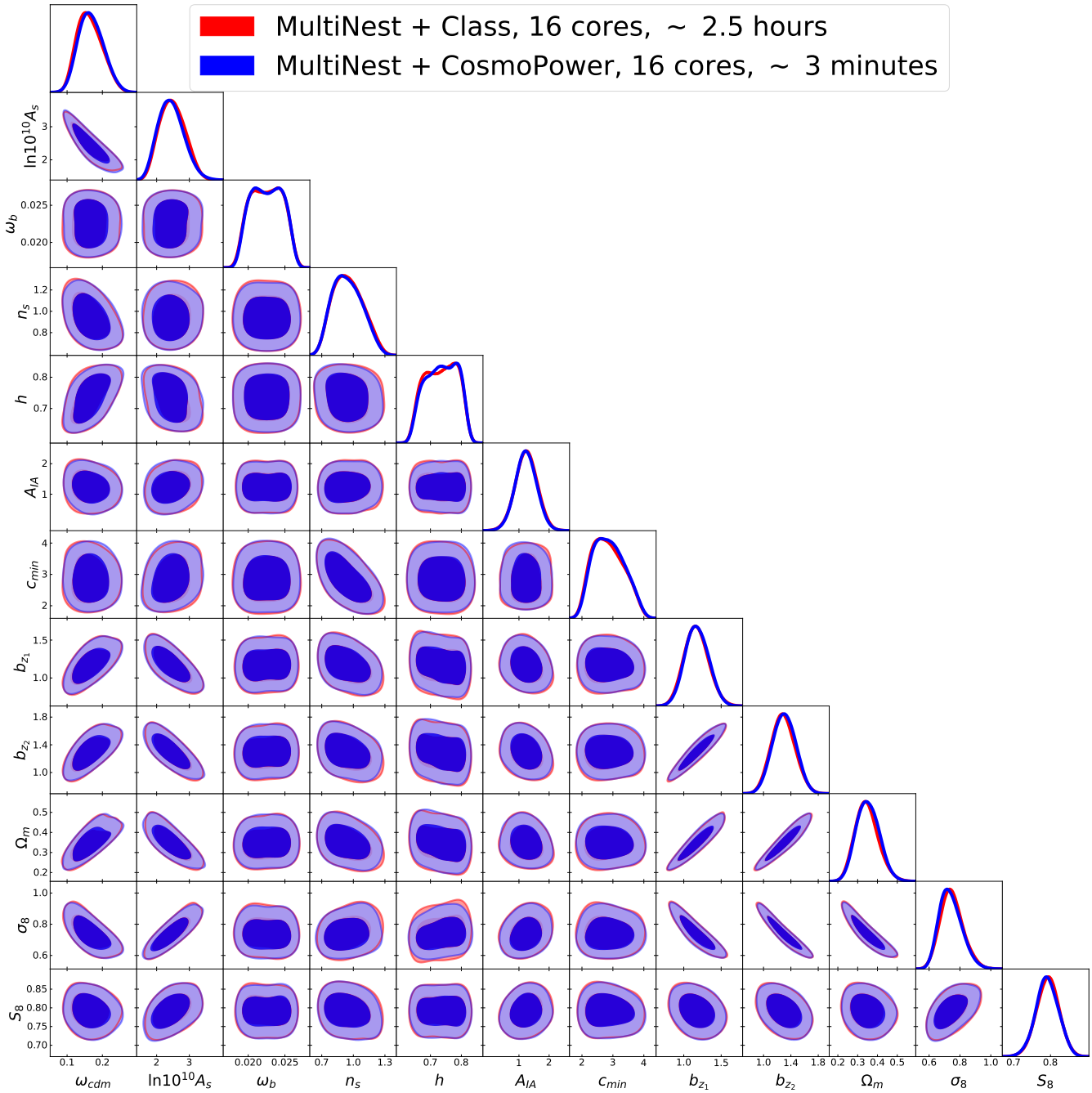

Figure 5. Contours for the 3x2pt analysis of the KiDS-450 and GAMA surveys. In *red* contours obtained with CLASS, in *blue* those obtained with COSMOPOWER.

we sample the posterior distribution with the MULTINEST algorithm (Feroz et al. 2009), as implemented in the Python wrapper PYMULTINEST (Buchner et al. 2014). We run the pipelines with parallelisation across 16 Intel Xeon E5640 cores. The posteriors with our emulator are obtained in under 3 minutes, compared with the ∼ 2.5 hours required when using CLASS. This produces a speed-up factor of approximately 50.

We also compare the values of the log-evidence $\log \mathcal{Z}$ obtained with MULTINEST in the two runs, and find $\log \mathcal{Z} = -73.77 \pm 0.19$ and $\log \mathcal{Z} = -73.79 \pm 0.19$ for the run sourcing power spectra from CLASS and COSMOPOWER, respectively. The good agreement between these evidence values is another, arguably stronger indicator of the accuracy of our emulators. In order to get an accurate estimate of this quantity one needs indeed accurate values of the posterior distribution across the whole prior range, while for parameter estimation it may be sufficient to get accurate estimates around the peak of the distribution.

We also applied COSMOPOWER to a more recent KiDS analysis, namely that considering cosmic shear band powers from 1000 deg^2 of the survey, as presented in Asgari et al. (2021). We use this analysis as another test case for the performance of our emulator, finding similar speed-up factors provided by replacing the Boltzmann code CLASS with COSMOPOWER. We refer the reader to Appendix B for details and plots for this analysis.

ORIGINAL

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

Parameter	Prior range	Fiducial value
ω_b	[0.01875, 0.02625]	0.02242
ω_{cdm}	[0.05, 0.255]	0.11933
h	[0.64, 0.82]	0.6766
n_s	[0.84, 1.1]	0.9665
$\ln 10^{10} A_s$	[1.61, 3.91]	3.047
c_{min}	[2, 4]	2.6
η_0	[0.5, 1]	0.7
A_{IA}	[-6, 6]	0.8
η_{IA}	[-6, 6]	0
D_{z_i}, $i = 1 \ldots 10$	$\mathcal{N}(0, 10^{-4})$	0

Table 3. Prior ranges and fiducial values of the cosmological parameters for the simulated *Euclid*-like cosmic shear analysis. Prior distributions are all taken to be uniform across these ranges, except for the redshift mean shifts D_{z_i} which have a Gaussian prior with mean 0 and standard deviation 10^{-4}. For the cosmological parameters and the baryonic feedback parameters c_{min}, η_0 the prior range corresponds to the range of validity of our emulators (cf. Table 1).

3.4 Validation on the Euclid-like cosmic shear likelihood

We re-use the same emulator employed in Sect. 3.3 to perform a full inference analysis of a simulated cosmic shear survey for a typical Stage IV *Euclid*-like configuration. The prior ranges are reported in Table 3. For the cosmological and baryonic feedback parameters they correspond to the validity ranges of our emulators, reported in Table 1. The fiducial values of the parameters used to calculate the mock data vector are also reported in Table 3. 10 tomographic bins are equipopolated with galaxies following a distribution (Smail et al. 1995; Joachimi & Bridle 2010; Laureijs et al. 2011):

$$n(z) \propto z^2 \exp\{-(z/z_0)^{3/2}\} \tag{10}$$

with $z_0 = 0.64$. For the covariance matrix, we use a simple analytical Gaussian computation following Tutusaus et al. (2020), with a sky coverage $f_{sky} = 0.3$, a galaxy number density $n = 30$ arcmin^{-2} and an ellipticity dispersion $\sigma_\epsilon = 0.3$.

We implement the likelihood in the cosmological sampler Cobaya (Torrado & Lewis 2019; Torrado & Lewis 2021) and use the Polychord algorithm (Handley et al. 2015a,b) to sample the posterior distribution. We run the pipelines with parallelisation across 48 Intel Xeon E5640 cores and obtain a speed-up of approximately 50. Fig. 6 shows the excellent agreement between contours obtained with Class and CosmoPower. The log-evidence $\log \mathcal{Z} = -45.92 \pm 0.33$ computed for the run with spectra sourced from Class also compares favourably with that of the emulator run, $\log \mathcal{Z} = -45.99 \pm 0.34$.

4 COSMIC MICROWAVE BACKGROUND

4.1 Emulating CMB temperature, polarisation and lensing power spectra

The three main probes in the analysis of the CMB are the temperature power spectrum (C_ℓ^{TT}), the polarisation power spectrum (C_ℓ^{EE}), and the temperature-polarisation cross power spectrum (C_ℓ^{TE}). We use Camb to produce the training set, which consists of $\sim 5 \cdot 10^5$ power spectra for each probe, with their associated parameters. The parameters are sampled from the range indicated in Table 1 using Latin Hypercube Sampling. We verified that the specific method chosen for sampling parameter space is comparatively less important than the number of training points. We obtained essentially the same accuracy results in our analysis using alternatives to Latin Hypercube Sampling such as Orthogonal Sampling (Owen 1992) or even a simple uniform sampling for each parameter. This is due to the fact that differences between different sampling schemes tend to become negligible as the number of parameters becomes large, as it is in our case.

To emulate the CMB spectra, we consider both methods described in Sect. 2. We use the direct NN mapping for C_ℓ^{TT} and C_ℓ^{EE}, while we found that the C_ℓ^{TE} emulation is better performed by the second method, i.e. PCA compression followed by a NN, due to its zero-crossing values. To show the flexibility of our approach, we also train a PCA plus NN emulator on the lensing potential power spectrum $C_\ell^{\phi\phi}$; note, however, that this probe is not included in the likelihood analysis performed in the next section.

The accuracy of the emulators over the ℓ range is measured with respect to the instrumental noise given by the upcoming Simons Observatory (Ade et al. 2019) combined with cosmic variance. In particular, we calculate the emulation error as:

$$\frac{|C_{\ell,\text{emulated}}^{\{TT,EE,TE,\phi\phi\}} - C_{\ell,\text{true}}^{\{TT,EE,TE,\phi\phi\}}|}{\sigma_{\ell,\text{CMB}}^{\{TT,EE,TE,\phi\phi\}}}, \tag{11}$$

where

$$\sigma_{\ell,\text{CMB}}^{\{TT,EE,\phi\phi\}} = \sqrt{\frac{2}{f_{sky}(2\ell+1)}} \left(C_{\ell,\text{true}}^{\{TT,EE,\phi\phi\}} + N_\ell^{\{TT,EE,\phi\phi\}}\right), \tag{12}$$

$$\sigma_{\ell,\text{CMB}}^{TE} = \sqrt{\frac{1}{f_{sky}(2\ell+1)}} \times \sqrt{C_{\ell,\text{true}}^{TE} C_{\ell,\text{true}}^{TE} + \left(C_{\ell,\text{true}}^{TT} + N_\ell^{TT}\right)\left(C_{\ell,\text{true}}^{EE} + N_\ell^{EE}\right)}, \tag{13}$$

$f_{sky} = 0.4$, and $N_\ell^{\{TT,EE,TE,\phi\phi\}}$ refers to the Simons Observatory goal noise curves due to instrumental and atmospheric effects (Ade et al. 2019)[6].

We validate our emulators using two sets containing $\sim 2 \cdot 10^4$ spectra each. The first one corresponds to cosmological parameters sampled from a restricted range, corresponding to 5σ intervals around the best fit values from *Planck* (Aghanim et al. 2020); the results are reported in Fig. 8 for all probes. As one can see, the distribution of the quantiles is very tight, and almost always 99% of the spectra are within less than $0.1\sigma_{CMB}$ at all ℓ values.

The second set of spectra corresponds to the same range the emulators were trained on, i.e. the intervals in Table 1; the results are reported in Appendix C. Unsurprisingly, the errors are slightly larger, reaching $0.2\sigma_{CMB}$; however, in the next section we show that this level of accuracy for spectra emulation is sufficient to provide accurate and unbiased inference of cosmological parameters in a CMB analysis.

[6] https://github.com/simonsobs/so_noise_models

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

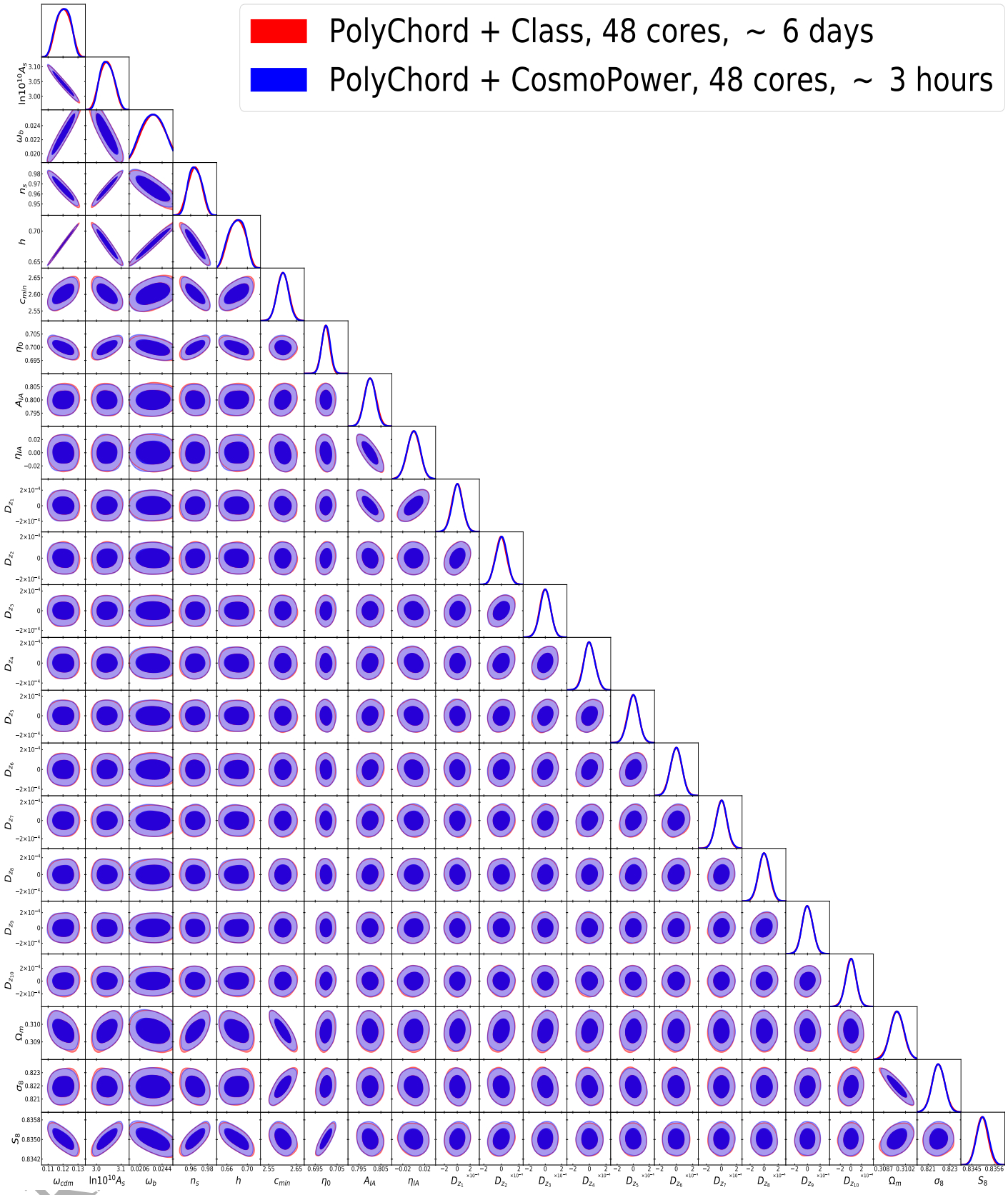

Figure 6. Contours for a simulated cosmic shear analysis of a *Euclid*-like Stage IV surveys. In *red* contours obtained with CLASS, in *blue* those obtained with COSMOPOWER.

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

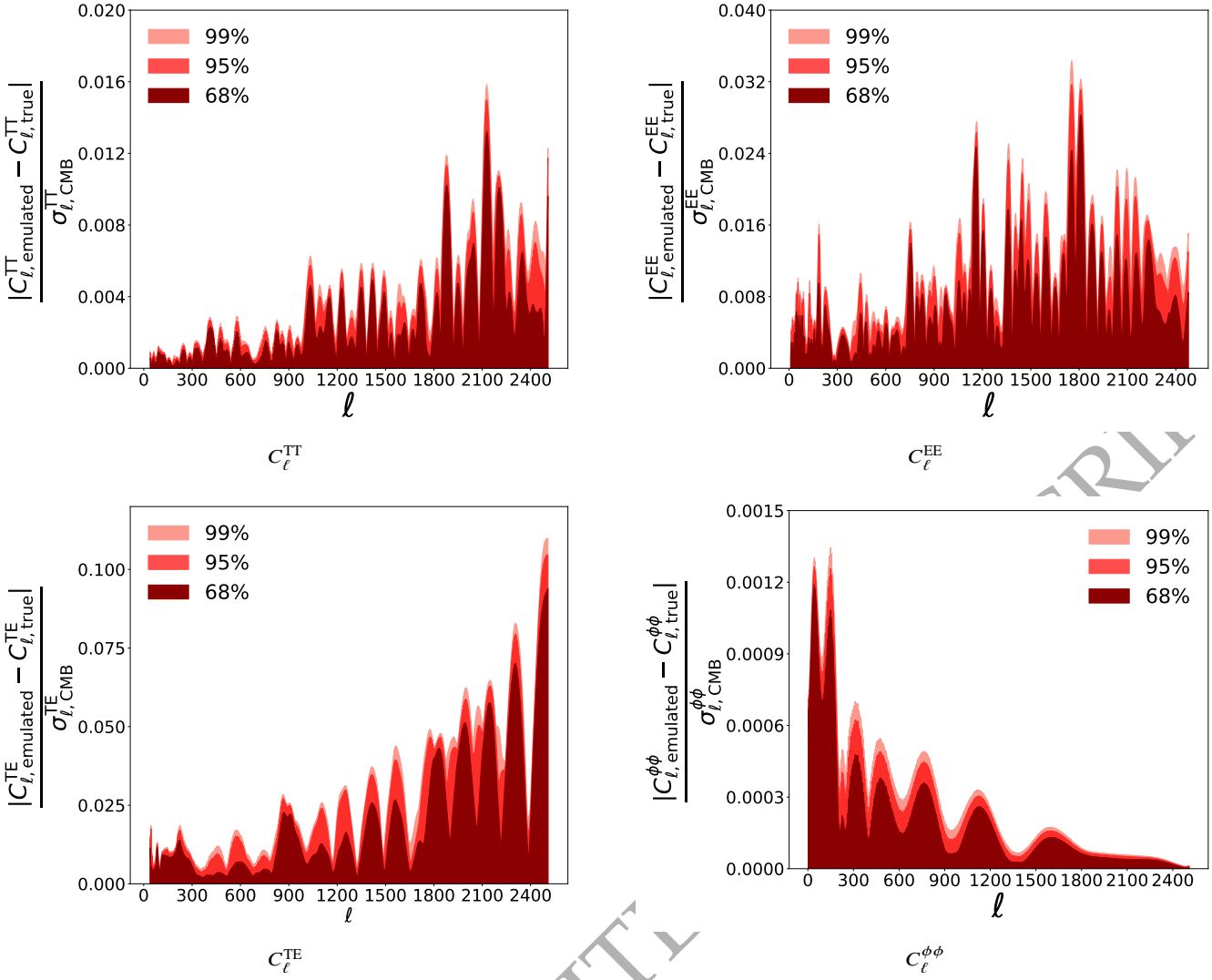

Figure 8. CMB power spectra emulation accuracy on the 5σ range test set for a) the temperature power spectrum, b) the polarisation power spectrum, c) the temperature-polarisation cross power spectrum, d) the lensing potential power spectrum. The emulation error is defined with respect to both instrumental and statistical noise, and is defined in Eqs. (11-13). *Dark red*, *red* and *salmon* areas enclose the 68, 95 and 99 percentiles of the test set. Details of the neural models are reported in Appendix A1.

Parameter	Prior range
ω_b	[0.005, 0.04]
ω_{cdm}	[0.001, 0.99]
h	[0.2, 1.0]
τ_{reio}	[0.01, 0.8]
n_s	[0.7, 1.3]
$\ln 10^{10} A_s$	[1.61, 5]
A_{Planck}	$\mathcal{N}(1, 0.0025)$

Table 4. Prior ranges for the *Planck* analysis. Prior distributions are all taken to be uniform across these ranges, except for the nuisance parameter A_{Planck} which has a Gaussian prior with mean 1 and standard deviation 0.0025. The ranges on the cosmological parameters correspond to the ranges of validity of our emulators (cf. Table 1).

4.2 Validation on the Planck 2018 likelihood

After assessing the accuracy of our emulators by looking at the residuals of their predictions on the testing set, we performed the final validation check by using the emulators to speed up parameter estimation in a *Planck* CMB inference pipeline (Aghanim et al. 2020). We considered the pure Python implementation of the Planck 2018 plik-lite likelihood available from Prince & Dunkley (2019)[7], which is pre-marginalised over a series of nuisance parameters. The only remaining calibration parameter is a multiplicative correction factor A_{Planck}. The prior ranges for all of the parameters varied in the analysis are reported in Table 4.

To further showcase the strength of CosmoPower, to draw

[7] https://github.com/heatherprince/planck-lite-py. Note that a similar pure Python implementation is available in the cosmological sampler Cobaya, https://github.com/CobayaSampler/cobaya/blob/master/cobaya/likelihoods/base_classes/planck_pliklite.py

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

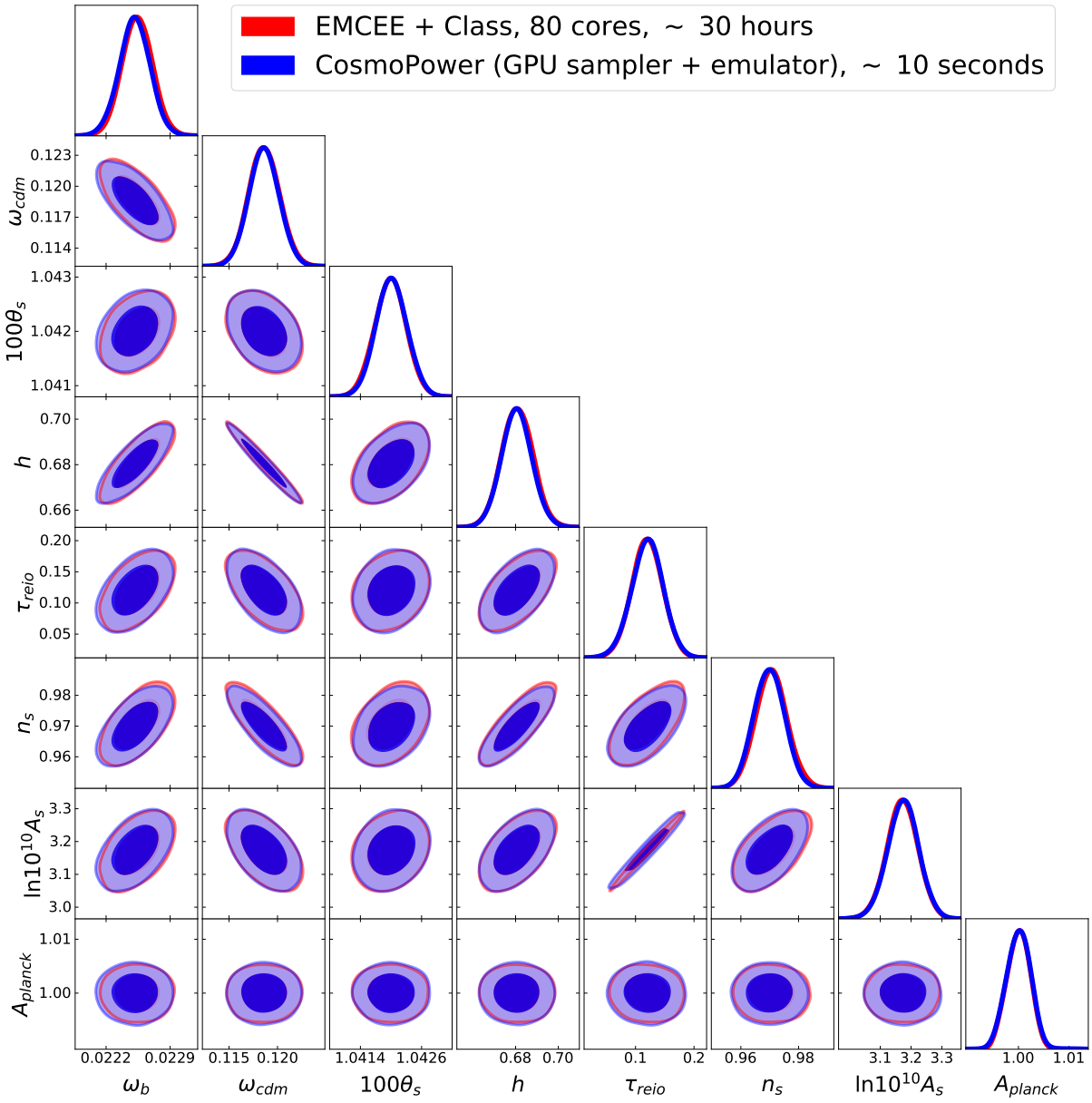

Figure 9. Planck 2018 3x2pt analysis considering $C_\ell^{\rm TT}$, $C_\ell^{\rm EE}$ and $C_\ell^{\rm TE}$. The red contours are obtained in $\sim 1.2 \cdot 10^5$ seconds on 80 CPU cores using CLASS, while the blue contours take roughly 10 seconds on a single GPU using our neural emulators. Note that the constraints on $100\theta_{\rm S}$ are derived from the rest of the samples using a Gaussian Process.

from the posterior distribution we use a parallelised affine-invariant MCMC sampler[8], that is a parallel, GPU-compatible TENSORFLOW implementation of a variant of the algorithm underlying the EMCEE sampler (Foreman-Mackey et al. 2013), based on Goodman & Weare (2010). This sampler fully exploits the neural emulators by allowing large batches of likelihood calls to be performed in parallel. Using a single GeForce GTX 1080 GPU, we can obtain the full contours in ~ 10 seconds, while the contours using CLASS and EMCEE take a total wall-clock time of $\sim 1.2 \cdot 10^5$ seconds using 80 CPU cores, for a final speedup of $O(10^4)$. Moreover, we train a single GP, as described in Sect. 2, to derive constraints on $100\theta_{\rm S}$ as well, in the same fashion of the σ_8 contours obtained in Sect. 3. We show the posterior contours in Fig. 9.

To obtain log-evidence estimates we re-run both likelihoods with MULTINEST and find $\log\mathcal{Z} = -313.72 \pm 0.15$ and $\log\mathcal{Z} = -313.79 \pm 0.15$ for the run with CLASS and COSMOPOWER, respectively.

5 CONCLUSIONS

We presented COSMOPOWER, a suite of cosmological power spectra emulators developed to accelerate by orders of magnitude parameter estimation from Large-Scale Structure (LSS) and Cosmic Microwave Background (CMB) surveys. COSMOPOWER emulates matter and CMB power spectra computed by Boltzmann codes such as

[8] https://github.com/justinalsing/affine

ORIGINAL UNPUBLISHED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

Camb and Class. Sourcing power spectra from Boltzmann codes is the computational bottleneck for two-point statistics analyses of cosmological fields; CosmoPower bypasses this step, providing orders-of-magnitude acceleration to the inference pipeline. Power spectra emulation is performed using a neural network (NN) to parameterise the mapping between cosmological parameters and power spectra or their Principal Component Analysis (PCA) coefficients.

In this paper, we presented emulators for the linear and non-linear matter power spectrum, as well as for the CMB temperature, polarisation and lensing power spectrum. We showcased the performance of CosmoPower with applications to Stage III and simulated Stage IV surveys, including: a 3x2pt and cosmic shear analysis of the Kilo-Degree Survey (KiDS), a mock *Euclid*-like cosmic shear analysis and a *Planck* 2018 joint temperature and polarisation analysis. In all of these cases, the power spectra emulators provided unbiased cosmological inference, at a fraction of the time required by the same pipelines run with power spectra sourced from Boltzmann solvers. In the following, we summarise the main properties of CosmoPower, compare its performance with that of other emulators, and discuss some of its planned future extensions.

5.1 Key properties of CosmoPower

The following key properties of CosmoPower make it an invaluable tool for application to future Stage IV analyses.

- **Speed-up**. First and foremost, the use of CosmoPower to replace Boltzmann codes in likelihood evaluations provides an impressive speed-up factor. In the applications considered in this paper, CosmoPower provided an acceleration factor up to $O(10^4)$ with respect to standard analyses with Boltzmann codes. These numbers refer to the full inference pipeline; if we restrict to timing a single power spectrum evaluation, the speedup increases even further up to $O(10^5)$, respectively. These numbers are expected to increase as we extend CosmoPower to cosmologies beyond the flat ΛCDM model which was assumed throughout this paper. In this sense, the acceleration quoted in this analysis is to be regarded as a lower bound for the speed-up achievable with CosmoPower.

- **GPU acceleration**. Our emulators are based on neural networks implemented in TensorFlow. As such, they benefit from an additional speed-up when run on Graphics Processing Units (GPU) or Tensor Processing Units (TPU)[9].

The speed-up calculated for the full inference pipeline differs from the one computed for a single power spectrum evaluation because a single likelihood evaluation is slower than a power spectrum evaluation: computing the projected angular spectra of the LSS probes, or binning the CMB spectra as performed in the *Planck* likelihood, requires a series of numerical operations. These parts of the likelihood evaluation are computationally intensive regardless of the method employed to source power spectra. This is particularly true for Stage IV LSS surveys, which will have a high number of redshift bins and hence will require the computation of a high number of bin cross-correlations (cf. Eq. 2). These expensive loops in the likelihood evaluation can be massively accelerated by implementing the likelihood in TensorFlow or JAX (Frostig et al. 2018) and running the inference pipeline on GPUs.

[9] both freely available on Google Colab `http://colab.research.google.com/`. In the CosmoPower GitHub repository we provide Jupyter notebook examples to run CosmoPower on Colab GPU.

In this paper we showed how running CosmoPower in a pure TensorFlow-based version of the *Planck* likelihood, embedded within a framework for cosmological inference also implemented in TensorFlow, provided contours in ∼ 10 seconds. Running an inference pipeline with the *Planck* likelihood is a notoriously computationally intensive task: this example of speed-up achieved with a Tensorflow-based likelihood and power spectra clearly shows how beneficial the combination of highly optimised software for power spectra emulation and Bayesian posterior sampling can be. We advocate for moving towards implementations of cosmological inference software in TensorFlow or JAX, to fully exploit the power of highly optimised software that can be run on GPU and is auto-differentiable. The reason for this is the high number of nuisance parameters that is expected to be required to model the observables, in addition to the large size of the data vector. Particularly, if one desires to make use of the unprecedented amount of data provided by these surveys to investigate beyond-ΛCDM cosmologies, the implementation of cosmological inference frameworks leveraging GPU acceleration is of the utmost importance. CosmoPower aims at providing such a framework, incorporating not only trained emulators but also template TensorFlow-based likelihoods for LSS and CMB surveys that can be easily adapted for application to different surveys.

- **Full differentiability**. CosmoPower provides emulators for the power spectra that are based on neural networks and implemented in TensorFlow (Abadi et al. 2015). Thus, these emulators are by construction fully differentiable, a feature which makes them ideal for gradient-based inference, such as Hamiltonian Monte Carlo (Hajian 2007). If desired, they can also be used for instantaneous Fisher matrix computation and linear data compression with e.g. the MOPED algorithm (Heavens et al. 2000), leveraging the possibility of calculating derivatives with respect to the input parameters by auto-differentiation.

- **Accuracy**. The procedure followed to validate the accuracy of our emulators guarantees that they can be safely used for analyses of Stage IV surveys. Crucially, we verified this statement not only by checking the residuals between emulated and real power spectra in the testing set, but also by validating our emulators with full posterior inference analyses from state-of-the-art surveys, as well as from simulated Stage IV surveys. In carrying out this comparison at the contours level, we performed an additional validity check between CosmoPower and the Boltzmann codes Camb and Class. To train our models we used power spectra generated with Camb; instead, when comparing the CosmoPower contours against those obtained from a traditional inference pipeline sourcing power spectra from a Boltzmann code, we used Class for the latter. As shown in Figs. 5, 6 and 9, contours obtained with the emulators (trained on Camb) were always found in excellent agreement with the contours obtained from the inference pipelines sourcing power spectra from Class, including in the simulated Stage IV *Euclid*-like configuration. We also verified that replacing Class with Camb in the inference pipelines provides contours with a similar level of agreement: this means that the difference between CosmoPower predictions and Boltzmann-computed power spectra are not bigger than the differences between power spectra computed with different Boltzmann codes.

- **Parameter range**. The parameter range over which our models are trained is very large, covering the full Planck prior range in the CMB case and the full KiDS-1000 prior range in the LSS case. The combination of high accuracy and wide validity range allows the user of CosmoPower to safely replace Boltzmann codes with our

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

emulators when computing power spectra, even for those practical applications where high accuracy over broad prior ranges is crucial, such as posterior predictive cross-validation. The accuracy of our emulators even in extreme regions of the parameter space considered for their training is confirmed by the good agreement between log-evidence values obtained in the likelihood runs with power spectra sourced from CLASS and COSMOPOWER.

• **Flexibility**. By construction, COSMOPOWER emulates cosmological power spectra taking in input only those cosmological parameters that are part of the mapping between input cosmologies and output power spectra. This means that, for example, in the LSS case, the key emulated quantity is the matter power spectrum and not the cosmic shear, galaxy-galaxy lensing or galaxy clustering projected power spectra. The rationale behind this choice is that the angular spectra of cosmological probes are quantities derived from the matter power spectrum, by integrating it over a kernel which depends on the redshift distributions. In addition, contaminant terms due to e.g. intrinsic alignments are also obtained by integration of the matter power spectrum, and modulated by nuisance parameters which are not part of the cosmological model. By avoiding to include those additional parameters in the target mapping for the emulator, COSMOPOWER acquires a unique flexibility that makes it completely independent of astrophysical nuisance parameters, such as intrinsic alignment and galaxy bias parameters, that do not modify the matter power spectrum prediction. This means that our emulators do not need to be trained for different choices of these astrophysical parameters. In particular, no re-training is required if one wishes to implement different prescriptions for the modelling of contaminants such as intrinsic alignments, e.g. by inserting additional nuisance parameters, as long as those parameters do not modify the prediction for the matter power spectrum. A similar argument is applicable to the modelling of redshift distributions, which will likely require even more nuisance parameters than the mean shifts used in our simulated *Euclid*-like analysis (see e.g. Hasan et al. 2021). As an additional bonus, emulating the 3D matter power spectrum will allow us to investigate in future work the use of COSMOPOWER for emulating cosmological power spectra beyond the Limber approximation (see below).

• **Linear and nonlinear power spectra**. COSMOPOWER provides emulators for both linear power spectra and nonlinear correction factor. For the latter, the HMCODE prescription is currently implemented in the emulator. The HALOFIT model (which HMCODE is based on) is also available to the user. The separation between linear power spectrum and nonlinear correction factor is particularly useful as it allows us to integrate in COSMOPOWER additional models for nonlinearities as they become available. On the linear level, modified Boltzmann codes for beyond-ΛCDM models exist (for example HICLASS for Horndeski models; ZumalacÃąrregui et al. 2017) that provide linear predictions for the matter power spectrum in these extended cosmologies. As we extend COSMOPOWER to these models, we can add new emulators for linear power spectra trained on these modified Boltzmann codes.

• **"Train-once-use-repeatedly" approach and interface with cosmological samplers**. While we provide all the tools necessary to repeat the training if desired, we stress that this operation has already been performed and does not need to be repeated, as long as the emulators are employed assuming the cosmological model and range indicated in Table 1. In addition, COSMOPOWER can be called from all commonly used cosmological samplers. In this paper, for example, we used COSMOPOWER within the cosmological samplers MONTEPYTHON and COBAYA. The user of COSMOPOWER simply needs to write a likelihood for the LSS or CMB survey considered, and replace the call to the Boltzmann code, necessary to obtain the matter or CMB power spectra, with a call to COSMOPOWER.

• **Derived parameters**. Emulators developed in the literature usually provide a fixed parameterisation to emulate from. For example, if an emulator is trained using $\ln 10^{10} A_s$ it is not possible to get a prediction for a corresponding value of σ_8 or S_8. COSMOPOWER provides emulators trained on different combinations of parameters. For example, the 3x2pt KiDS-450+GAMA analysis was performed with an emulator trained on $\ln 10^{10} A_s$ as input parameter, while the KiDS-1000 analysis used σ_8 in input. In addition, COSMOPOWER also allows the user to post-process a sampled chain to obtain very efficiently derived parameters that were not originally sampled. Moreover, we provide Gaussian Processes (GPs) to obtain derived parameters that were not used as input to the emulators. The accuracy of these GPs was tested not only on a test set, but also against the actual contours obtained with the Boltzmann codes.

5.2 Comparison with previous work

Here we compare our emulators to other existing approaches to power spectra emulation. We start by noticing that COSMOPOWER provides an emulation framework for *both* LSS and CMB. To our knowledge, this is a unique feature, only partially shared by PICO and COSMONET (both, however, emulating matter transfer functions rather than power spectra). These two packages are not actively maintained nor trained with the same accuracy or across the same parameter ranges, which limits their applicability to Stage IV analyses. As far as the matter power spectrum is concerned, the methods closest to ours in terms of emulation are the one implemented in Aricò et al. (2021), even though limited to the linear power spectrum, and Agarwal et al. (2012), limited to HALOFIT nonlinearities.

We note that applying the emulator to a complete inference analysis from a simulated Stage IV survey, as done in our paper, is a necessary step to ensure that the newly developed tool can be safely applied in practical analyses. On the contrary, checking residuals in the testing set between predicted and real spectra is not a sufficient accuracy test. While an emulator may be performing with e.g. sub-percent accuracy at the level of residuals, this may still not be enough to retrieve unbiased cosmological contours, as we verified first-hand while testing COSMOPOWER. This is due to the fact that the accuracy threshold for the emulation can only be defined by the specific application for which these emulators are designed. In other words, it is the inference pipeline that dictates the accuracy threshold to be met by the emulator. In general, parameter estimation in Bayesian inference pipelines requires a certain level of accuracy in the observables computed, which in the specific case of cosmological two-point statistics analyses reflects into certain accuracy requirements in the power spectra computed by Boltzmann codes. Hence, we argue that the principled approach to validate an emulator accuracy is to compare its performance within an inference pipeline for a target experiment, which in our case is a Stage IV survey configuration. Note that, while testing COSMOPOWER, we experienced first-hand that emulators performing greatly on Stage III experiments failed in producing equally correct contours on a simulated Stage IV survey.

GP-based emulators such as the ones used in Mootoovaloo et al. (2020, 2022); Ramachandra et al. (2021); Ho et al. (2021) require fewer training samples than a NN emulation framework like COSMOPOWER; however, GPs also provide reduced speed-ups compared to NNs. On the other hand, GPs also

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

provide a way to propagate the uncertainty in their prediction to the final posterior distribution, whereas simple NNs like those implemented in this version of CosmoPower lack this feature. In future versions of CosmoPower we will investigate the use of Bayesian Neural Networks or ensemble NN predictions for this purpose.

Mootoovaloo et al. (2020) developed GP emulators of cosmic shear band powers for the KiDS-450 survey, with the option of compressing the band powers into MOPED coefficients and learning those coefficients with the GP instead of the band powers themselves. Similarly, Manrique-Yus & Sellentin (2019) developed NN emulators of the 3x2pt angular power spectra. Both of these approaches are constrained by the choice of the redshift distributions specific to the survey, which enters the expression of the angular power spectra. In addition, the method of Mootoovaloo et al. (2020) also relies heavily on the choice of nuisance parameters used to model the power spectra; these parameters need to be 'learnt' by their GP. CosmoPower is free from any of these restrictions: targeting emulation of the matter power spectra, our emulators are completely flexible to be used for any redshift distribution and choice of nuisance parameters. While the speed-up obtained by emulating the matter power spectrum may be smaller than that obtained from emulating the angular power spectra of the different probes, we believe this computational overhead can be avoided by rewriting the likelihood of interest in TensorFlow and running it on a GPU together with the emulators.

Finally, Albers et al. (2019) implemented an interesting NN-based acceleration of source functions in the calculation of CMB power spectra within the Boltzmann code Class. Emulating source functions provides great flexibility in the pipeline, as for example it allows one to compute higher-order correlators and non-linear transfer functions without retraining. On the other hand, emulation of full power spectra performed in CosmoPower provides greater speed-ups and is better suited for implementation of full inference pipelines on GPUs.

5.3 Future work

CosmoPower is an open-source package provided to the cosmological community as a tool to accelerate intensive computations within Bayesian inference pipelines of LSS and CMB surveys. In this paper we considered the emulation of power spectra, which represents the bottleneck for two-point statistics analyses of cosmological fields. However, this paper also marks the starting point of a longer-term project, with the goal of extending the CosmoPower framework to accelerate the forward-modelling of multiple cosmological observables with machine learning.

- **Higher-order statistics, systematics and beyond-Limber spectra**. We plan to train emulators for higher-order statistics such as the bispectrum. Not only does the bispectrum contain complementary cosmological information to the power spectrum (see e.g. Pyne & Joachimi 2021), but it is also required to calculate (computationally expensive) corrections to the power spectrum as those arising from dropping the reduced shear approximation (Deshpande et al. 2020). In addition, multiple observational systematics in cosmic shear analyses (see e.g. Euclid Collaboration et al. 2020) can be modelled with machine learning techniques and their effect on cosmological parameter estimation can thus be properly accounted for. Finally, for LSS a key advantage within our emulator is given by targeting the matter power spectrum, as opposed to the angular power spectra of the cosmological probes. This choice will allow us to investigate efficient computation of non-Limber projected quantities in future work.

- **Beyond-ΛCDM cosmologies**. As already mentioned above, we plan to extend CosmoPower to models beyond the flat ΛCDM one considered in this paper. For example, emulation of power spectra in non-flat cosmologies is of the utmost importance, since their calculation by Boltzmann codes are computationally intensive (see e.g. Handley 2021). More generally, power spectra computed in alternative cosmologies are considerably more demanding to compute for Boltzmann codes than in ΛCDM. Instead, we expect NN architectures similar to the ones considered in this paper to be equally accurate for emulating beyond-ΛCDM spectra (while possibly requiring larger training sets). Consequently, the evaluation time of these beyond-ΛCDM emulators will remain essentially the same as those reported in this paper. This will produce an even greater speed-up factor over Boltzmann codes. Some of these extensions to ΛCDM are already being developed and will be presented in a future publication (Spurio Mancini et al., in prep.).

- **A fully differentiable cosmology library**. In the longer term, CosmoPower will be extended to provide a completely differentiable library for cosmological computations. This will also include much simpler functions to emulate, such as cosmological distances. As we move towards the era of Stage IV surveys, the statistical challenges in analysing those datasets will certainly require increased sophistications in the Bayesian inference engines available. Differentiability is key in unlocking the possibility of efficient gradient-based inference, a promising avenue to tackle the challenge represented by the high-dimensional parameter spaces characterising the analyses of Stage IV surveys. Therefore, endowing the cosmological community with a fully differentiable forward model of multiple observables is a task of paramount importance, which we aim to accomplish with CosmoPower.

- **Interpretable machine learning**. Alternative methods for compression and emulation of cosmological quantities, such as autoencoders and symbolic regression (Udrescu et al. 2020), will be explored within CosmoPower, with the goal of maximising the computational efficiency and interpretability of the machine learning framework developed.

CosmoPower is a tool that allows for principled, non-invasive application of machine learning within a rigorous Bayesian framework for uncertainty quantification. We are confident that emulation techniques like those presented here will greatly enhance the scientific return from Bayesian inference analyses of upcoming Stage IV surveys.

DATA AVAILABILITY

CosmoPower is freely available at `https://github.com/alessiospuriomancini/cosmopower`.

ACKNOWLEDGEMENTS

We thank S. Joudaki for a thorough review of the manuscript and H. Peiris, A. Pourtsidou for useful discussions. We thank the anonymous referee for a meticulous and insightful review of this paper. ASM is supported by the MSSL STFC Consolidated Grant. DP is supported by the STFC UCL Centre for Doctoral Training in Data Intensive Science. JA is supported by the research project grant *Fundamental Physics from Cosmological Surveys* funded by the

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

Swedish Research Council (VR) under Dnr 2017-04212. Generation of the data used in this work has been performed in part on the Beaker and Hypatia clusters at UCL. We thank Edd Edmondson for technical support. This work has been partially enabled by funding from the UCL Cosmoparticle Initiative. We acknowledge the use of GetDist (Lewis 2019) to obtain corner plots.

Based on data products from observations made with ESO Telescopes at the La Silla Paranal Observatory under programme IDs 177.A-3016, 177.A-3017 and 177.A-3018. Based on observations obtained with Planck (http://www.esa.int/Planck), an ESA science mission with instruments and contributions directly funded by ESA Member States, NASA, and Canada.

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

REFERENCES

Abadi M., et al., 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, https://www.tensorflow.org/
Ade P., et al., 2019, Journal of Cosmology and Astroparticle Physics, 2019, 056âĂŞ056
Agarap A. F., 2018, CoRR, abs/1803.08375
Agarwal S., Abdalla F. B., Feldman H. A., Lahav O., Thomas S. A., 2012, Monthly Notices of the Royal Astronomical Society, 424, 1409âĂŞ1418
Agarwal S., Abdalla F. B., Feldman H. A., Lahav O., Thomas S. A., 2014, Monthly Notices of the Royal Astronomical Society, 439, 2102âĂŞ2121
Aghanim N., et al., 2020, Astronomy & Astrophysics, 641, A6
Albers J., Fidler C., Lesgourgues J., SchÃűneberg N., Torrado J., 2019, Journal of Cosmology and Astroparticle Physics, 2019, 028âĂŞ028
Alsing J., et al., 2020, The Astrophysical Journal Supplement Series, 249, 5
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, arXiv e-prints, p. arXiv:2104.14568
Asgari M., et al., 2021, Astronomy & Astrophysics, 645, A104
Auld T., Bridges M., Hobson M. P., Gull S. F., 2007, MNRAS, 376, L11
Auld T., Bridges M., Hobson M. P., 2008, Monthly Notices of the Royal Astronomical Society, 387, 1575
Bevins H. T. J., Handley W. J., Fialkov A., de Lera Acedo E., Javid K., 2021, MNRAS, 508, 2923
Bird S., Viel M., Haehnelt M. G., 2012, Monthly Notices of the Royal Astronomical Society, 420, 2551âĂŞ2561
Bird S., Rogers K. K., Peiris H. V., Verde L., Font-Ribera A., Pontzen A., 2019, Journal of Cosmology and Astroparticle Physics, 2019, 050
Bishop C. M., 2006, Pattern Recognition and Machine Learning. Springer
Blas D., Lesgourgues J., Tram T., 2011, J. Cosmology Astropart. Phys., 2011, 034
Brinckmann T., Lesgourgues J., 2019, Physics of the Dark Universe, 24, 100260
Buchner J., et al., 2014, A&A, 564, A125
Buffington T., Cabrera J.-M., Kurzawski A., Ezekoye O., 2020, Fire Technology, pp 1–27
Chisari N. E., et al., 2019, The Open Journal of Astrophysics, 2
Czekala I., Andrews S. M., Mandel K. S., Hogg D. W., Green G. M., 2015, ApJ, 812, 128
Das S., Chen X., Hobson M. P., Phadke S., van Beest B., Goudswaard J., Hohl D., 2018, Geophysical Journal International, 215, 1257
Deshpande A. C., et al., 2020, Astronomy & Astrophysics, 636, A95
Driver S. P., et al., 2009, Astronomy & Geophysics, 50, 5.12
Driver S. P., et al., 2011, Monthly Notices of the Royal Astronomical Society, 413, 971
Duncan C. A. J., Joachimi B., Heavens A. F., Heymans C., Hildebrandt H., 2013, Monthly Notices of the Royal Astronomical Society, 437, 2471âĂŞ2487
Euclid Collaboration et al., 2020, A&A, 635, A139
Euclid Collaboration et al., 2021, MNRAS, 505, 2840
Fendt W. A., Wandelt B. D., 2007, The Astrophysical Journal, 654, 2âĂŞ11
Feroz F., Hobson M. P., Bridges M., 2009, Monthly Notices of the Royal Astronomical Society, 398, 1601âĂŞ1614
Foreman-Mackey D., Hogg D. W., Lang D., Goodman J., 2013, Publications of the Astronomical Society of the Pacific, 125, 306âĂŞ312
Frostig R., Johnson M., Leary C., 2018, in SysML. https://mlsys.org/Conferences/doc/2018/146.pdf
Goodman J., Weare J., 2010, Communications in Applied Mathematics and Computational Science, 5, 65
Hajian A., 2007, Physical Review D, 75
Handley W., 2021, Physical Review D, 103
Handley W. J., Hobson M. P., Lasenby A. N., 2015a, Monthly Notices of the Royal Astronomical Society: Letters, 450, L61âĂŞL65
Handley W. J., Hobson M. P., Lasenby A. N., 2015b, Monthly Notices of the Royal Astronomical Society, 453, 4385âĂŞ4399
Hasan I., Schmidt S. J., Schneider M. D., Tyson J. A., 2021, arXiv e-prints, p. arXiv:2104.03407
Heavens A. F., Jimenez R., Lahav O., 2000, MNRAS, 317, 965
Heitmann K., Higdon D., White M., Habib S., Williams B. J., Lawrence E., Wagner C., 2009, The Astrophysical Journal, 705, 156âĂŞ174
Heitmann K., White M., Wagner C., Habib S., Higdon D., 2010, The Astrophysical Journal, 715, 104âĂŞ121
Heitmann K., Lawrence E., Kwan J., Habib S., Higdon D., 2013, The Astrophysical Journal, 780, 111
Heymans C., et al., 2021, Astronomy & Astrophysics, 646, A140
Hildebrandt H., et al., 2016, Monthly Notices of the Royal Astronomical Society, 465, 1454âĂŞ1498
Hildebrandt H., et al., 2020, Astronomy & Astrophysics, 633, A69
Ho M.-F., Bird S., Shelton C. R., 2021, MNRAS,
Ivezić Ž., et al., 2019, ApJ, 873, 111
Jimenez R., Verde L., Peiris H., Kosowsky A., 2004, Phys. Rev. D, 70, 023005
Joachimi B., Bridle S. L., 2010, Astronomy & Astrophysics, 523, A1
Joachimi B., et al., 2021, Astronomy & Astrophysics, 646, A129
Kaiser N., 1992, ApJ, 388, 272
Kasim M. F., et al., 2021, Machine Learning: Science and Technology, 3, 015013
Kern N. S., Liu A., Parsons A. R., Mesinger A., Greig B., 2017, The Astrophysical Journal, 848, 23
KÃűhlinger F., et al., 2017, Monthly Notices of the Royal Astronomical Society, 471, 4412âĂŞ4435
Kingma D. P., Ba J., 2014, arXiv e-prints, p. arXiv:1412.6980
Knabenhans M., et al., 2019, Monthly Notices of the Royal Astronomical Society, 484, 5509âĂŞ5529
Laureijs R., et al., 2011, arXiv e-prints, p. arXiv:1110.3193
Lawrence E., Heitmann K., White M., Higdon D., Wagner C., Habib S., Williams B., 2010, The Astrophysical Journal, 713, 1322âĂŞ1331
Lawrence E., et al., 2017, The Astrophysical Journal, 847, 50
Lesgourgues J., 2011, arXiv e-prints, p. arXiv:1104.2932
Lewis A., 2019, arXiv e-prints, p. arXiv:1910.13970
Lewis A., Challinor A., Lasenby A., 2000, ApJ, 538, 473
Liske J., et al., 2015, Monthly Notices of the Royal Astronomical Society, 452, 2087
LoVerde M., Afshordi N., 2008, Physical Review D, 78

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

OPTIONAL UNEDITED MANUSCRIPT

Manrique-Yus A., Sellentin E., 2019, Monthly Notices of the Royal Astronomical Society
Mead A. J., Peacock J. A., Heymans C., Joudaki S., Heavens A. F., 2015, Monthly Notices of the Royal Astronomical Society, 454, 1958
Mead A. J., Heymans C., Lombriser L., Peacock J. A., Steele O. I., Winther H. A., 2016, Monthly Notices of the Royal Astronomical Society, 459, 1468
Mead A. J., Brieden S., TrÃűster T., Heymans C., 2021, Monthly Notices of the Royal Astronomical Society, 502, 1401
Mootoovaloo A., Heavens A. F., Jaffe A. H., Leclercq F., 2020, Monthly Notices of the Royal Astronomical Society, 497, 2213
Mootoovaloo A., Jaffe A., Heavens A., Leclercq F., 2022, Astronomy and Computing, 38, 100508
Neal R. M., 1996, Bayesian Learning for Neural Networks. Springer-Verlag, Berlin, Heidelberg
Owen A. B., 1992, Statistica Sinica, 2, 439
Piras D., Spurio Mancini A., Joachimi B., Hobson M. P., 2021, arXiv e-prints, p. arXiv:2101.04724
Prince H., Dunkley J., 2019, Physical Review D, 100
Pyne S., Joachimi B., 2021, Monthly Notices of the Royal Astronomical Society, 503, 2300âĂŞ2317
Ramachandra N., Valogiannis G., Ishak M., Heitmann K., LSST Dark Energy Science Collaboration 2021, Phys. Rev. D, 103, 123525
Rasmussen C. E., Williams C. K. I., 2005, Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning). The MIT Press
Rogers K. K., Peiris H. V., Pontzen A., Bird S., Verde L., Font-Ribera A., 2019, Journal of Cosmology and Astroparticle Physics, 2019, 031
Schmit C. J., Pritchard J. R., 2017, Monthly Notices of the Royal Astronomical Society, 475, 1213âĂŞ1223
Schneider P., van Waerbeke L., Mellier Y., 2002, Astronomy & Astrophysics, 389, 729âĂŞ741
Smail I., Ellis R. S., Fitchett M. J., Edge A. C., 1995, Monthly Notices of the Royal Astronomical Society, 273, 277âĂŞ294
Smith R. E., et al., 2003, Monthly Notices of the Royal Astronomical Society, 341, 1311âĂŞ1332
Spergel D., et al., 2015, arXiv e-prints, p. arXiv:1503.03757
Spurio Mancini A., et al., 2019, Monthly Notices of the Royal Astronomical Society, 490, 2155
Spurio Mancini A., Piras D., Ferreira A. M. G., Hobson M. P., Joachimi B., 2021, Solid Earth, 12, 1683âĂŞ1705
Takahashi R., Sato M., Nishimichi T., Taruya A., Oguri M., 2012, The Astrophysical Journal, 761, 152
Thiagarajan J. J., Venkatesh B., Anirudh R., Bremer P.-T., Gaffney J., Anderson G., Spears B., 2020, Nature Communications, 11
Torrado J., Lewis A., 2019, Cobaya: Bayesian analysis in cosmology (ascl:1910.019)
Torrado J., Lewis A., 2021, Journal of Cosmology and Astroparticle Physics, 2021, 057
Tutusaus I., et al., 2020, Astronomy & Astrophysics, 643, A70
Udrescu S.-M., Tan A., Feng J., Neto O., Wu T., Tegmark M., 2020, arXiv e-prints, p. arXiv:2006.10782
Wan X., 2019, Journal of Physics: Conference Series, 1213, 032021
ZumalacÃąrregui M., Bellini E., Sawicki I., Lesgourgues J., Ferreira P. G., 2017, Journal of Cosmology and Astroparticle Physics, 2017, 019âĂŞ019
de Mijolla D., Viti S., Holdship J., Manolopoulou I., Yates J., 2019, Astronomy & Astrophysics, 630, A117
van Uitert E., et al., 2018, Monthly Notices of the Royal Astronomical Society, 476, 4662âĂŞ4689
von Wietersheim-Kramsta M., et al., 2021, Monthly Notices of the Royal Astronomical Society, 504, 1452âĂŞ1465

APPENDIX A: EMULATION DETAILS

A1 Neural network

A neural network (NN) consists of a sequence of layers connecting some input to some output, in this case cosmological parameters to power spectra (or their Principal Components). As depicted in Fig. 2, each layer performs a linear combination of the previous layer, and applies a nonlinear function to increase the expressiveness of the model. Following Alsing et al. (2020), we choose the following activation function for all hidden layers in our neural networks

$$f(\boldsymbol{x}) = \left(\boldsymbol{\gamma} + \left(1 + e^{-\boldsymbol{\beta}\odot\boldsymbol{x}}\right)^{-1} \odot (1-\boldsymbol{\gamma})\right) \odot \boldsymbol{x}\,, \tag{A1}$$

where $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are optimised together with the rest of the network parameters, and $\odot$ indicates the element-wise product. The activation function in Eq. A1 can be seen as a stack of scalar activation functions for each node j with independent hyperparameters $\beta_{\rm j}$ and $\gamma_{\rm j}$. We experimented with other, more traditional activation functions such as the hyperbolic tangent (tanh) or Rectified Linear Unit (ReLU) (Agarap 2018) and we always found them to perform slightly worse than our custom function in terms of accuracy.

In the CMB case the output layer is made of 2507 nodes, i.e. one for each multipole ℓ from $\ell_{\rm min} = 2$ to $\ell_{\rm max} = 2508$ included. When building the training and testing set, we make sure to ask the Boltzmann code for an explicit calculation of the CMB power spectrum at each one of the multipoles in the range, as this reduces the amount of interpolation performed internally by the Boltzmann code and, in turn, increases the accuracy of the CosmoPower emulation. For the matter power spectrum the output layer is made of 420 nodes, corresponding to the sampled k-modes: these are chosen such that the baryon acoustic oscillations features are more densely sampled, in an analogous fashion to what is performed internally by class and camb, albeit in this case with a cosmology-independent sampling.

Our neural networks always use 4 hidden layers of 512 neurons each, for both CMB and LSS; each neuron is associated to a weight and a bias, as described in Sect. 2 and Fig. 2. We did not perform a fully exhaustive optimization of hyperparameters of the neural networks such as the number of hidden layers or neurons. In our experiments we typically found that an architecture with at least three hidden layers of a few hundred neurons each was needed to achieve high accuracy results, with a four-layer configuration being even more performing. It is possible that equally, if not more accurate results may be achieved with a smaller architecture, albeit with more training samples. However, handling such large datasets may be a computationally non-trivial challenge, particularly in terms of memory requirements.

The network's parameters are optimised using Adam (Kingma & Ba 2014) with default parameters, and the loss function to minimise is chosen to be the mean squared error between the emulated and the true power spectra. We keep apart 20% of the training set for validation purposes. The learning rate is initially set to $1 \cdot 10^{-2}$, and then decreased by a factor of 10 each time the validation loss does not decrease for 20 epochs, where each epoch corresponds to feeding the whole training set into the network; the final learning rate is $1 \cdot 10^{-6}$. The batch size is changed accordingly, starting from $1 \cdot 10^3$, then $1 \cdot 10^4$, up to $5 \cdot 10^4$.

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

A2 Principal Component Analysis

We compared the performance of the direct NN mapping with an alternative emulation method where the spectra are first compressed to their Principal Components, following Speculator (Alsing et al. 2020), which we refer to for the full details. Note that this is necessary for the $C_\ell^{\rm TE}$ case, as due to the dynamic range of these spectra it is not possible to consider its logarithmic features; moreover, we found the performance of this second emulator superior over the direct NN mapping when emulating $C_\ell^{\phi\phi}$.

We keep 512 and 64 Principal Components for $C_\ell^{\rm TE}$ and $C_\ell^{\phi\phi}$, respectively; the neural network is then trained in the same way as described in Appendix A1. To obtain a power spectrum, we map the cosmological parameters to the predicted Principal Components, and then use the learnt change of base to map the Principal Components into the predicted power spectrum.

A3 Gaussian Process

We use a Gaussian Process (GP) to obtain derived cosmological parameter constraints from given samples of the posterior distribution of the other parameters. When obtaining a derived parameter $\theta_{\rm der}$, we assume that there exists a mapping $\theta_{\rm der} = f(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ indicates the parameters the emulator was trained on, and model $f(\boldsymbol{\theta})$ as a GP. Following Spurio Mancini et al. (2021), we assume that the function $f(\boldsymbol{\theta})$ follows a normal distribution with zero mean and a parametric covariance matrix, usually referred to as *kernel* $K(\boldsymbol{\theta}, \boldsymbol{\theta}'; \boldsymbol{\psi})$, with $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ indicating two points in parameter space, and $\boldsymbol{\psi}$ representing the trainable hyperparameters.

We choose the Automatic Relevance Discovery (ARD) 3/2 Matérn kernel (Neal 1996; Rasmussen & Williams 2005), which is expressed as

$$K_{\rm ARD-Matérn-3/2}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'; \boldsymbol{\psi}\right) = \alpha^2 \left(1 + \sqrt{3} r\right) \exp\left(-\sqrt{3} r\right) , \tag{A2}$$

where

$$r = \sqrt{\sum_{j=1}^{n} \frac{(\theta_j - \theta'_j)^2}{\sigma_j^2}} , \tag{A3}$$

n is the length of the vector $\boldsymbol{\theta}$ (in this case the number of cosmological parameters), and the hyperparameters $\boldsymbol{\psi} = \{\alpha, \boldsymbol{\sigma}\}$ are the signal standard deviation α and a characteristic scale σ_j for each input parameter $j = 1, \ldots, n$. We use the software GPy[10] to train the GP, i.e. to optimise the hyperparameters of the kernel, using the tuples $(\boldsymbol{\theta}, \theta_{\rm der})$ from the Boltzmann solver Camb as training data.

APPENDIX B: VALIDATION ON THE KIDS-1000 LIKELIHOOD

We include an application of CosmoPower to a recent cosmic shear band power analysis of the KiDS survey, covering ~1000 deg^2 (KiDS-1000, Asgari et al. 2021). For this analysis we need to train a matter power spectrum emulator for a model with one massive neutrino with mass $m_\nu = 0.06$ eV/c^2. We report in Fig. B2 the same accuracy plots shown in Fig. 4 for a massless neutrino model. Similar levels of accuracy of the linear power spectrum and nonlinear correction are achieved. In future extensions of CosmoPower we will integrate emulation over the neutrino mass as an additional parameter. Note also that in this analysis, following Asgari et al. (2021), we sample in S_8 rather than in $\ln 10^{10} A_{\rm s}$.

The uniform prior ranges assumed for this analysis are the same used in Asgari et al. (2021) and they are reported in Table B1. Following Asgari et al. (2021) we include one z-shift for each of the five redshift bins, D_{z_i}, for $i = 1, \ldots, 5$; for these shift parameters we set a correlated Gaussian prior. We refer to (Asgari et al. 2021; Joachimi et al. 2021) for further details, in particular on the redshift distributions and covariance matrix modelling. We run the inference pipeline[11] within the cosmological sampler MontePython. In Fig. B3 we show the excellent agreement between contours obtained with Class and CosmoPower, with the latter providing a speed-up factor of approximately 50. Values of the log-evidence are also in good agreement: $\log \mathcal{Z} = -81.45 \pm 0.14$ and $\log \mathcal{Z} = -81.40 \pm 0.14$ for the run with Class and CosmoPower, respectively.

APPENDIX C: RESULTS FOR CMB POWER SPECTRUM EMULATOR ON LARGER RANGE

In Fig. C2 we report the emulation accuracy of our CMB emulators applied to the test set sampled with cosmological parameters in the range of Table 1. The performance reaches no more than $0.2\sigma_{\rm CMB}$ as defined in Eqs. (11-13), which we verified corresponds to an adequate accuracy for Stage IV surveys, as we showed in the main body.

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

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

[10] http://github.com/SheffieldML/GPy
[11] adapted from https://github.com/BStoelzner/KiDS-1000_MontePython_likelihood

Parameter	Prior range
$\omega_{\rm b}$	[0.01875, 0.02625]
$\omega_{\rm cdm}$	[0.05, 0.255]
h	[0.64, 0.82]
n_s	[0.84, 1.1]
S_8	[0.1, 1.3]
$c_{\rm min}$	[2, 4]
$A_{\rm IA}$	[-6, 6]
D_{z_i}	$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{C})$

Table B1. Prior ranges for the KiDS-1000 cosmic shear band power analysis. Prior distributions are all taken to be uniform across these ranges, except for the z-shifts D_{z_i}, which are sampled from a correlated Gaussian prior with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{C}$ (see Asgari et al. 2021 for details on their values). Note that in this case we sample S_8 instead of $\ln 10^{10} A_{\rm s}$. As explained in the main text, our emulators are trained on different choices of this sampling parameter, and one can always recover the parameter that is not directly sampled in a post-processing step, using Gaussian Process regression.

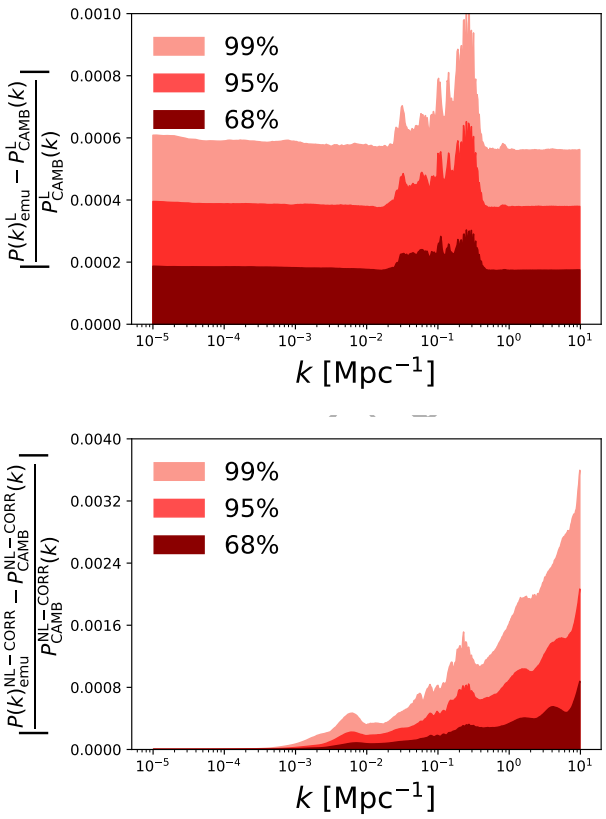

Figure B2. Same as Fig. 4 but for a cosmological model with one massive neutrino with mass $m_\nu = 0.06$ eV/c^2.

ORIGINAL MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

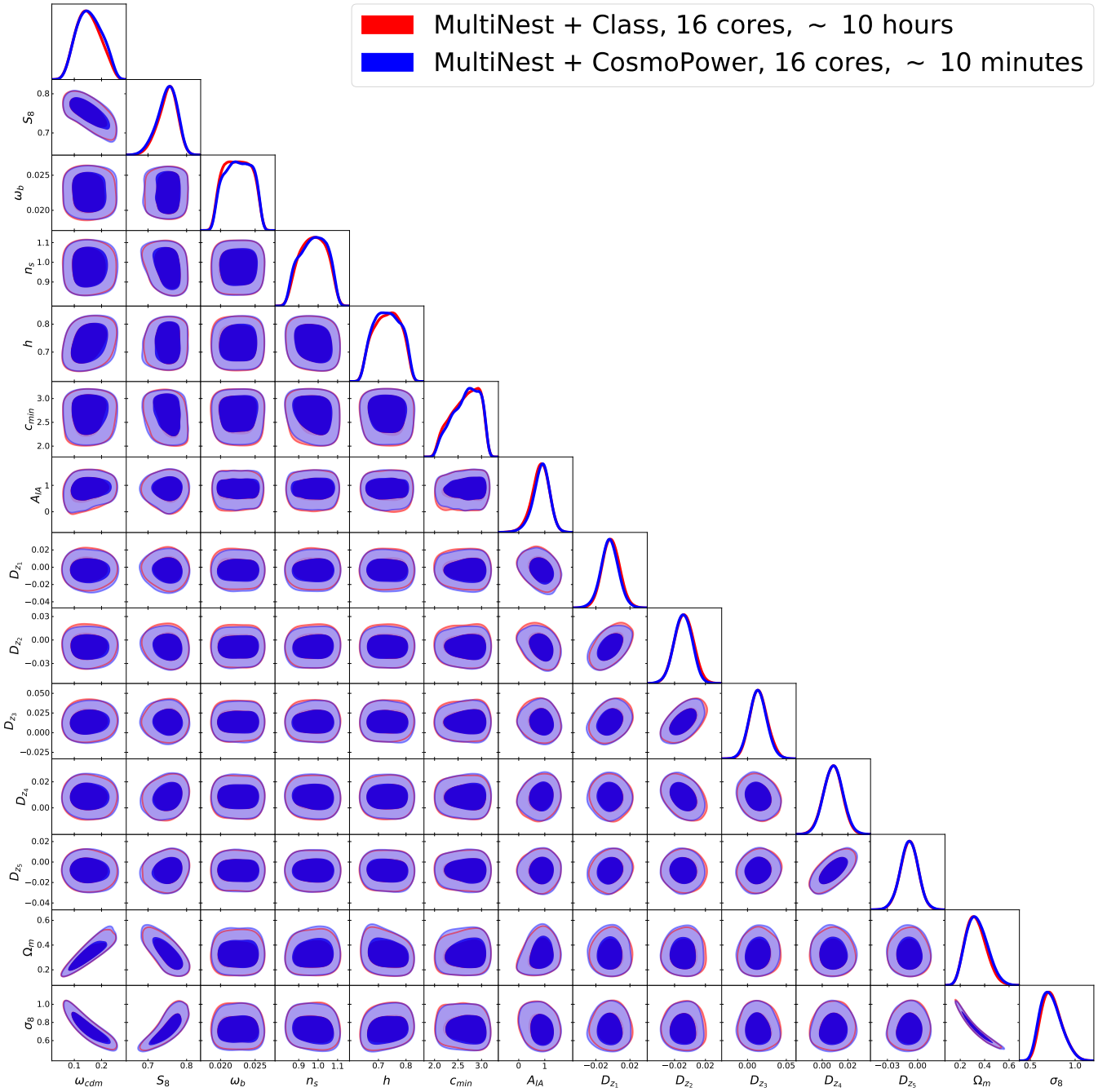

Figure B3. Contours for the KiDS-1000 cosmic shear band power analysis. In *red* contours obtained with Class, in *blue* those obtained with CosmoPower.

ORIGINAL

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022

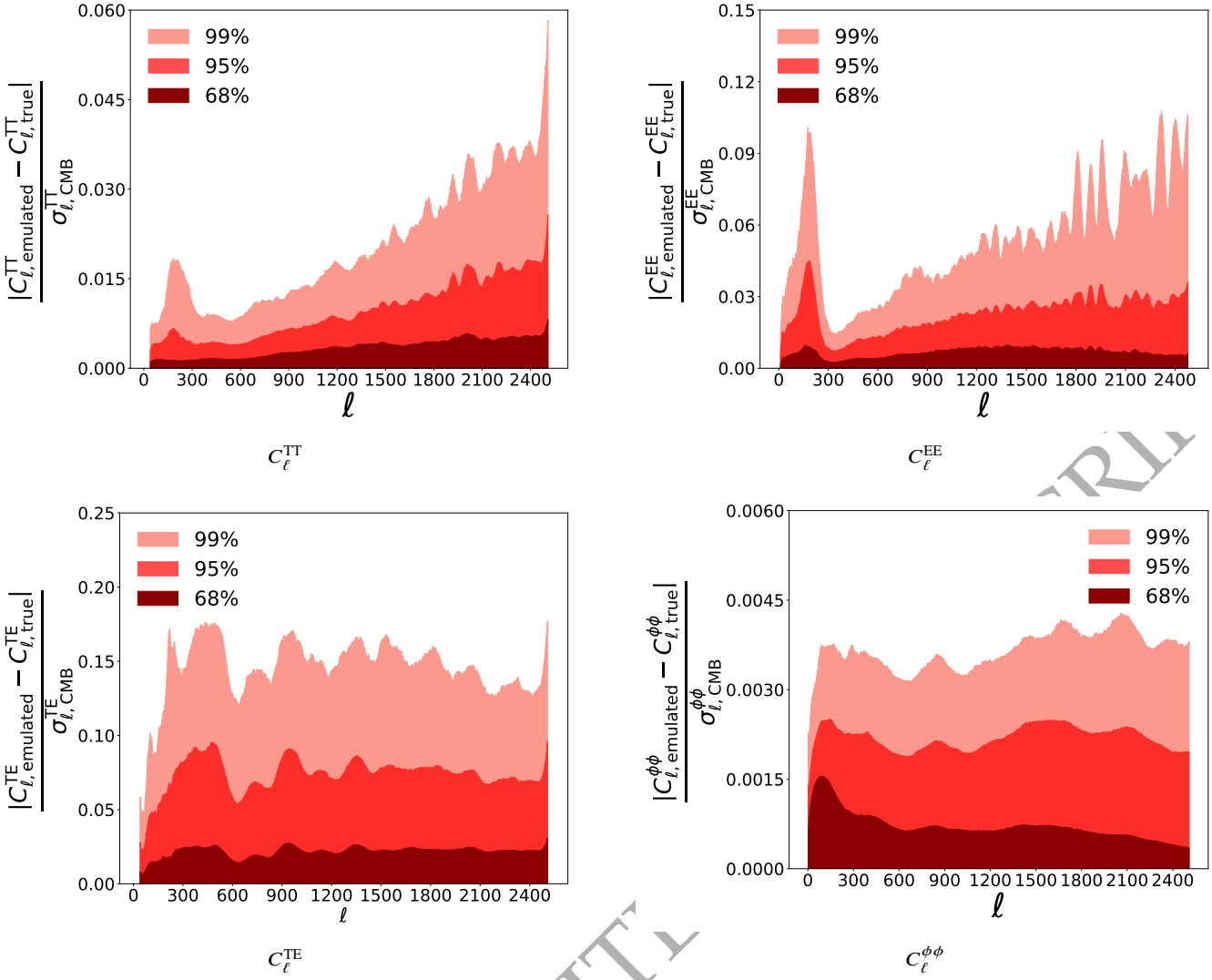

Figure C2. CMB power spectra emulation accuracy on the test set sampled from the range indicated in Table 1 for a) the temperature power spectrum, b) the polarisation power spectrum, c) the temperature-polarisation cross power spectrum, d) the lensing potential power spectrum. The emulation error is defined with respect to both instrumental and statistical noise, and is defined in Eqs. (11-13). *Dark red*, *red* and *salmon* areas enclose the 68, 95 and 99 percentiles of the test set. Details of the neural models are reported in Appendix A1.

ORIGINAL UNEDITED MANUSCRIPT

Downloaded from https://academic.oup.com/mnras/advance-article/doi/10.1093/mnras/stac064/6505144 by UCL, London user on 02 February 2022