A&A 669, L2 (2023)
https://doi.org/10.1051/0004-6361/202245156


Astronomy & Astrophysics

LETTER TO THE EDITOR

# Window function convolution with deep neural network models

D. Alkhanishvili[1]★, C. Porciani[1], and E. Sefusatti[2,3,4]

[1] Argelander Institut für Astronomie der Universität Bonn, Auf dem Hügel 71, 53121 Bonn, Germany
e-mail: daalkh@astro.uni-bonn.de
[2] Istituto Nazionale di Astrofisica, Osservatorio Astronomico di Trieste, Via Tiepolo 11, 34143 Trieste, Italy
[3] Istituto Nazionale di Fisica Nucleare, Sezione di Trieste, Via Valerio 2, 34127 Trieste, Italy
[4] Institute for Fundamental Physics of the Universe, Via Beirut 2, 34151 Trieste, Italy

Received 6 October 2022 / Accepted 6 December 2022

**ABSTRACT**

Traditional estimators of the galaxy power spectrum and bispectrum are sensitive to the survey geometry. They yield spectra that differ from the true underlying signal since they are convolved with the window function of the survey. For the current and future generations of experiments, this bias is statistically significant on large scales. It is thus imperative that the effect of the window function on the summary statistics of the galaxy distribution is accurately modelled. Moreover, this operation must be computationally efficient in order to allow sampling posterior probabilities while performing Bayesian estimation of the cosmological parameters. In order to satisfy these requirements, we built a deep neural network model that emulates the convolution with the window function, and we show that it provides fast and accurate predictions. We trained (tested) the network using a suite of 2000 (200) cosmological models within the cold dark matter scenario, and demonstrate that its performance is agnostic to the precise values of the cosmological parameters. In all cases, the deep neural network provides models for the power spectra and the bispectrum that are accurate to better than 0.1% on a timescale of 10 µs.

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

## 1. Introduction

The most common approach to extract cosmological information from a galaxy redshift survey involves measuring the power spectrum and/or the bispectrum of the galaxy distribution. In the majority of cases, the spectra are derived using traditional estimators (Yamamoto et al. 2006; Bianchi et al. 2015; Scoccimarro 2015) based on the ideas originally introduced by Feldman et al. (1994, hereafter FKP). One drawback of this method is that the survey geometry leaves an imprint on the measured spectra, which is difficult to model. The observed galaxy overdensity field $\delta_{\rm obs}(\boldsymbol{x})$ does not coincide with the actual fluctuations $\delta(\boldsymbol{x})$. The reasons are twofold. First, galaxy surveys cover only finite sections of our past light cone. Second, tracers of the large-scale structure need to be weighted based on the selection criteria of the survey. In compact form we write $\delta_{\rm obs}(\boldsymbol{x}) = W(\boldsymbol{x})\,\delta(\boldsymbol{x})$, where $W(\boldsymbol{x})$ denotes the window function of the survey. It follows that the observed power spectrum and the underlying true power $P(k)$ satisfy the relation (Peacock & Nicholson 1991; FKP)

$$P_{\rm obs}(\boldsymbol{k}) = \int |\tilde{W}_2(\boldsymbol{k}-\boldsymbol{q})|^2\, P(\boldsymbol{q})\, \frac{{\rm d}^3 q}{(2\pi)^3}, \tag{1}$$

where $\tilde{W}_n(\boldsymbol{k})$ denotes the Fourier transform of the function $W(\boldsymbol{x})$ normalised such that

$$\tilde{W}_n(\boldsymbol{k}) = \frac{\int W(\boldsymbol{x})\, {\rm e}^{i\,\boldsymbol{k}\cdot\boldsymbol{x}}\, {\rm d}^3 x}{\left\{\int [W(\boldsymbol{x})]^n\, {\rm d}^3 x\right\}^{1/n}}. \tag{2}$$

Similarly, for the bispectrum we obtain

$$B_{\rm obs}(\boldsymbol{k}_1,\boldsymbol{k}_2,\boldsymbol{k}_3) = \int \tilde{W}_3(\boldsymbol{k}_1-\boldsymbol{q})\, \tilde{W}_3(\boldsymbol{k}_2-\boldsymbol{p})\, \tilde{W}_3(\boldsymbol{k}_3+\boldsymbol{q}+\boldsymbol{p})\, B(\boldsymbol{q},\boldsymbol{p},-\boldsymbol{q}-\boldsymbol{p})\, \frac{{\rm d}^3 q}{(2\pi)^3}\, \frac{{\rm d}^3 p}{(2\pi)^3}. \tag{3}$$

The convolution in Eq. (1) mixes Fourier modes with different wavevectors and modifies the power[1] significantly on large scales. Since the survey window is generally not spherically symmetric, it also creates an anisotropic signal in addition to redshift-space distortions and the Alcock-Paczynski effect. These consequences need to be accounted for in order to fit theoretical models to the observational data, in particular when trying to constrain the level of primordial non-Gaussianity (e.g., Castorina et al. 2019) or general relativistic effects (e.g., Elkhashab et al. 2022). Two approaches are possible: by trying to remove the effect from the data (e.g., Sato et al. 2011) or accounting for the window in the models (e.g., de Laix & Starkman 1998; Percival et al. 2001; Ross et al. 2013). This second line of attack is much more popular: starting from an estimate for the window function, a model for

★ Member of the International Max Planck Research School (IMPRS) for Astronomy and Astrophysics at the Universities of Bonn and Cologne.

[1] A different class of estimators based on pixelised maps of the galaxy density is immune to this problem and directly provides noisy measurements of the unwindowed spectra (see e.g., the so-called quadratic estimators for the power spectrum in Tegmark et al. 1998 and Philcox 2021b and the cubic estimator for the bispectrum in Philcox 2021a). However, these estimators are sub-optimal on small scales. Moreover, at the moment, there is no such estimator for the anisotropic bispectrum in redshift space.

$P_{\rm obs}(\boldsymbol{k})$ is obtained by solving the convolution integral numerically. Developing numerical procedures for fast likelihood evaluation is pivotal in multivariate Bayesian inference. For this reason, Blake et al. (2013) reformulated the convolution integrals as matrix multiplications and made use of pre-computed 'mixing matrices' to evaluate the impact of the survey window on the power spectra averaged within wavector bins. A computationally efficient way to evaluate the effect of the window function on the multipole moments of the power spectrum in the distant-observer approximation is presented by Wilson et al. (2017) and generalised by Beutler et al. (2017) to the local plane-parallel case (in which the line of sight varies with the galaxy pair). In this approach the convolution is cast in terms of a sequence of one-dimensional Hankel transforms that are performed using the FFTlog algorithm (Hamilton 2000). The key idea is to compress the information about the window function into a finite number of multipole moments of its autocorrelation function (see also Beutler et al. 2014). Further extensions account for wide-angle effects (e.g., Castorina & White 2018; Beutler et al. 2019).

Evaluating the impact of the survey window on the bispectrum has only recently received attention in the literature. From a computational perspective, performing the six-dimensional convolution integral in Eq. (3) is a challenging task that cannot form the basis of a toolbox for Bayesian inference. It is thus necessary to develop faster techniques. Inspired by perturbation theory at leading order, Gil-Marín et al. (2015) proposed an approximation where the monopole moment of the convolved bispectrum is given by the linear superposition of products of two convolved power spectra given by Eq. (1). Although this approximation is accurate enough for the BOSS survey (barring squeezed triangular configurations, which are excluded from the analysis by Gil-Marín et al. 2015), it would likely introduce severe biases in the analysis of the next generation of wide and deep surveys such as *Euclid* (Laureijs et al. 2011) or DESI (DESI Collaboration 2016), which will provide measurements with much smaller statistical uncertainties (see e.g., Yankelevich & Porciani 2019). Sugiyama et al. (2019) introduced a new bispectrum estimator based on the tri-polar spherical harmonic decomposition with zero total angular momentum and showed, in this case, that it is possible to compute the models for the convolved bispectrum following a FFT-based approach. The issue of developing a similar method for more traditional estimators of the bispectrum multipoles (Scoccimarro 2015) has been recently addressed by Pardede et al. (2022), who derived an expression based on two-dimensional Hankel transforms that can be computed using the 2D-FFTlog method (Fang et al. 2020). In this case the survey window is described in terms of the multipoles of its three-point correlation function. Developing optimal estimators for these quantities is still an open problem.

In this Letter we propose employing deep learning as a method to compute the impact of the survey window function on theoretical models for the power spectrum and bispectrum. Specifically, we use a deep neural network (DNN) to approximate the mapping from the unconvolved to the convolved spectra. This technique allows us to consider multiple cosmological models, while drastically reducing computer-memory demands and the wall-clock time of computation with respect to performing the convolution integrals numerically. All these features are key for building efficient Bayesian inference samplers and determining the posterior distribution of cosmological parameters. The structure of the Letter is as follows. In Sect. 2 we briefly describe the architecture of our DNN models and introduce the data sets we employ for training and testing them. Our results are presented in Sect. 3. We draw our conclusions in Sect. 4.

## 2. Methods

### 2.1. Philosophy and goals

It is well known that artificial neural networks are able to approximate any arbitrary continuous function of real variables (Cybenko 1989; White 1990; Hornik 1991). They learn how to map some inputs (features, in machine learning jargon) to outputs (labels) from examples in a training data set. The training process consists of fitting the parameters of the machine (weights and biases of the neurons) by minimising a loss function that quantifies how good the prediction is with respect to the correct result. After the training the accuracy of the model is determined using the testing data.

In our applications the features that form the input of the DNN are the spectra $P(\boldsymbol{k})$ and $B(\boldsymbol{k}_1, \boldsymbol{k}_2, \boldsymbol{k}_3)$ evaluated at specific sets of wavevectors. Different options are available when choosing these sets. For instance, we could use many closely separated wavevectors around the output configurations. In this case the DNN would learn how the convolution integrals mix the contributions coming from different configurations. At the opposite extreme, we could consider inputs and outputs evaluated for the very same set of configurations, so that, in some sense, the DNN model also interpolates among the sparser inputs. We opted for this second approach, which is more conducive to a simpler machine learning set-up: choosing a smaller size of features requires fewer model parameters to be tuned, and the trained model is evaluated more quickly. The only implicit assumption here is that the input power spectrum is smooth between the sampled configurations. In our implementation the machine learns to predict the functions

$$R_{\rm P}(\boldsymbol{k}) = \frac{P_{\rm obs}(\boldsymbol{k})}{P(\boldsymbol{k})} \quad \text{and} \quad R_{\rm B}(\boldsymbol{k}_1, \boldsymbol{k}_2, \boldsymbol{k}_3) = \frac{B_{\rm obs}(\boldsymbol{k}_1, \boldsymbol{k}_2, \boldsymbol{k}_3)}{B(\boldsymbol{k}_1, \boldsymbol{k}_2, \boldsymbol{k}_3)} \tag{4}$$

evaluated at the same arguments of the input.

Since this Letter is about giving a proof of concept, for simplicity we only predict the effect of window function on the linear matter power spectrum and the so-called tree-level bispectrum, which can be trivially computed using the linear power spectrum (e.g., Fry 1984), neglecting redshift-space distortions in both cases. Moreover, as an example, we consider a spherically symmetric top-hat window function which assumes the value of one for distances smaller than the radius $R$ and zero otherwise. In Fourier space this corresponds to

$$\tilde{W}_n(k) = \frac{4\pi\,(kR)^2\, j_1(kR)}{k^3\, V^{1/n}}, \tag{5}$$

where the symbol $j_\nu(x)$ denotes the spherical Bessel functions of the first kind and $V = 4\pi R^3/3$ is the comoving volume enclosed by the window function. Basically, $\tilde{W}_n(k)$ rapidly oscillates (see Fig. 1) which makes it challenging to numerically compute the integrals in Eqs. (1) and (3). The oscillations are damped, and the main contribution to the convolution comes from the first peak at $k = 0$, which mixes Fourier modes within a shell of width $\Delta k \simeq R^{-1}$. Given our assumptions, $R_{\rm P}(k)$ and $R_{\rm B}(k_1, k_2, k_3)$ only depend on the modulus of the wavevectors.

### 2.2. Deep learning models

Since the power spectrum is a smooth function of $k$, we adopt the convolutional neural network (CNN) architecture to model

$R_{\rm P}(k)$. The first layer of the network applies a convolution to the input with 16 trainable filters (kernel size 3) and a rectified linear unit (ReLU) activation function, defined as $f(x) = \max(0, x)$. This is followed by a dropout layer with a rate of 0.5, which acts as a regulariser and prevents overfitting (Goodfellow et al. 2016). The final layer is a dense one in which the number of neurons matches the length of the output data vector. Since the convolved power spectrum must be positive, the last layer is processed through a softplus activation function of the form $f(x) = \ln(1 + e^x)$.

For the bispectrum, we opt for a different CNN architecture. We chose this because we organise the data in a one-dimensional array where each entry corresponds to a different triangular configuration. As a consequence, the sequence of data is not necessarily smooth and we prefer to use a model that can detect features on multiple scales. The network we chose is based on the U-Net architecture (Ronneberger et al. 2015), which consists of a contracting path (encoder) followed by an expansive path (decoder). The former combines convolutional and pooling layers to down-sample the original data, and thus builds a compressed representation of them. The latter decompresses the compact representation to construct an output of the desired size. We include two down-sampling and two up-sampling steps, followed by a dropout layer with a rate of 0.5 and a dense layer in which the number of neurons matches the length of the output data vector.

For building and training the neural networks, we use the Keras library (Chollet 2015) under the TensorFlow framework (Abadi et al. 2015). During the training phase, the parameters of the machine are adjusted to minimise a loss function $L$, which we identify with the mean absolute error (MAE) $L = N^{-1} \sum_{i=1}^{N} |y_i^{\rm pred}/y_i^{\rm truth} - 1|$, where $y_i^{\rm pred}$ denotes the DNN prediction, $y_i^{\rm truth}$ is the corresponding item in the training data set, and $N$ is the number of entries in these data vectors. The minimisation of this loss function (one of the most popular choices for regression problems) is controlled by the Adam Optimizer (based on the stochastic gradient descent method, Kingma & Ba 2014) with an initial learning rate of 0.001, which is reduced as the training progresses using the inverse time decay schedule of TensorFlow. In order to prevent overfitting, five per cent of the training data are set aside as a validation set (i.e., these data are not used to fit the network parameters), and training is automatically stopped early if the validation loss starts to increase. This step makes sure that the network predicts previously unseen data more accurately.

### 2.3. Training and testing data sets

We used the suite of 2000 linear power spectra in the Quijote database (Villaescusa-Navarro et al. 2020) to build the training set for our DNNs. These spectra are obtained by sampling five cosmological parameters on a Latin hypercube over the ranges defined in Table 1.

For the power-spectrum analysis, we use 47 linearly spaced wavenumbers in the range $[0.004, 0.2]\,h\,{\rm Mpc}^{-1}$. The corresponding values for $P_{\rm obs}(k)$ are computed using the 3D FFT method to evaluate Eq. (1) employing a top-hat window, which covers a comoving volume of $V = 700^3\,h^{-3}\,{\rm Mpc}^3$ (i.e., $R = 434.25\,h^{-1}$ Mpc).

For the bispectrum we compute the tree-level expression from the linear power spectra in the Quijote suite. We consider 564 triangular configurations in which the three sides span the range $[0.01, 0.2]\,h\,{\rm Mpc}^{-1}$. In this case the integration in Eq. (3) is carried out in six dimensions using the VEGAS routine of the Cuba library (Hahn 2005). To facilitate the convergence of the numerical integrals, we consider a top-hat window with $V = 200^3\,h^{-3}\,{\rm Mpc}^3$ ($R = 124.07\,h^{-1}$ Mpc).

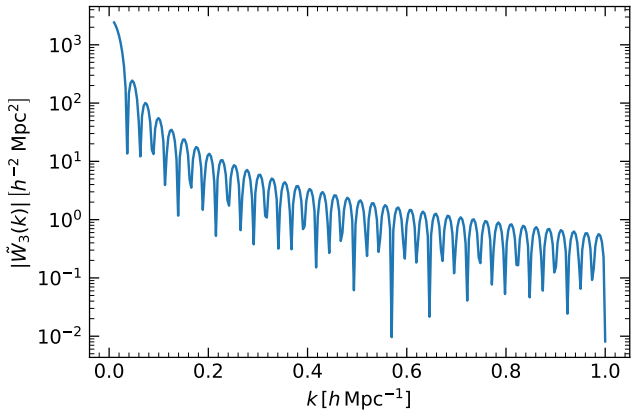


**Fig. 1.** Top-hat window function with $V = 200^3\,h^{-3}\,{\rm Mpc}^3$.

We generate the test data set by randomly generating 200 sets of cosmological parameters with Latin hypercube sampling. Since the Quijote database spans a broader region of parameter space compared with that allowed by current observational constraints, our test data are sampled within a narrower region mimicking the actual constraints from Planck Collaboration VI (2020); see the bottom row of Table 1. For each set we compute the linear matter power spectrum using CAMB (Lewis et al. 2000; Howlett et al. 2012) and employ the same procedure used for the training data set to obtain the convolved power spectra and bispectra.

## 3. Results

In the top panels of Fig. 2 we consider one of the test samples for the power spectrum. The orange triangles in the left panel show the function $R_{\rm P}(k)$ computed using Eq. (1): the convolution with the window function flattens out the power spectrum on large scales and changes the amplitude of the baryonic acoustic oscillations by a few per cent. Although the window function considered here is arbitrary, similarly sized (and measurable) corrections are expected for the next generation of galaxy redshift surveys (see e.g., Fig. 6 in Elkhashab et al. 2022), which should deliver per cent accuracy for the power spectrum. The black dots indicate the output of the trained DNN model. The right panel shows the relative error between the DNN prediction and the true signal, which is always smaller than 0.1%. To assess the overall performance of the DNN model, in the bottom panels of Fig. 2 we plot the MAEs for each test sample (left) and error percentiles over the test samples as a function of the wavenumber. The residual mean inaccuracy of the model is well below the per cent level.

The effect of the window function on the bispectrum is much more pronounced than for the power spectrum and the ratio $R_{\rm B}(k_1, k_2, k_3)$ assumes values below 0.5 for some triangle configurations (top left panel in Fig. 3). The DNN model predicts the corrections accurately in all cases (top right and bottom panels) and vastly outperforms the approximated method introduced by Gil-Marín et al. (2015), which, for the compact survey volume considered here, does not accurately reproduce the amplitude of the convolved bispectrum (green circles in the top left panel).

**Table 1.** Parameter spaces spanned by the training and testing data sets.

| Data set | $\Omega_m$ | $\Omega_b$ | $h$ | $n_s$ | $\sigma_8$ |
|---|---|---|---|---|---|
| Training | [0.1,0.5] | [0.03,0.07] | [0.5,0.9] | [0.8,1.2] | [0.6,1.0] |
| Testing | [0.2,0.4] | [0.03,0.06] | [0.6,0.8] | [0.9,1.1] | [0.7,1.0] |

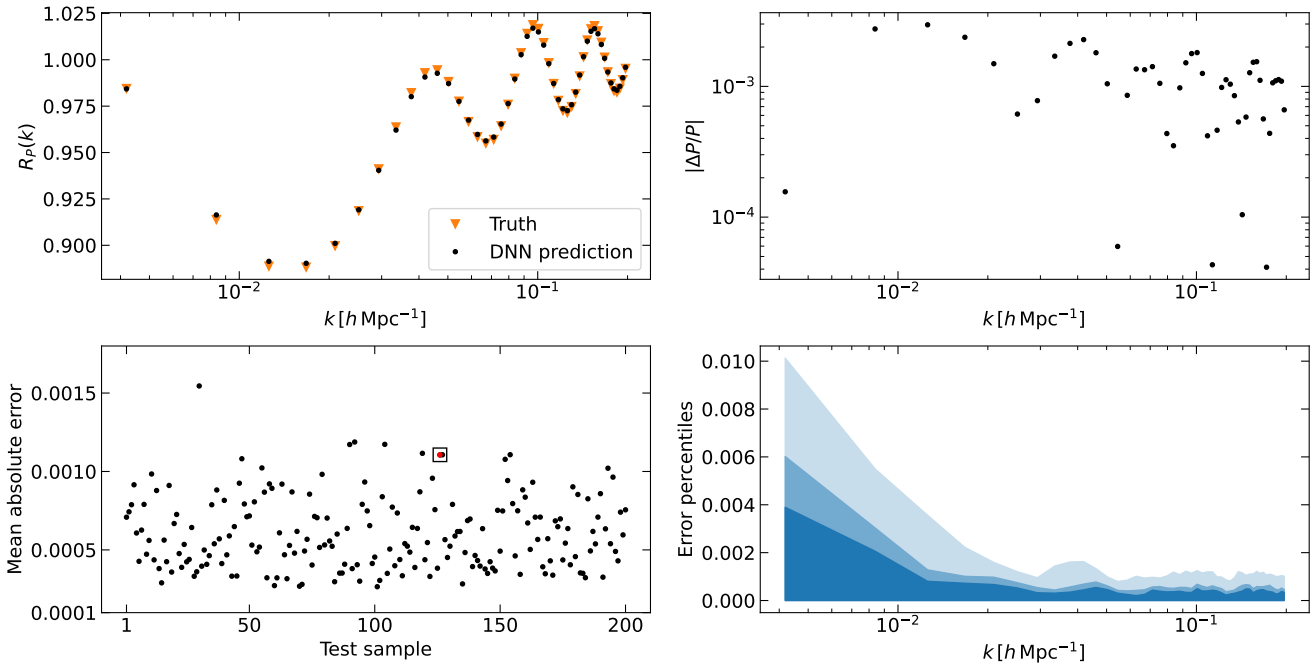


**Fig. 2.** Accuracy of the trained DNN for the power spectrum convolution. Top left: function $R_P(k)$ obtained with the convolution integral in Eq. (1) (orange triangles) compared with the DNN model (black dots) for one test sample. Top right: relative error of the DNN model in the same test sample used in the left panel. Bottom left: MAEs for all the test samples (the one used in the top panels is highlighted in red and surrounded by a square). Bottom right: 50th, 68th, and 95th error percentiles of the DNN model as a function of $k$. Generally, the DNN model yields sub-per cent accuracy.

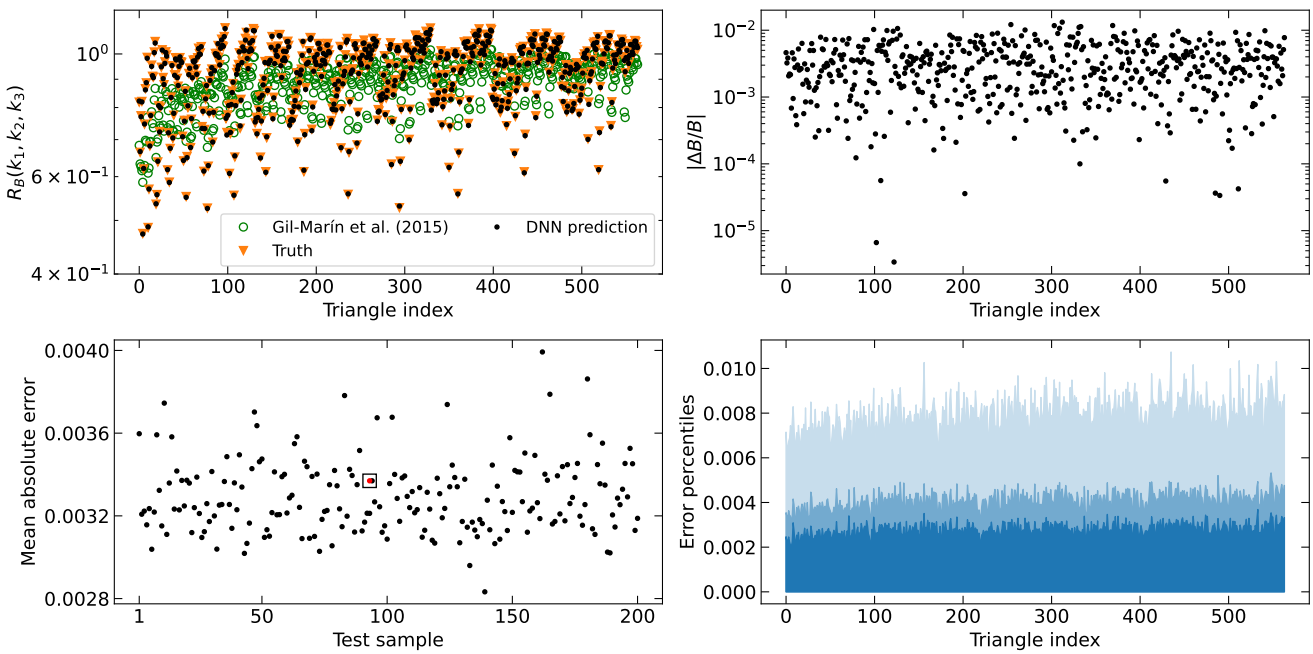


**Fig. 3.** As in Fig. 3, but for the bispectrum. The triangular configurations in the top and bottom right panels satisfy the constraint $k_1 \geq k_2 \geq k_3$ and are ordered so that first $k_3$ increases (at fixed $k_1$ and $k_2$), then $k_2$ (at fixed $k_1$), and finally $k_1$. For reference, in the top left panel, the ratio $R_B(k_1, k_2, k_3)$ is also plotted, computed according to the approximated method introduced by Gil-Marín et al. (2015; green circles).

## 4. Conclusions

In this Letter we employed a DNN model to predict the impact of the window function on the power spectrum and the bispectrum measured in a galaxy redshift survey. Overall, the trained DNN models show very promising results with sub-per cent MAEs for all test samples (well below the statistical uncertainty expected from the next generation of surveys). These errors can be further reduced by increasing the size of the training data set.

Our DNN model is meant as a proof of concept and, for this reason, we made some simplifications in our study. First, we used the linear power spectrum and the tree-level bispectrum for matter fluctuations. Second, we considered a top-hat window function with a fixed volume in which the number density of tracers does not vary with the radial distance from the observer. Although this is an ideal case, we do not see a reason why a DNN model should not be able to accurately predict the effect of more realistic survey masks, given an appropriately sized training sample.

It takes less than 10 microseconds to generate a complete sample for either $R_{\rm P}(k)$ or $R_{\rm B}(k_1, k_2, k_3)$ with the trained DNN. This is ideal for sampling posterior probabilities in Bayesian parameter estimation. Our method can be straightforwardly generalised to the multipoles of the spectra, and could also be combined with emulators that make predictions for the true clustering signal (including galaxy biasing) based on perturbation theory (e.g., Donald-McCann et al. 2023; DeRose et al. 2022; Eggemeier et al. 2022). Additional corrections due to binning the theory predictions in exactly the same way as done for the measurements (see e.g., Sect. 3.2 in Oddo et al. 2020 and Sect. 4.1 in Alkhanishvili et al. 2022) can be computed by suitably averaging the output of the DNN model or, more efficiently, can be accounted for in the model. Since in this Letter we do not perform a Bayesian inference for cosmological parameters, we skipped this step when we generated the training sample.

The bottleneck operation in the DNN approach is the creation of the training data set, which requires a significant time investment in the case of the bispectrum (in our case, the calculation of the 2000 convolved bispectra with 64 processor cores took approximately one month of wall-clock time). This step can be sped up using massive parallelisation and, possibly, by relying on more computationally friendly formulations of the convolution integral (e.g., Pardede et al. 2022). It is also conceivable that using larger input vectors that densely sample the wavenumbers within shells of size $\Delta k$ around the output configurations might facilitate the task of the machine, and would thus help to reduce the training data. The time required to build the training set is not a good reason to dismiss the DNN approach. Even for the simple case of the isotropic bispectrum of matter-density fluctuations in real space, sampling the posterior distribution of the five cosmological parameters we considered would require many more than 2000 evaluations of the window-convolved signal. Thus, using the DNN model would lead in any case to a notable speed up. In any practical application, accounting for redshift-space distortions, shot noise, and perturbative counterterms would substantially increase the number of adjustable coefficients in the perturbative model for the bispectrum (and, correspondingly, the number of likelihood evaluations needed to constrain them from experimental data). We thus conclude that using the DNN model would be advantageous as long as the size of the necessary training set is substantially smaller than the number of the required likelihood evaluations in the Bayesian estimation of the model parameters.

*Acknowledgements.* We thank Alexander Eggemeier and Héctor Gil-Marin for useful discussions. D.A. acknowledges partial financial support by the Shota Rustaveli National Science Foundation of Georgia (GNSF) under the grant FR-19-498.

## References

Abadi, M., Agarwal, A., Barham, P., et al. 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, software available from tensorflow.org
Alkhanishvili, D., Porciani, C., Sefusatti, E., et al. 2022, MNRAS, 512, 4961
Beutler, F., Saito, S., Seo, H.-J., et al. 2014, MNRAS, 443, 1065
Beutler, F., Seo, H.-J., Saito, S., et al. 2017, MNRAS, 466, 2242
Beutler, F., Castorina, E., & Zhang, P. 2019, J. Cosmol. Astropart. Phys., 2019, 040
Bianchi, D., Gil-Marín, H., Ruggeri, R., & Percival, W. J. 2015, MNRAS, 453, L11
Blake, C., Baldry, I. K., Bland-Hawthorn, J., et al. 2013, MNRAS, 436, 3089
Castorina, E., & White, M. 2018, MNRAS, 476, 4403
Castorina, E., Hand, N., Seljak, U., et al. 2019, J. Cosmol. Astropart. Phys., 2019, 010
Chollet, F. 2015, Keras, https://keras.io
Cybenko, G. 1989, Math. Control Signals Syst., 2, 303
de Laix, A. A., & Starkman, G. 1998, ApJ, 501, 427
DeRose, J., Chen, S.-F., White, M., & Kokron, N. 2022, J. Cosmol. Astropart. Phys., 2022, 056
DESI Collaboration (Aghamousa, A., et al.) 2016, ArXiv e-prints [arXiv:1611.00036]
Donald-McCann, J., Koyama, K., & Beutler, F. 2023, MNRAS, 518, 3106
Eggemeier, A., Camacho-Quevedo, B., Pezzotta, A., et al. 2022, MNRAS, accepted [arXiv:2208.01070]
Elkhashab, M. Y., Porciani, C., & Bertacca, D. 2022, MNRAS, 509, 1626
Fang, X., Eifler, T., & Krause, E. 2020, MNRAS, 497, 2699
Feldman, H. A., Kaiser, N., & Peacock, J. A. 1994, ApJ, 426, 23
Fry, J. N. 1984, ApJ, 279, 499
Gil-Marín, H., Noreña, J., Verde, L., et al. 2015, MNRAS, 451, 539
Goodfellow, I., Bengio, Y., & Courville, A. 2016, Deep Learning (MIT Press)
Hahn, T. 2005, Comput. Phys. Commun., 168, 78
Hamilton, A. J. S. 2000, MNRAS, 312, 257
Hornik, K. 1991, Neural Netw., 4, 251
Howlett, C., Lewis, A., Hall, A., & Challinor, A. 2012, J. Cosmol. Astropart. Phys., 1204, 027
Kingma, D. P., & Ba, J. 2014, ArXiv e-prints [arXiv:1412.6980]
Laureijs, R., Amiaux, J., Arduini, S., et al. 2011, ArXiv e-prints [arXiv:1110.3193]
Lewis, A., Challinor, A., & Lasenby, A. 2000, ApJ, 538, 473
Oddo, A., Sefusatti, E., Porciani, C., Monaco, P., & Sánchez, A. G. 2020, J. Cosmol. Astropart. Phys., 2020, 056
Pardede, K., Rizzo, F., Biagetti, M., et al. 2022, J. Cosmol. Astropart. Phys., 2022, 066
Peacock, J. A., & Nicholson, D. 1991, MNRAS, 253, 307
Percival, W. J., Baugh, C. M., Bland-Hawthorn, J., et al. 2001, MNRAS, 327, 1297
Philcox, O. H. E. 2021a, Phys. Rev. D, 104, 123529
Philcox, O. H. E. 2021b, Phys. Rev. D, 103, 103504
Planck Collaboration VI. 2020, A&A, 641, A6
Ronneberger, O., Fischer, P., & Brox, T. 2015, ArXiv e-prints [arXiv:1505.04597]
Ross, A. J., Percival, W. J., Carnero, A., et al. 2013, MNRAS, 428, 1116
Sato, T., Hütsi, G., & Yamamoto, K. 2011, Progr. Theoret. Phys., 125, 187
Scoccimarro, R. 2015, Phys. Rev. D, 92, 083532
Sugiyama, N. S., Saito, S., Beutler, F., & Seo, H.-J. 2019, MNRAS, 484, 364
Tegmark, M., Hamilton, A. J. S., Strauss, M. A., Vogeley, M. S., & Szalay, A. S. 1998, ApJ, 499, 555
Villaescusa-Navarro, F., Hahn, C., Massara, E., et al. 2020, ApJS, 250, 2
White, H. 1990, Neural Netw., 3, 535
Wilson, M. J., Peacock, J. A., Taylor, A. N., & de la Torre, S. 2017, MNRAS, 464, 3121
Yamamoto, K., Nakamichi, M., Kamino, A., Bassett, B. A., & Nishioka, H. 2006, PASJ, 58, 93
Yankelevich, V., & Porciani, C. 2019, MNRAS, 483, 2078