

## Learning cosmology and clustering with cosmic graphs

PABLO VILLANUEVA-DOMINGO  <sup>1</sup> AND FRANCISCO VILLAESCUSA-NAVARRO  <sup>2,3</sup>

<sup>1</sup> Computer Vision Center - Universitat Autònoma de Barcelona (UAB), 08193 Bellaterra, Barcelona, Spain

<sup>2</sup> Center for Computational Astrophysics, Flatiron Institute, 162 5th Avenue, New York, NY, 10010, USA

<sup>3</sup> Department of Astrophysical Sciences, Princeton University, Peyton Hall, Princeton NJ 08544, USA

### ABSTRACT

We train deep learning models on thousands of galaxy catalogues from the state-of-the-art hydrodynamic simulations of the CAMELS project to perform regression and inference. We employ Graph Neural Networks (GNNs), architectures designed to work with irregular and sparse data, like the distribution of galaxies in the Universe. We first show that GNNs can learn to compute the power spectrum of galaxy catalogues with a few percent accuracy. We then train GNNs to perform likelihood-free inference at the galaxy-field level. Our models are able to infer the value of  $\Omega_m$  with a  $\sim 12\% - 13\%$  accuracy just from the positions of  $\sim 1000$  galaxies in a volume of  $(25 h^{-1}\text{Mpc})^3$  at  $z = 0$  while accounting for astrophysical uncertainties as modelled in CAMELS. Incorporating information from galaxy properties, such as stellar mass, stellar metallicity, and stellar radius, increases the accuracy to  $4\% - 8\%$ . Our models are built to be translational and rotational invariant, and they can extract information from any scale larger than the minimum distance between two galaxies. However, our models are not completely robust: testing on simulations run with a different subgrid physics than the ones used for training does not yield as accurate results.

### 1. INTRODUCTION

The standard model of cosmology is the current most accepted theoretical framework that accurately describes a very large set of cosmological observations. This model contains a set of free parameters characterizing fundamental properties on the Universe, such as its geometry, expansion rate, and composition. Constraining the value of these parameters is a major priority in cosmology; we expect to improve our knowledge on fundamental physics by measuring these parameters with the highest accuracy possible.

The spatial distribution of galaxies in the Universe contains a wealth of information about the value of the cosmological parameters. That information is typically extracted using summary statistics like the power spectrum, that are known to be suboptimal descriptors of the statistical properties of non-Gaussian density fields, like the galaxy density field on non-linear scales. Recently, a large effort has been carried out to compare how much cosmological information can be extracted from

different summary statistics of the web, such as correlation functions and higher momenta (Villaescusa-Navarro et al. 2020a; Samushia et al. 2021; Gualdi et al. 2020, 2021; Hahn et al. 2020; Hahn & Villaescusa-Navarro 2021a), counts in cells (Banerjee & Abel 2021a,b; Uhlemann et al. 2020), Minkowski functionals (Vicinanza et al. 2019; Liu et al. 2022; Marques et al. 2019), peak counts (Li et al. 2019; Ajani et al. 2020; Harnois-Déraps et al. 2021, 2022), wavelets (Valogiannis & Dvorkin 2021; Eickenberg et al. 2022; Cheng & Ménard 2021), via the kinetic Sunyaev Zel'dovich effect (Kuruvilla & Aghanim 2021; Kuruvilla 2021; Giri & Smith 2020) or through other approaches (Banerjee et al. 2020; Bayer et al. 2021; Ally et al. 2020; Naidoo et al. 2021; de la Bella et al. 2020; Massara et al. 2021; Liu & Madhavacheril 2019; Ryu & Lee 2020; Lee & Ryu 2020; Zhang et al. 2020; Simpson et al. 2011, 2013; Friedrich et al. 2020; Dai et al. 2020).

Alternatively, machine learning (ML) methods can be used to optimally extract information from the field itself without using summary statistics. This task has been carried out, typically using Convolutional Neural Networks (CNNs), for density fields (Ravanbakhsh et al. 2017; Villaescusa-Navarro et al. 2021a,b; Lazanu 2021; Hortua 2021; Makinen et al. 2021), weak lensing maps

pablo.villanueva.domingo@gmail.com

fvillaescusa@flatironinstitute.org

(Schmelzle et al. 2017; Gupta et al. 2018; Ribli et al. 2019; Fluri et al. 2019; Zorrilla Matilla et al. 2020; Jeffrey et al. 2020; Lu et al. 2022), cosmic microwave background maps (He et al. 2018), and 21 cm maps (Gillet et al. 2019; Hassan et al. 2020; Villanueva-Domingo & Villaescusa-Navarro 2021) among others.

In this work we use machine learning to extract the maximum information from galaxy catalogues without using summary statistics. For this task we employ Graph Neural Networks (GNNs), that are capable of dealing with irregular and sparse data (Bronstein et al. 2021; Battaglia et al. 2018; Hamilton 2020), like the spatial distribution of galaxies. The models we use here employ the so-called message passing scheme, where information from a defined neighborhood is shared with a particular galaxy, exploiting global and local relations. While these models are already by construction permutation invariant, we build them to respect the physical symmetries associated to this task: rotational and translational invariance. Another advantage of GNNs is that they do not impose any cut in scale. With CNNs, the grid size determines the minimum scale<sup>1</sup> where information can be extracted, while there is not such scale cut-off in GNNs, where galaxies can be arbitrarily close and the model can extract information from features on such scales. Due to their multiple advantages, GNNs are increasingly being applied in cosmology and astrophysics (Cranmer et al. 2019; Cranmer et al. 2020; Beck & Sadowski 2019; Cranmer et al. 2021; Villanueva-Domingo et al. 2022, 2021).

Previous works have already tried to infer cosmological parameters from 3D galaxy distributions using machine learning. For instance, CNNs have been applied to galaxy catalogues created combining Halo Occupation Distribution (HOD) models with N-body simulations (Ntampaka et al. 2020). However, we believe that our approach represents a step forward for several reasons. First, we use a deep learning architecture based on GNNs that we believe is more appropriate to deal with the sparse and irregular data associated to galaxies. Second, our galaxy catalogues come from state-of-the-art hydrodynamic simulations of the CAMELS project (Villaescusa-Navarro et al. 2020b) that model galaxy positions and properties more accurately than other methods. Third, we use GNNs to extract information not only from galaxy clustering, but also from galaxy properties such as stellar mass, stellar metallicity, and stellar radius. Fourth, we construct our model

to be invariant under the desired symmetries: translations, rotations, and permutations. Fifth, we perform likelihood-free inference to obtain a confident estimate of the expected uncertainty of our models. Sixth, we perform robustness tests to establish whether the model can potentially be used with real data or whether it is too dependent on the type of data used for training.

This paper is organized as follows. In Sec. 2 we describe the data we use in this work. The architecture of our model is described in Sec. 3. We first show in Sec. 4 that graph neural networks can learn estimators of galaxy clustering like the power spectrum. We then show the results of inferring cosmological parameters from galaxy distributions in Sec. 5. Finally, we conclude and discuss the main results of this work in 6.

## 2. DATA

In this work we have used galaxy catalogues from simulations of the CAMELS project (Villaescusa-Navarro et al. 2020b; Villaescusa-Navarro et al. 2022a). Each galaxy catalogue is obtained from a snapshot of a state-of-the-art hydrodynamic simulation that follows the evolution of  $256^3$  dark matter particles and  $256^3$  fluid elements from  $z = 127$  down to  $z = 0$ . All simulations share the value of these cosmological parameters,  $\Omega_b = 0.049$ ,  $\Omega_K = 0$ ,  $h = 0.6711$ ,  $n_s = 0.9624$ ,  $w = -1$ ,  $\sum_i m_{\nu_i} = 0$  eV, and they can be classified into two different suites:

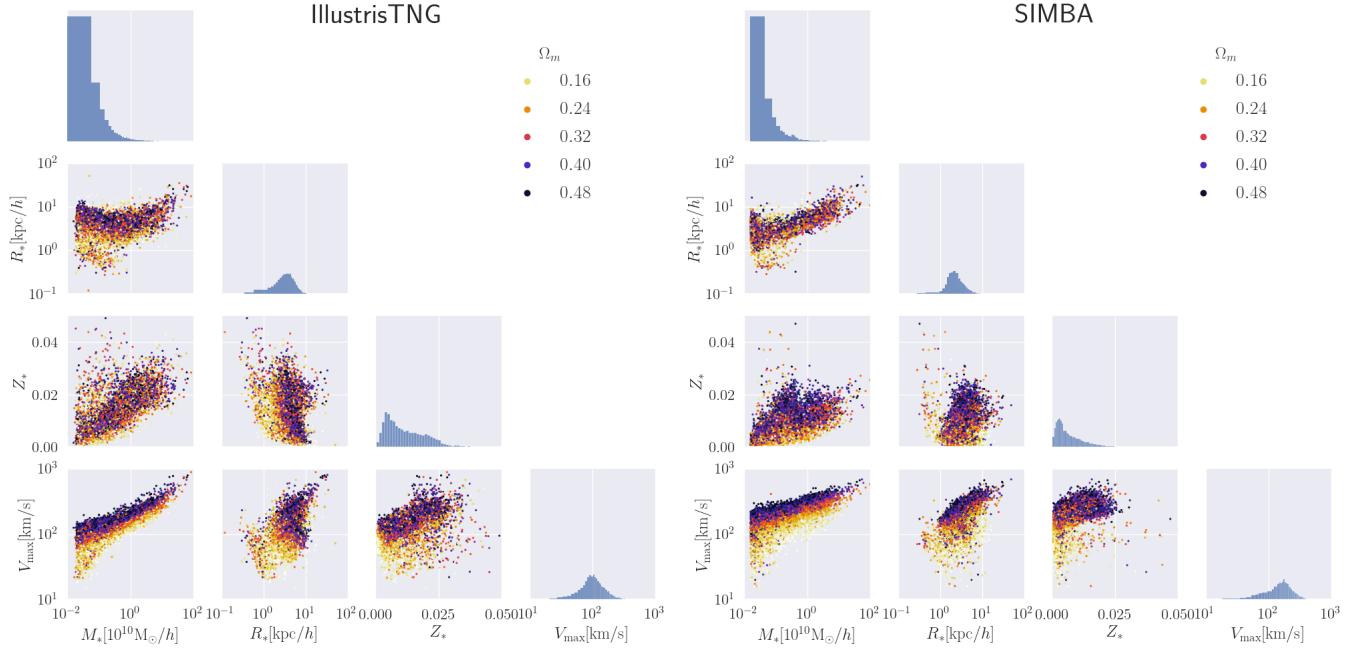
- **IllustrisTNG.** These simulations have been run with the AREPO<sup>2</sup> code (Springel 2010; Weinberger et al. 2020) and employ the IllustrisTNG subgrid physics model (Weinberger et al. 2017; Pillepich et al. 2018; Nelson et al. 2019).
- **SIMBA.** These simulations have been run with the GIZMO<sup>3</sup> code (Hopkins 2015) and employ the SIMBA subgrid physics model (Davé et al. 2019).

We employ the so-called LH set of both suites, that contains 1,000 simulations with different values of the cosmological and astrophysical parameters, and initial random seed. The cosmological parameters varied in the simulations are the matter density parameter,  $\Omega_m$ , and the variance of the linear field on scales of  $8 h^{-1}\text{Mpc}$  at  $z = 0$ ,  $\sigma_8$ . The astrophysical parameters are denoted by  $A_{\text{SN}1}$ ,  $A_{\text{SN}2}$ ,  $A_{\text{AGN}1}$ , and  $A_{\text{AGN}2}$ , and control the efficiency of supernovae ( $A_{\text{SN}1}$  and  $A_{\text{SN}2}$ ) and active galactic nuclei ( $A_{\text{AGN}1}$  and  $A_{\text{AGN}2}$ ) feedback. The values of these six parameters are arranged in a latin-hypercube

<sup>1</sup> While it is possible to take a finer grid, making the cutoff scale smaller, it comes at the cost of using more memory and increasing the sparsity of the data.

<sup>2</sup> <https://arepo-code.org/>

<sup>3</sup> <http://www.tapir.caltech.edu/~phopkins/Site/GIZMO.html>



**Figure 1.** This figure shows 1D and 2D distributions of galaxy properties in the IllustrisTNG (left) and SIMBA (right) simulations. Each point represents a galaxy, and it is color-coded according to the value of  $\Omega_m$ . The considered galaxy properties are: stellar mass,  $M_*$ , stellar half-mass radius,  $R_*$ , stellar metallicity,  $Z_*$ , and maximum circular velocity,  $V_{\max}$ . As can be seen, in the CAMELS simulations, galaxy properties occupy different regions in the feature space depending on the value of  $\Omega_m$ . Hence, galaxy properties may be used to infer the value of the cosmological parameters.

with boundaries set by

$$\Omega_m \in [0.1; 0.5] \quad (1)$$

$$\sigma_8 \in [0.6; 1.0] \quad (2)$$

$$A_{\text{SN}1}, A_{\text{AGN}1} \in [0.25; 4.0] \quad (3)$$

$$A_{\text{SN}2}, A_{\text{AGN}2} \in [0.5; 2.0] . \quad (4)$$

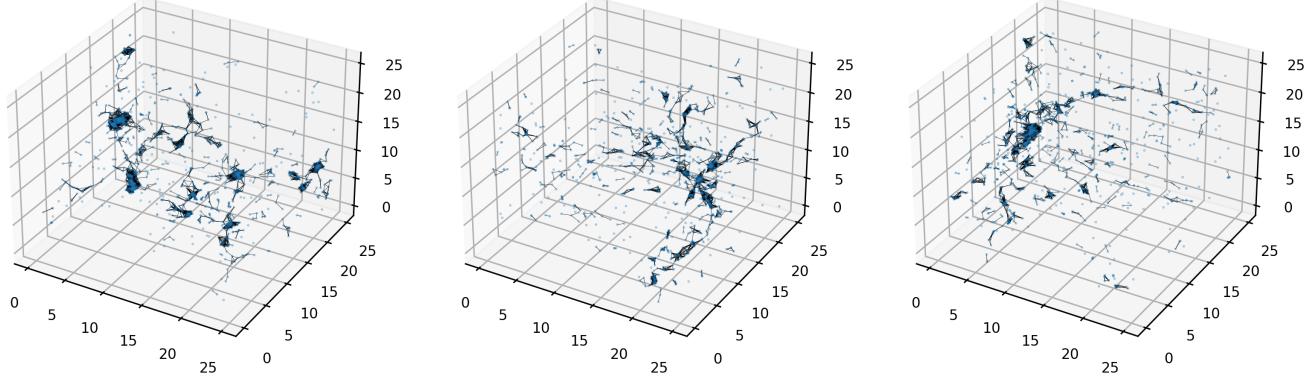
The simulations in the two suites differ not only on the value of the cosmological and astrophysical parameters, but also in the way the hydrodynamic equations are solved and in the subgrid model recipes. We emphasize that having two independent models to simulate galaxies, cosmic gas, and in general the large-scale structure of the Universe is key in order to asset the *robustness* of the developed machine learning model.

In total we have 2,000 galaxy catalogues at a given redshift: 1,000 from the IllustrisTNG simulations and 1,000 from the SIMBA simulations. At  $z = 0$ , the galaxy catalogues from the IllustrisTNG simulations contain an average of 700 galaxies, while this number raises to 1200 for the SIMBA simulations. For each snapshot of each simulation we have identified halos and subhalos by running SUBFIND (Springel et al. 2001). In this paper we work with galaxies, that we define as subhalos that contain at least 20 star particles. Thus, our galaxy catalogues contain galaxies with stellar masses above  $\sim 10^8 h^{-1} M_\odot$ . We have checked that using a slightly different thresh-

old does not change our conclusions. We use galaxy catalogues in real-space, and leave for future work the analysis for galaxy catalogues in redshift-space. While for regression we will work with galaxy catalogues at different redshifts, for inference we only consider galaxy catalogues at  $z = 0$ . For each simulation we have thus a galaxy catalogue that contains the following information for each galaxy:

- 3D position,  $\mathbf{p}$ . The Cartesian coordinates of the galaxy in real-space.
- Stellar mass,  $M_*$ . Defined as the total mass in stars in the galaxy.
- Stellar radius,  $R_*$ . Defined as the radius that contains half of the galaxy's stellar mass.
- Stellar metallicity,  $Z_*$ . Defined as the mass-weighted metallicity of the stars in the galaxy.
- Maximum circular velocity,  $V_{\max}$ . Defined as the maximum value of  $\sqrt{GM(r)/r}$ . This quantity is typically dominated by the dark matter content of a given subhalo.

We emphasize that all considered galaxy properties, with the exception of  $V_{\max}$ , are observationally accessible. The idea behind the usage of these features is



**Figure 2.** The input to our model is a graph, that we build from a given galaxy catalogue. In the graph, the nodes represent galaxies and two nodes are linked by an edge if their distance is smaller than the linking radius  $r$ . This figure show three graphs from three different CAMELS-IllustrisTNG simulations where  $r = 1.25 h^{-1}\text{Mpc}$ . Each box represents the galaxy catalogue of a simulation at  $z = 0$ . Box units are in  $h^{-1}\text{Mpc}$ .

that galaxy properties, individually or collectively, may 1) carry out cosmological information and/or 2) contain astrophysical information that can be used to break cosmological-astrophysical degeneracies.

To illustrate the potential information that galaxy properties may encode, we show in Fig. 1 correlations among different galaxy features, color coded by  $\Omega_m$ , for galaxies in the  $z = 0$  catalogues. As can be seen, and already noted in Villaescusa-Navarro et al. (2022b), there are some noticeable correlations between galaxy properties and  $\Omega_m$ . Thus, our model may learn to extract cosmological information based on galaxy properties, on top of galaxy clustering. It is important to note that the trends between the different galaxy features and  $\Omega_m$  can significantly vary from the IllustrisTNG suite (left) to the SIMBA suite (right). This fact illustrates the differences in the subgrid models employed in the IllustrisTNG and SIMBA simulations. We will investigate in Sec. 5.3 the robustness of our results to changes in the subgrid model.

When training GNNs with galaxy catalogues, we will consider all models equally probable. For instance galaxy catalogues from simulations with very efficient supernova feedback are considering equally probable than catalogues from simulations with more realistic feedback efficiencies. This is equivalent to consider a flat prior on the value of both the cosmological and astrophysical parameters. Given the broad range in the value of the astrophysical parameters, and the fact that we are interested in inferring the value of the cosmological parameters, it is appropriate to say that our networks are trained to learn to marginalize over baryonic effects, as implemented in CAMELS.

### 3. MODEL

In this section we first describe how we transform the galaxy catalogues into graphs, the input to our model. We then describe the architecture of our GNNs and explain how the loss function is set depending on the task to be carried out: regression or likelihood-free inference.

#### 3.1. From galaxy catalogues to cosmic graphs

We represent each galaxy catalogue as a mathematical graph  $\mathcal{G}$  that is built as follows. Galaxies represent the graph nodes, and two galaxies are connected by an edge if their separation is smaller than the linking radius  $r$ , an hyperparameter. We show some examples of graphs constructed from galaxy catalogues of IllustrisTNG simulations in Fig. 2.

We note that the method to construct the graphs from the galaxy catalogues is not unique. For instance, the  $k$  nearest neighbors of a given node can be used to define the graph's edges. However, we have checked that this method led to worse results than the linking radius approach.

We now describe in detail the different elements of the graphs and what features are used in order to respect the underlying symmetries. As we shall see below, the architecture of our model uses several graph blocks, that will take a given graph and output another one with some of its attributed updated. Therefore, we will describe the graph properties at a given graph block  $l$ .

##### 3.1.1. Node features

The features of the node  $i$  at block  $l$  are represented by  $\mathbf{h}_i^{(l)}$ . The initial node features,  $\mathbf{h}_i^{(0)}$ , are set by the intrinsic properties of the node's host galaxy introduced in Sec. 2: 1) the stellar mass  $M_*$ , 2) the stellar half-

mass radius  $R_*$ , 3) the stellar metallicity  $Z_*$ , and 4) the maximum subhalo velocity  $V_{\max}$ . We note that a subset of these properties can also be used as node features, e.g. only the stellar mass. In some cases we employ catalogues that only contain galaxy positions. In such cases, we do not use initial node features, although hidden node features are computed, as discussed in Sec. 3.2.

### 3.1.2. Edge features

We denote the edge features between nodes  $i$  and  $j$  at the block  $l$  as  $\mathbf{e}_{ij}^{(l)}$ . The edges play a crucial role in our model since they account for the spatial information. Due to statistical homogeneity and isotropy, we would like our model to be invariant under translations and rotations, i.e., rigid transformations of the Euclidean group. Let  $\mathbf{p}_i$  be the position of the galaxy in the node  $i$ . Given a rotation matrix  $\mathbf{R}$  and a translation  $\mathbf{T}$ , positions transform as  $\mathbf{p}_i \rightarrow \mathbf{R}\mathbf{p}_i + \mathbf{T}$ . Requiring that the output of our network is invariant under these transformations impose restrictions over the spatial information to be employed.

Translational symmetry can be imposed demanding that the edge features should be given in terms of relative positions rather than the absolute ones. Using  $\mathbf{d}_{ij} = \mathbf{p}_i - \mathbf{p}_j$ , instead of  $\mathbf{p}_i$  or  $\mathbf{p}_j$ , will already fulfill this requirement.

Symmetry under rotations can be built by using as features the distance and the scalar products of the directions of two vectors defining the edge (Villar et al. 2021). Lets define the unit vectors  $\mathbf{s}_{ij} = \mathbf{d}_{ij}/|\mathbf{d}_{ij}|$  and  $\mathbf{n}_i = (\mathbf{p}_i - \bar{\mathbf{p}})/|\mathbf{p}_i - \bar{\mathbf{p}}|$ , where  $\bar{\mathbf{p}}$  is the centroid of the distribution<sup>4</sup>. Then, we can employ as edge features the scalar products  $\alpha_{ij} = \mathbf{n}_i \cdot \mathbf{n}_j$  and  $\beta_{ij} = \mathbf{n}_i \cdot \mathbf{s}_{ij}$ , together with the relative distance,  $|\mathbf{d}_{ij}|$ . Note that these three scalar features are equivalent to the vector  $\mathbf{p}_i - \mathbf{p}_j$ , and therefore our model does not loose information by using these features instead of the relative distance vector.<sup>5</sup> Since it is better to input the model features that are dimensionless and normalized, we divide  $|\mathbf{d}_{ij}|$  by the linking radius  $r$ . Thus, the initial edge features between nodes  $i$  and  $j$ ,  $\mathbf{e}_{ij}^{(0)}$ , are given by

$$\mathbf{e}_{ij}^{(0)} = [|\mathbf{d}_{ij}|/r, \alpha_{ij}, \beta_{ij}], \quad (5)$$

<sup>4</sup> The center of mass or the position of any galaxy could also be employed, as long as  $\bar{\mathbf{p}}$  transforms as  $\mathbf{p}_i$ .

<sup>5</sup> Note that there is a third possible product,  $\mathbf{n}_j \cdot \mathbf{s}_{ij}$ , but this scalar should not provide additional independent information since the vectors  $\mathbf{p}_i$ ,  $\mathbf{p}_j$  and  $\mathbf{d}_{ij}$  form a triangle whose angles must sum  $\pi$  radians. Thus, only two angles are independent, and we evaluate them using  $\alpha_{ij}$  and  $\beta_{ij}$ . Other way to reason is that  $\mathbf{d}_{ij}$  carries three degrees of freedom, the same than the quantities  $|\mathbf{d}_{ij}|$ ,  $\alpha_{ij}$  and  $\beta_{ij}$ .

where [...] denotes concatenation (in the *features* axis). Note that by construction,  $\mathbf{e}_{ij}^{(0)}$  is translation and rotation invariant, i.e., rigid transformations of the form  $\mathbf{p}_i \rightarrow \mathbf{R}\mathbf{p}_i + \mathbf{T}$  leave  $\mathbf{e}_{ij}^{(0)}$  invariant. Since all spatial information of the data is encoded in the edge features, and these are constructed to be translational and rotational invariant, the entire network will satisfy those symmetries by construction.

We have explicitly checked that using the vector  $\mathbf{d}_{ij}$  and the distance as edge features leads to similar performance, although in this case data augmentation (e.g. random rotations) is needed to teach the model the rotational symmetry.

We consider self-loops when galactic features are included, but we disregard them when only using galaxy positions. For self-loops the edge attributes are set to  $\alpha_{i,i} = 1$  and  $\beta_{i,i} = 0$  for consistency.

### 3.1.3. Global features

Besides node and edge attributes, a graph can have global features,  $\mathbf{u}$ , that characterize properties of the entire graph. These properties can be used in conjunction with the edge and/or node features to update the different components of the graph. In our case, we make use of the logarithm of the total number of galaxies in the simulation as a global feature.

We have checked that in the case of catalogues that contain galaxy properties, using additional global properties, like the total stellar mass of the catalogue, does not improve the accuracy of the model. Thus, we do not consider additional global properties.

## 3.2. Architecture

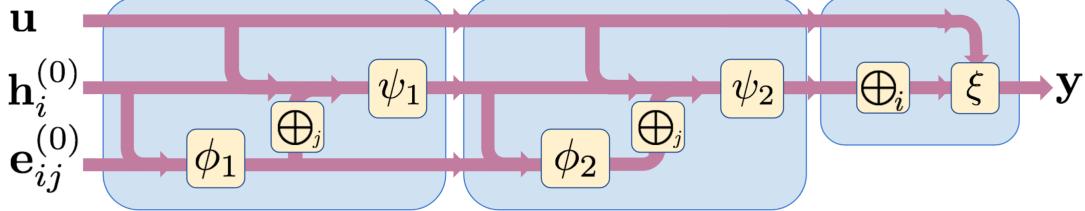
We follow the Graph Network framework (Battaglia et al. 2018) to build the architecture of our model. Our GNNs are composed by different graph blocks, also called message passing layers, which update node and edge features, followed by a final aggregation layer which outputs global quantities from the full graph. In the following we shall discuss in detail how the different graph elements are updated by the graph blocks and how the final aggregation layer works. An scheme with building blocks of our GNN is shown in Fig. 3. We refer the reader to Bronstein et al. (2021); Battaglia et al. (2018); Hamilton (2020) for general references on GNNs.

The graph block  $l$  is composed by:

- An edge model that takes as input the edge and node features and outputs new edge features:

$$\mathbf{e}_{ij}^{(l+1)} = \phi_{l+1}([\mathbf{h}_i^{(l)}, \mathbf{h}_j^{(l)}, \mathbf{e}_{ij}^{(l)})], \quad (6)$$

- A node model that takes as input the node and global features together with the updated edge fea-



**Figure 3.** Example of the GNN architecture with two graph blocks. Node, edge and global features are input into the network and updated through the different message passing layers. A final pooling step aggregates the information from all nodes to output the target  $\mathbf{y}$ .

tures and returns new node features:

$$\mathbf{h}_i^{(l+1)} = \psi_{l+1}([\mathbf{h}_i^{(l)}, \bigoplus_{j \in \mathcal{N}_i} \mathbf{e}_{ij}^{(l+1)}, \mathbf{u}]) \quad (7)$$

In the above expressions,  $\phi_{l+1}$  and  $\psi_{l+1}$  represent generic functions modeled through multilayer perceptrons (MLPs),  $\mathcal{N}_i$  denotes the neighborhood of node  $i$ , i.e., the collection of all other nodes with an edge connected to  $i$ , and  $\bigoplus$  is a permutation invariant aggregation operator, which aggregates the information from all the neighbors of node  $i$ . Examples of such operators are the sum, the mean, and the maximum. In our case, we do not know a-priori which operator will perform better. Thus, we use the three of them and concatenate the results into an array:

$$\bigoplus_{j \in \mathcal{N}_i} \mathbf{e}_{ij}^{(l+1)} = \left[ \max_{j \in \mathcal{N}_i} \mathbf{e}_{ij}^{(l+1)}, \sum_{j \in \mathcal{N}_i} \mathbf{e}_{ij}^{(l+1)}, \frac{\sum_{j \in \mathcal{N}_i} \mathbf{e}_{ij}^{(l+1)}}{\sum_{j \in \mathcal{N}_i}} \right]. \quad (8)$$

We have checked that using this *multi-pooling* operation yields slightly more accurate results than using a single pooling operation, consistently with previous works which show the advantage of combining multiple aggregators (Corso et al. 2020).

In some situations, we disregard galactic features and only work with galaxy positions. In such cases, no initial node features are provided, and the initial graph blocks are modified to perform

$$\mathbf{e}_{ij}^{(1)} = \phi_1(\mathbf{e}_{ij}^{(0)}) \quad (9)$$

and

$$\mathbf{h}_i^{(1)} = \psi_1([\bigoplus_j \mathbf{e}_{ij}^{(1)}, \mathbf{u}]), \quad (10)$$

while the rest of the graph blocks present the same form of Eq. 6 and Eq. 7.

It is well known that residual layers, such as adding the input of the layer to its output, can ease the training process, leading to smoother loss functions (see e.g. Li et al. 2017). We make use of residuals layers in all node

and edge models, excluding the first graph block (due to the different dimensionality in their feature spaces). In such cases, we modify the output of the block  $l+1$  as  $\mathbf{e}_{ij}^{(l+1)} \leftarrow \mathbf{e}_{ij}^{(l+1)} + \mathbf{e}_{ij}^{(l)}$  and  $\mathbf{h}_i^{(l+1)} \leftarrow \mathbf{h}_i^{(l+1)} + \mathbf{h}_i^{(l)}$ .

The architecture of our model consists in  $L$  graph blocks, where  $L$  is a hyperparameter to be optimized. After the  $L$  graph blocks, an additional final layer is used, which aggregates the information for each node on the graph to predict an array of outputs  $\mathbf{y}$ ,

$$\mathbf{y} = \xi([\bigoplus_{i \in \mathcal{G}} \mathbf{h}_i^{(L)}, \mathbf{u}]), \quad (11)$$

where  $\xi$  is another MLP, and  $\bigoplus_{i \in \mathcal{G}}$  is the multi-pooling operation defined in Eq. 8.

Note that by construction, the node and edge models are permutation equivariant, which means that if a permutation is applied to the ordering of nodes and edges, their hidden representations are transformed in the same way, as desired in GNNs (Bronstein et al. 2021). Furthermore, the output of the network  $\mathbf{y}$  is by construction permutation invariant. Moreover, as discussed above, our model is also translation and rotational invariant, due to the way the edge features are taken.

We emphasize that our model does not impose a cutoff on a given scale, contrary to grid-based methods such as CNNs, which are limited by the resolution of the grid/mesh. Treating the galaxy catalogue as a point cloud avoids such restriction, being able to extract information on all relevant scales. These are determined by the minimum distance between galaxies in the catalogue, which is about  $\sim 5 h^{-1}\text{kpc}$  at  $z = 0$  for both IllustrisTNG and SIMBA simulations.

The implementation of the GNNs outlined in this section, CosMOGRAPHNET (Villanueva-Domingo 2022), makes use of PyTorch Geometric (Fey & Lenssen 2019) and is publicly available on GitHub [GitHub](https://github.com/PabloVD/CosmoGraphNet).<sup>6</sup>

### 3.3. Loss function

<sup>6</sup> <https://github.com/PabloVD/CosmoGraphNet>

The loss function is chosen depending on the task to be carried out by the model. In this work we consider two main tasks:

- **Regression.** We will use GNNs to predict the power spectrum of a set of galaxies. In this case, the target  $\mathbf{y}$  is a vector with the values of the power spectrum  $P(k)$  at each wavenumber  $k$ , for 79 intervals in  $k$  from 0.3 to 20  $h\text{Mpc}^{-1}$ . For this task we employ the traditional mean squared loss function.
- **Inference.** We will also use GNNs to perform likelihood-free inference on the value of the cosmological parameters. In this case, the network is trained to predict the posterior mean  $\mu_a$  and standard deviation  $\sigma_a$  of the considered parameter  $a$  (either  $\Omega_m$  or  $\sigma_8$ ):

$$\mu_a(\mathcal{G}) = \int_{\theta_a} p(\theta_a|\mathcal{G})\theta_a d\theta_a , \quad (12)$$

$$\sigma_a^2(\mathcal{G}) = \int_{\theta_a} p(\theta_a|\mathcal{G})(\theta_a - \mu_a)^2 d\theta_a , \quad (13)$$

where  $\theta_a$  is the value of the considered cosmological parameter,  $\mathcal{G}$  represents the graph, and  $p(\theta_a|\mathcal{G})$  is the marginal posterior over the parameter  $a$

$$p(\theta_a|\mathcal{G}) = \int_{\theta} p(\theta_1, \theta_2, \dots, \theta_n|\mathcal{G})d\theta_1 \dots d\theta_{a-1} d\theta_{a+1} \dots d\theta_n . \quad (14)$$

The output of the GNN is hence  $\mathbf{y} = [\mu_a, \sigma_a]$ . This task can be carried out by using a particular form for the loss function following [Jeffrey & Wandelt \(2020\)](#). The specific form of the loss function we use to perform this task is

$$\begin{aligned} \mathcal{L} = & \log \left( \sum_{i \in \text{batch}} (\theta_{a,i} - \mu_{a,i})^2 \right) \\ & + \log \left( \sum_{i \in \text{batch}} ((\theta_{a,i} - \mu_{a,i})^2 - \sigma_{a,i}^2)^2 \right) \end{aligned} \quad (15)$$

where indexes  $i$  indicate each galaxy catalogue. We refer the reader to [Jeffrey & Wandelt \(2020\)](#) and [Villaescusa-Navarro et al. \(2022c\)](#) for further details on this loss function.

#### 3.4. Accuracy metrics

We consider several metrics to assess the accuracy of our models:

- **Mean relative error,**  $\epsilon$ , defined as

$$\epsilon = \frac{1}{N} \sum_i^N \frac{|y_{\text{truth},i} - y_{\text{infer},i}|}{y_{\text{truth},i}} , \quad (16)$$

where  $N$  is the number of galaxy catalogues in the test set and  $y_{\text{truth},i}$  and  $y_{\text{pred},i}$  are the truth and predicted targets respectively. Note that for the inference task,  $y_{\text{truth},i}$  and  $y_{\text{pred},i}$  correspond to the true value of the parameter ( $\theta_{a,i}$ ) and posterior mean ( $\mu_{a,i}$ ) for a given parameter  $a$  (either  $\Omega_m$  or  $\sigma_8$ ), while for regression, they correspond to each component of the array  $\mathbf{y}$ .

- **Coefficient of determination,**  $R^2$ , defined as

$$R^2 = 1 - \frac{\sum_i^N (y_{\text{truth},i} - y_{\text{pred},i})^2}{\sum_i^N (y_{\text{truth},i} - \bar{y}_{\text{truth}})^2} , \quad (17)$$

with  $\bar{y}_{\text{truth}}$  the mean of true values.

- $\chi^2$ , defined as

$$\chi^2 = \frac{1}{N} \sum_{i=1}^N \frac{(\theta_{a,i} - \mu_{a,i})^2}{\sigma_{a,i}^2} . \quad (18)$$

and only used for the inference task. A value close to 1 indicates that the standard deviations are correctly predicted<sup>7</sup>, while a larger (lower) value can be seen as an underestimation (overestimation) of the uncertainties.

#### 3.5. Hyperparameter optimization

We perform hyperparameter optimization following a Bayesian-based procedure with the Tree Parzen Estimator (TPE, [Bergstra et al. 2011](#)), making use of the Python package Optuna<sup>8</sup> ([Akiba et al. 2019](#)) to maximize the accuracy of the model. The hyperparameters we optimize are: 1) the learning rate, 2) the number of graph blocks, 3) the linking radius, and 4) the number of hidden features in each MLP.

The tuning is performed by searching the value of the hyperparameters that minimize the validation loss of the model. We perform a minimum of 30 trials, where each trial corresponds to the result of training the model with a given value of the hyperparameters.

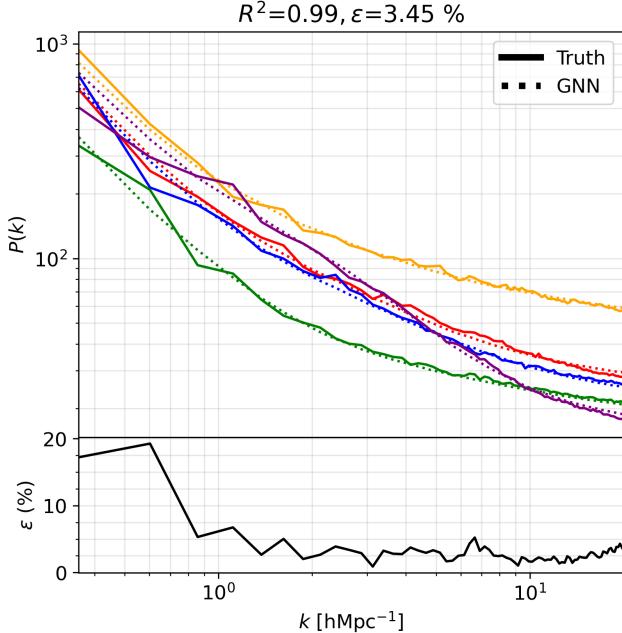
### 4. LEARNING CLUSTERING

We know that on sufficiently large scales, where a cosmic field resembles a Gaussian distribution, the power spectrum is the optimal estimator to extract all available information. Computing the power spectrum of a distribution of points is a non-trivial task that requires several steps<sup>9</sup>: 1) assign the points to a regular grid, 2)

<sup>7</sup> This can be seen as minimizing the second term of Eq. 15.

<sup>8</sup> <https://optuna.readthedocs.io>

<sup>9</sup> Note that there are different methods to estimate the power spectrum.



**Figure 4.** We have trained GNNs to investigate whether they can learn to compute the power spectrum of a distribution of points, using IllustrisTNG and SIMBA galaxy catalogues at different redshifts as the training dataset. The top panel shows with solid lines power spectra from galaxy catalogues in the test set, while the predictions of the model are displayed with dotted lines. The bottom panel depicts the mean relative error (averaged over the full test dataset) as a function of  $k$ . As can be seen, our model is more precise on small scales, where the relative error can be as small as 2.5%, while on larger scales it increases up to 20%. This trend can be explained taking into account that the network seems to have learned to compute a smoothed version of the power spectrum. Because of cosmic variance effects, the power spectrum is expected to be smoother on small scales and noisier on large scales, explaining why the accuracy decreases on large scales.

perform a Fourier transform, 3) compute the square of the modulus of each mode  $\delta(\vec{k})^2$ , and 4) compute the mean value of  $\delta(\vec{k})^2$  for all modes in a given  $k$ -interval. This computation involves evaluating the clustering of the points on a variety of scales, from large to small. Here we want to investigate whether GNNs are able to learn to compute clustering properties of a set of points such as their power spectrum.

We train GNNs on galaxy catalogues from both IllustrisTNG and SIMBA simulations that only contain galaxy positions. We note that in this case there is no need to separate the IllustrisTNG and SIMBA catalogues, as we are only interested in training a model that takes a distribution of points and outputs its power spectrum. For the same reason, we can also use galaxy catalogues at different redshifts simultaneously to increase our training dataset. We thus use catalogues at redshifts 0, 1, and 2. We emphasize that the graphs built from these catalogues only contain the position of the galaxies, that are embedded in the edge attributes. Hence, these graphs do not contain initial node features;

we thus made use of Eqs. 9 and 10 for the first graph block, as explained in Sec 3.2. The target of our network  $\mathbf{y}$  is an array including the values of the power spectrum  $P(k)$  at each wavenumber  $k$ , for 79 values of  $k$  from 0.3 to  $20 \text{ hMpc}^{-1}$ .<sup>10</sup>

Top panel of Fig. 4 shows the target (solid lines) and predicted (dotted lines) power spectra for 5 different catalogues from CAMELS simulations. We find that our model accurately predicts the galaxy power spectrum with a mean relative error of  $\epsilon \simeq 3.5\%$  and a  $R^2$  coefficient of 0.99. The bottom panel depicts the mean relative error (averaged over the full test dataset) as a function of  $k$ . As can be seen, our model is more precise on small scales, where the relative error can be as small as 2.5%, while on larger scales it increases up to 20%. This trend can be explained taking into account that the network seems to have learned to compute a smoothed version of the power spectrum. Because of cosmic variance effects, the power spectrum is expected to be smoother on small scales and noisier on large scales, explaining why the accuracy decreases on large scales.

We have tried to improve the accuracy of our model by training GNNs with loss functions that give more weight to the amplitude of the power spectrum on large scales. However, we did not find a significant increase in the accuracy of the model. We however believe that our results can be improved in many different ways, like using a larger dataset for training, using a more complex model, and using data with very different clustering properties on different scales. We leave all this for future work.

Nonetheless, these results show that GNNs are capable of learning to compute the clustering properties of a distribution of points. However, while this task has been carried out using galaxy catalogues of CAMELS simulations, it is not clear how well it could extrapolate when using it to compute the power spectrum from an arbitrary distribution of points. We have evaluated the extrapolation performance of this model, trained in CAMELS, by testing it on several synthetic catalogues with different clustering properties. We find that in general, the GNN fails to reproduce the correct amplitude of the power spectrum on large scales of the synthetic catalogues, while it works much better on small scales. We provide further details on this test in the Appendix A.

It is interesting to see that the hyperparameter optimization method selected models that only contain one

<sup>10</sup> Note however than even smaller scales can be accessed through our graphs.

graph block with a large linking radius. Surprisingly, the linking radii in these models is around  $1 h^{-1}\text{Mpc}$  (the value used in Fig. 4), that produces graphs where some nodes are isolated and therefore the model can not be using information from all galaxies when performing the regression. We believe that these models work because the GNN exploits the correlations between large and small scales to infer the former from the latter. If most of the information would be located on large scales, our model would prefer a large linking length and will avoid isolated nodes. This would be the case if our galaxy distribution would follow a Gaussian distribution, whose scales are uncorrelated. However, in this case, most of the information is located on small scales, which have a larger signal-to-noise ratio, and the graph is constructed to exploit that. While larger linking radii can slightly improve the relative error down to  $\sim 2\%$ , they lead to worse performance for the extrapolation test in synthetic distributions described in Appendix A. Note also that larger radii imply denser graphs (with more edges), that increase the memory needed to store and manipulate them.

## 5. LEARNING COSMOLOGY FROM GALAXIES

In the previous section we showed that GNNs can learn to compute clustering properties of point distributions. Since we do not know what is the optimal estimator to extract the maximum cosmological information from galaxy catalogues, we aim to use GNNs to perform such task. Our GNNs are trained to perform field-level likelihood-free inference on the value of the cosmological parameters while marginalizing over astrophysical uncertainties, as modelled in the CAMELS simulations.

We first trained our GNNs to infer simultaneously  $\Omega_m$  and  $\sigma_8$ , finding that the constraints on  $\sigma_8$  were very loose. We then trained models to infer  $\Omega_m$  and  $\sigma_8$  separately, and found that the bounds on  $\Omega_m$  improved significantly. Thus, from now on we only report results from models that are trained to infer the value of  $\Omega_m$  alone. The results for  $\sigma_8$  are shown in the Appendix B.

We start showing the results we obtain from galaxy catalogues that only contain galaxy positions. We then investigate how much information is gained by considering different galaxy properties together with galaxy positions. Next, the present the results of the tests performed to check the robustness of the models.

### 5.1. From galaxy positions to $\Omega_m$

We start by investigating whether it is possible to infer the value of  $\Omega_m$  from the 3D spatial distribution of galaxies. We train and test the models using galaxy catalogues that only include galaxy positions from the

IllustrisTNG and SIMBA simulations. Since this is the only property these catalogues contain, the cosmological information can only be extracted from the clustering properties of the galaxies, which can be learned in principle by a GNN, as shown in Sec. 4. The results are shown in the top row of Fig. 5. We find that the models are able to infer the value of  $\Omega_m$  with a mean relative error of  $\epsilon \sim 12 - 13\%$  and a coefficient of determination of  $R^2 \simeq 0.83 - 0.85$ . The  $\chi^2$  has values close to unity,  $\simeq 1.2$ , indicating that error bars are accurately predicted.

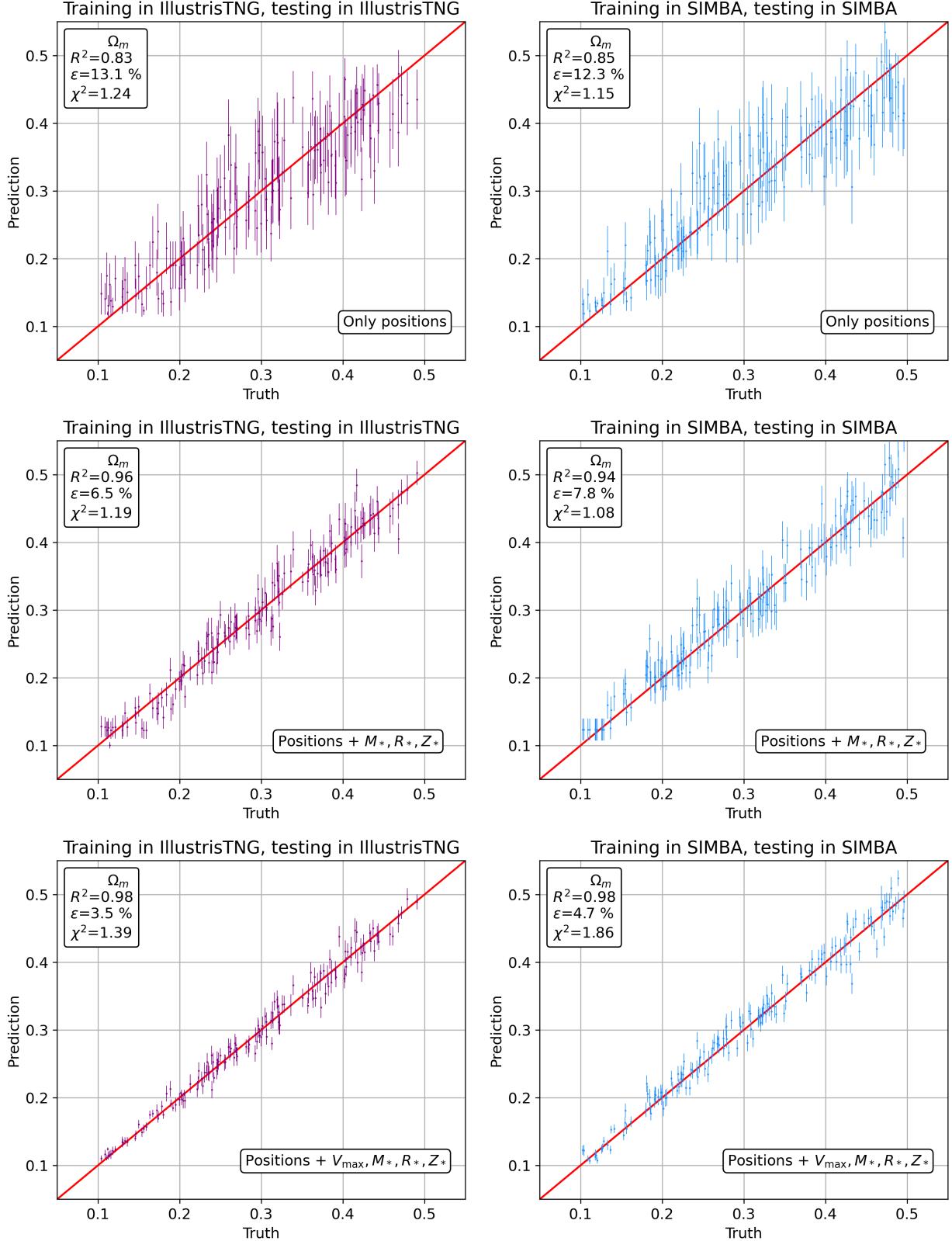
It is interesting to look at the preferred architectures found through hyperparameter optimization, that shows that the linking radius has the largest impact on the performance, preferring values between 0.1 and  $0.4 h^{-1}\text{Mpc}$ , while the specific value of the number of graph blocks and hidden channels present a milder effect. This relatively small value of the linking radius will leave some nodes without edges (see Fig. 2). We believe that this may be happening because most of the information resides on small scales; a large value of  $r$  may mix information from small with large scales in a suboptimal manner.

There are other works in the literature that have performed similar tasks. We note however that a comparison with those works is difficult due to the very different setups:

- Ntampaka et al. (2020) employed convolutional neural networks to constrain  $\Omega_m$  from the 3D galaxy distributions with an accuracy of  $\sim 4\%$ . However, their galaxy number density is lower than ours and their testing volume is more than 4000 times larger than ours. They also use galaxy catalogues obtained from HODs rather than full hydrodynamic simulations. On the other hand they incorporate effects from supersample covariance that we miss and, contrary to us, they were able to constrain  $\sigma_8$ .
- Perez et al. (2022) performed likelihood-free inference on several summary statistics from galaxy catalogues obtained from semi-analytics models. Their constraints on  $\Omega_m$  are better than ours, but their effect volume is also much larger, with simulations boxes of  $100 h^{-1}\text{Mpc}$  of length.

#### 5.1.1. Interpretability

Understanding where the information to infer the cosmological parameter comes from is not obvious. We have performed several tests to gain some physical intuition on its origin.



**Figure 5.** We have trained GNNs to perform likelihood-free inference on the value of  $\Omega_m$  from galaxy catalogues that contain: 1) only galaxy positions (top row), 2) galaxy positions, stellar masses, radii, and metallicities (middle row), and 3) galaxy positions, stellar masses, radii, metallicities, and  $V_{\max}$  (bottom row). The left column show the results of the models that have been trained and tested on catalogues from IllustrisTNG simulations, while the right column displays the equivalent for the SIMBA simulations. The numbers in the legend indicate the value of  $R^2$ , the mean relative error ( $\epsilon$ ), and  $\chi^2$  as defined in Eq. 18. As can be seen, in all cases our models are able to infer the value of  $\Omega_m$  accurately. As expected, when considering more galaxy properties, the constraints improve.

- The number of galaxies in a catalog could be expected to be correlated to the value of  $\Omega_m$ . However, while we have checked that there is a trend showing that simulations with higher values of  $\Omega_m$  contain more galaxies, the scatter is too large to be the reason that would explain our results. Therefore, the GNN must be employing some clustering information.
- We can compare the constraints our model is placing on  $\Omega_m$  versus the ones that can be set using standard summary statistics like the power spectrum. We have quantified how much information on  $\Omega_m$  can be extracted from the galaxy power spectrum by training a multilayer perceptron that takes as input the galaxy power spectrum computed with PYLIANS<sup>11</sup> (Villaescusa-Navarro 2018) and returns the posterior mean and standard deviation for  $\Omega_m$ . We find that the best model (after optimizing the hyperparameters) provides a mean relative error of  $\epsilon \sim 30\%$  and a correlation coefficient between the ground truth and the output of  $R^2 \simeq 0.5$ . We note that in this case the network is almost predicting the mean of the entire distribution with large errorbars, so this relative error of 30% is already heavily affected by the distribution priors. Thus, the gain from the power spectrum to the field-level inference is likely much larger than a factor of 3. This test illustrates how field-level inference yields much tighter constraints on the value of the cosmological parameters than the ones obtained from summary statistics that are suboptimal, like the power spectrum for non-Gaussian density fields.
- Summary statistics related to distribution and number of neighbors are known to carry out a large amount of cosmological and clustering information (Banerjee & Abel 2021a,b). As an additional test, given a linking radius, we have computed the degree for each node, i.e., the number of neighbors, and generated an array of the degree distribution for each catalog. Then, a standard MLP has been trained to infer  $\Omega_m$ . After tuning the hyperparameters (including the MLP depth and the linking radius), the best model can achieve a mean relative error of  $\epsilon \sim 18\%$  (with  $R^2 \simeq 0.74$ ). This result illustrates how connectivity between galaxies carries meaningful cosmological information. Indeed, it outperforms the previous MLP trained

with the galaxy power spectrum. Nonetheless, a GNN can exploit clustering information in a more meaningful way to improve the degree distribution MLP, achieving  $\sim 35\%$  higher accuracy.

From the above tests, it seems that the GNN is able to extract non-trivial clustering information which is absent in traditional and non-standard summary statistics, obtaining a better accuracy at inferring the cosmological parameter.

## 5.2. From galaxy positions and intrinsic features to $\Omega_m$

In the previous subsection we have shown how the clustering of galaxies can be used to infer the value of  $\Omega_m$ . We now study how much cosmological information can be extracted if we use both the clustering of galaxies and their internal properties. For this, we made use of galaxy catalogues that contain 1) galaxy positions, 2) stellar mass  $M_*$ , 3) stellar radius  $R_*$ , 4) stellar metallicity  $Z_*$ , and/or 5) maximum circular velocity  $V_{\max}$ .

We have trained GNNs using galaxy catalogues from both the IllustrisTNG and SIMBA simulations. The results of testing the models with catalogues from the same simulation suites are shown in the bottom row of Fig. 5. Our networks can accurately predict  $\Omega_m$  in both simulation suites within a mean relative error of  $\epsilon \simeq 3.5 - 4.7\%$  and with a coefficient of determination of  $R^2 \simeq 0.98$ . From the  $\chi^2$  values we can see that the error bars for the model trained on SIMBA can be slightly underpredicted.

In terms of hyperparameter optimization, we find that the best models prefer small number of hidden neurons (64) and linking radii of  $\sim 0.4 h^{-1}\text{Mpc}$ , which is the most relevant hyperparameter, while the number of graph blocks does not have a significant impact.

To further understand where does the information comes from, we have performed several tests:

- In order to quantify how much information comes from the galaxy features versus galaxy clustering, we have trained a DeepSet (Zaheer et al. 2017), a simplified version of a GNN, which does not include neither edges nor message passing layers. It only employs node features, that are updated according to  $\mathbf{h}_i^{(l+1)} = \psi_{l+1}(\mathbf{h}_i^{(l)})$  (instead of Eq. 7). With that architecture, we find the accuracy of the model to reach mean errors of  $\epsilon \sim 4 - 5\%$ . Since this model, by construction, cannot extract information from clustering, we conclude that galaxy features are responsible for most of the information extracted from the GNNs.
- We have trained models using only one out of the four galactic properties. We find that the most im-

<sup>11</sup> <https://github.com/franciscovillaescusa/Pylians3>

portant feature appears to be the maximum circular velocity. Using  $V_{\max}$  as the only node feature leads to accurate results, with mean relative errors around 5 % and  $R^2 \simeq 0.97$ . We note that Villaescusa-Navarro et al. (2022b) found  $V_{\max}$  to be the most relevant feature when inferring the value of  $\Omega_m$  from individual galaxies. This could be due to the fact that  $V_{\max}$  measures the depth of the gravitational potential, that may correlate better with  $\Omega_m$  than the other considered galaxy properties.

- We find that models trained using  $M_*$ ,  $R_*$ , and  $Z_*$  as galaxy properties are still accurate, with mean relative errors between 6 – 8 % and  $R^2 \simeq 0.94 – 0.96$ . This indicates that the stellar features also exhibit a strong correlation with  $\Omega_m$ . These results are shown in the middle panel of Fig. 5. This choice of galactic features presents an additional benefit since, contrary to  $V_{\max}$ , they are easily accessible from observational means, being more suitable to apply to real data.

To further assess the accuracy of our procedure, we can compare our results with previous works in the literature where  $\Omega_m$  is predicted from cosmological simulations. To establish a fair comparison, we limit this to studies using CAMELS simulations:

- In Villaescusa-Navarro et al. (2021b), the authors were able to infer the value of  $\Omega_m$  with an accuracy between 3.4% and 4.5% when using 2D total mass maps. Our results have thus comparable performance, although our analysis 1) takes place in 3D instead of 2D, 2) uses galaxies instead of the full underlying matter field.
- In Villaescusa-Navarro et al. (2021a), a similar approach was followed employing 13 cosmological fields. Those models reached accuracies around 2.5 %. We emphasize that while galaxies are directly observable, some of these 2D fields are not, or there are regions of them (like in voids and filaments) where observations will be highly affected by noise.
- In Villaescusa-Navarro et al. (2022b),  $\Omega_m$  was inferred from the properties of individual galaxies. The accuracy of their results is quantified with the root-mean-squared error,  $\sqrt{\langle (\Omega_{m,\text{truth}} - \Omega_{m,\text{pred}})^2 \rangle}$ , resulting in values of  $3.4 \times 10^{-2}$  and  $3.7 \times 10^{-2}$  for IllustrisTNG and SIMBA respectively. Using the same metric, our model outperforms those results, with root-mean-squared errors of  $1.4 \times 10^{-2}$  for both IllustrisTNG

and SIMBA. We note however that our GNN model uses only four intrinsic galactic properties, instead of the 17 features assumed in Villaescusa-Navarro et al. (2022b).

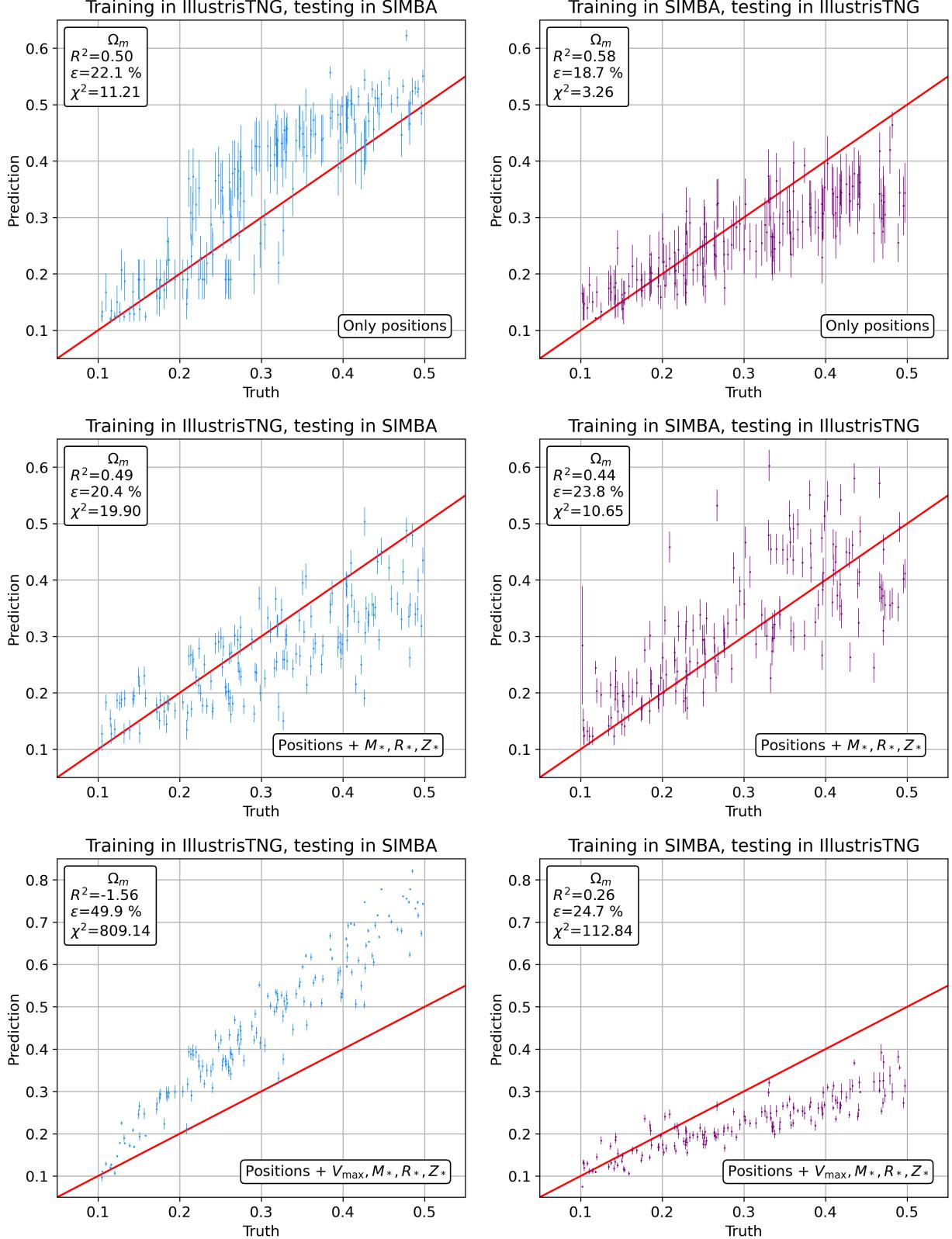
The above comparisons show that while one would expect a large gain in information by going from 2D to 3D, the usage of discrete and sparse tracers, instead of the continuous underlying field, would degrade that information. The combination of both effects yields similar results in 3D for galaxies than 2D for continuous fields.

### 5.3. Robustness tests

The simulations in the two CAMELS suites, IllustrisTNG and SIMBA, use different subgrid physics models, which can be seen as two potential different approximations to reality. Our models are trained using a large variety of astrophysical models, so that the GNNs will learn to be insensitive to the particular baryonic effects of a given simulation. In other words, our models are trained to learn to marginalize over baryonic effects within an assumed subgrid physics model. While these ML algorithms learn to extract clustering and cosmological information within a given suite, it is not clear whether they could still perform well when the underlying subgrid physics model is different to the one used for training.

In order to check whether our models are indeed insensitive to the baryonic modeling, we can train a model on galaxy catalogues from IllustrisTNG simulations and test the model on catalogues from SIMBA simulations (and the other way around). We note that a model that is designed to be used with real data should pass this test at the level of simulations. This kind of tests are non trivial to succeed, and have been carried out in other previous works using CAMELS simulations with diverse levels of performance (Villaescusa-Navarro et al. 2021b, 2022b; Villanueva-Domingo et al. 2022; Nicola et al. 2022; Shao et al. 2022; Wadekar et al. 2022).

We show in Fig. 6 the results of the robustness test employing the models trained in Secs. 5.1 and 5.2. Left column corresponds to networks trained in IllustrisTNG and cross tested in SIMBA, while right column corresponds to GNNs trained in SIMBA and tested in IllustrisTNG. We can see that the test fails in the two scenarios, i.e. for catalogues containing galaxy positions and for catalogues containing galaxy positions and properties. As can be seen in Fig. 6, relative errors raise up to  $\sim 20$  % when only positions (top row) or the combination  $M_*$ ,  $R_*$  and  $Z_*$  (middle row) are employed, and can be even worse if  $V_{\max}$  is also used (bottom row). The fact that the failure is less severe when only spatial distribution or the features  $M_*$ ,  $R_*$  and  $Z_*$  are consid-



**Figure 6.** Robustness test. We have tested the networks, trained on galaxy catalogues from the IllustrisTNG/SIMBA simulations, on catalogues from the SIMBA/IllustrisTNG simulations, showing the results in the left/right columns. As can be seen, our models are not robust in any of the considered scenarios. This can be due to the intrinsic differences that the distinct subgrid models imprint in both galaxy properties, abundance, and clustering. See text for details.

ered indicates that these properties are more robust than  $V_{\max}$  regarding the baryonic effects. It is interesting to see that, while training in IllustrisTNG and testing in SIMBA leads to overpredictions, training in SIMBA and testing in IllustrisTNG produces the opposite result.

It is perhaps not surprising that the models are not robust when using galaxy properties, as it is well known that the properties of the galaxies from the two different suites can be very different. Villaescusa-Navarro et al. (2022b) found that even at the level of a single galaxy properties in IllustrisTNG and SIMBA are different enough to make their model not robust.

On the other hand, it is more difficult to explain why the model that only make use of galaxy positions is not robust. We believe that this may be due to the fact that our galaxy catalogues contain a large fraction of small galaxies with stellar masses below  $10^9 h^{-1} M_\odot$  (see Fig. 1). The abundance and distribution of these galaxies may be very sensitive to the particular subgrid physics considered. For instance, the catalogues from SIMBA contain an average of 1200 galaxies, while this number lowers to 700 in the case of IllustrisTNG.

We thus conclude that our models are currently not robust enough, and therefore cannot be directly employed to perform parameter inference from galaxy catalogues of real data. The solution to this problem may not be simple and it is beyond the scope of this work.

## 6. CONCLUSIONS

The spatial distribution of galaxies embeds precious information about fundamental physics. Unfortunately, we do not know which is the optimal procedure to extract that information, as galaxies follow, on small scales, a complex non-Gaussian pattern. On the other hand, neural networks can identify unique signatures that the cosmological parameters leave on those patterns, and therefore find optimal estimators to extract the maximum amount of information from the data. In this work we have used Graph Neural Networks (GNNs) to perform likelihood-free inference at the galaxy-field level while accounting for astrophysical uncertainties, as modelled in CAMELS.

We have argued that GNNs may be more appropriate for this task than other deep learning approaches such as Convolutional Neural Networks (CNNs) because 1) they are the natural architectures to deal with sparse and irregular data, like galaxy catalogues, 2) they can extract information from all scales, while CNNs are limited to scales larger than the pixel/voxel size, 3) it is easy to en-

force the underlying symmetries such as translational<sup>12</sup> and rotational invariance.

The main take conclusions of this work are the following:

- We have shown that GNNs can be trained to predict non-trivial clustering properties of 3D distributions of points, like their power spectrum. This is our motivation to use GNNs to find optimal estimators that can extract the maximum information at the galaxy-field level.
- We have trained GNNs to perform a field-level inference on the value of the cosmological parameters from galaxy catalogues generated from the hydrodynamic simulations of the CAMELS project while accounting for uncertainties arising from baryonic effects. We emphasize that our models are rotational and translational invariant, and can extract information from any available scale down to the minimum distance between galaxies:  $\sim 5 h^{-1} \text{kpc}$  in the CAMELS simulations at  $z = 0$ .
- We have shown that our models can infer the value of  $\Omega_m$  from galaxy catalogues covering volumes as small as  $(25 h^{-1} \text{Mpc})^3$  and containing only  $\sim 10^3$  galaxies with an accuracy between  $\sim 4\%$  to  $\sim 13\%$  at  $z = 0$ .
- While our networks are able to extract information from the positions of galaxies alone, including other galaxy properties such as stellar mass, stellar radius, stellar metallicity, and maximum circular velocity can yield significantly tighter constraints. Most of the information our networks are extracting are due to galaxy properties and not to galaxy clustering.
- Unfortunately, our models are not completely robust, i.e. a model trained on galaxy catalogues from IllustrisTNG simulations do not perform well when tested on galaxy catalogues from SIMBA simulations. Therefore, our models cannot be used with real data yet.

### 6.1. Robustness

This work highlights the large amount of cosmological information that resides on small, non-linear, scales, and how machine learning methods can be used to extract it. On the other hand, we also showed how the models do not extrapolate well, perhaps indicating that the

<sup>12</sup> We note that translational invariance is also enforced with CNNs.

models are extracting too much information. This is a big obstacle that prevents the deployment of our models to be used with real data. We note that the lack of robustness has been recently seen in other works involving the CAMELS simulations (see e.g. Villaescusa-Navarro et al. 2021a, 2022b).

We emphasize that all our models exhibit a lack of robustness, although the models that only made use of galaxy positions are more robust than those employing all galactic features considered. There could be several reasons that explain the latter. First, even if the CAMELS simulations cover a very large volume in the astrophysical parameter space, perhaps the simulations in the IllustrisTNG and SIMBA suites do not fully overlap, so marginalizing over one does not guarantee the proper marginalization over the other. This can already be seen in Fig. 1, that shows how galaxy properties only overlap partially. Second, the networks may be learning numerical artifacts that are distinct in simulations with different subgrid physics. Third, perhaps the representation of the data is slightly different in the two simulations, leading to different dimensions where simulations do not overlap.

The fact that the models trained using only galaxy positions are also not robust is more worrisome. This may be happening because our galaxy catalogues contain a non-negligible fraction of small galaxies, whose abundance and clustering may be largely affected by baryonic effects. If this would be the case, these small galaxies can be seen as a challenge for hydrodynamic simulations. We believe it is important to investigate whether our conclusions hold when using galaxy catalogues containing more massive galaxies.

The volume covered by the galaxy catalogues may also be an important piece affecting the model robustness. On sufficiently large scales, we would expect the models using only galaxy positions to be robust. However, in this case, the field-level inference may not bring substantial improvement over the traditional power spectrum. If the problem is instead coming from the clustering on small scales, it will be important to develop models that do not make use of that information. For instance, one can create graphs from galaxy catalogues by setting edges only if the distance between two galaxies is above some scale.

## 6.2. Future work

We outline some tasks to be carried out in future work:

- In this work we have worked with galaxy catalogues in real-space. However, galaxies are observed in redshift-space. It will be interesting to quantify how much information is gained, or

lost, by performing the analysis with galaxy catalogues in redshift-space. In this case the symmetries of our models should be changed: the axis along which redshift-space distortions is implemented will break down the rotational symmetry considered here.

- We have made use of galaxy catalogues at  $z = 0$  for inferring  $\Omega_m$ . It is important to train networks with galaxy catalogues at redshifts higher than 0 and investigate how much information is gained, or lost, and whether the models are more robust at higher redshifts.
- It is critical to quantify the effects of volume in our results. Using galaxy catalogues covering larger volumes will help addressing 1) how much information is coming from clustering versus internal galaxy properties as function of scale, 2) whether the models are more robust with larger volumes or more massive galaxies, and 3) whether the model can infer  $\sigma_8$  and potentially other parameters. Galaxy catalogues from CAMELS-SAM (Perez et al. 2022) and Molino (Hahn & Villaescusa-Navarro 2021b), among others, will be perfect for this task.
- In this work we have not taken into account the effects of super-sample covariance, i.e., the effect of perturbations on scales larger than the volume of our simulation boxes. Addressing how much information is gained or lost by this is not only interesting, but needed in order to build models that can be used with real data. This can be achieved in several ways, like running simulations with larger volumes or using separate-Universe simulations (Li et al. 2014, 2018; Barreira et al. 2019).
- It will also be interesting to develop more interpretable models. For instance, for GNNs with one, or few, blocks, it may be possible to use symbolic regression to find analytic expressions that approximate the behaviour of the network (Cranmer et al. 2020; Lemos et al. 2022). In this case, it may be possible to understand what operation the network is doing and make it robust to changes in subgrid physics.
- It would be important to investigate whether training in galaxy catalogues from both IllustrisTNG and SIMBA make the models more robust, and whether they will work on simulations from a third suite. Alternatively, contrastive learning techniques can be applied to build models

which are by construction insensitive to the differences between subgrid models.

## DATA AVAILABILITY

The implementation of the GNNs underlying this article, COSMOGRAPHNET (Villanueva-Domingo 2022), is available on GitHub [GitHub](https://github.com/PabloVD/CosmoGraphNet)<sup>13</sup>. Details on the

CAMELS simulations can be found in <https://www.camel-simulations.org>.

## ACKNOWLEDGEMENTS

We thank the CAMELS team for useful conversations. The work of FVN is supported by the Simons Foundation. The training of the GNNs has been carried out using GPUs from the Tiger cluster at Princeton University.

## APPENDIX

### A. POWER SPECTRUM EXTRAPOLATION

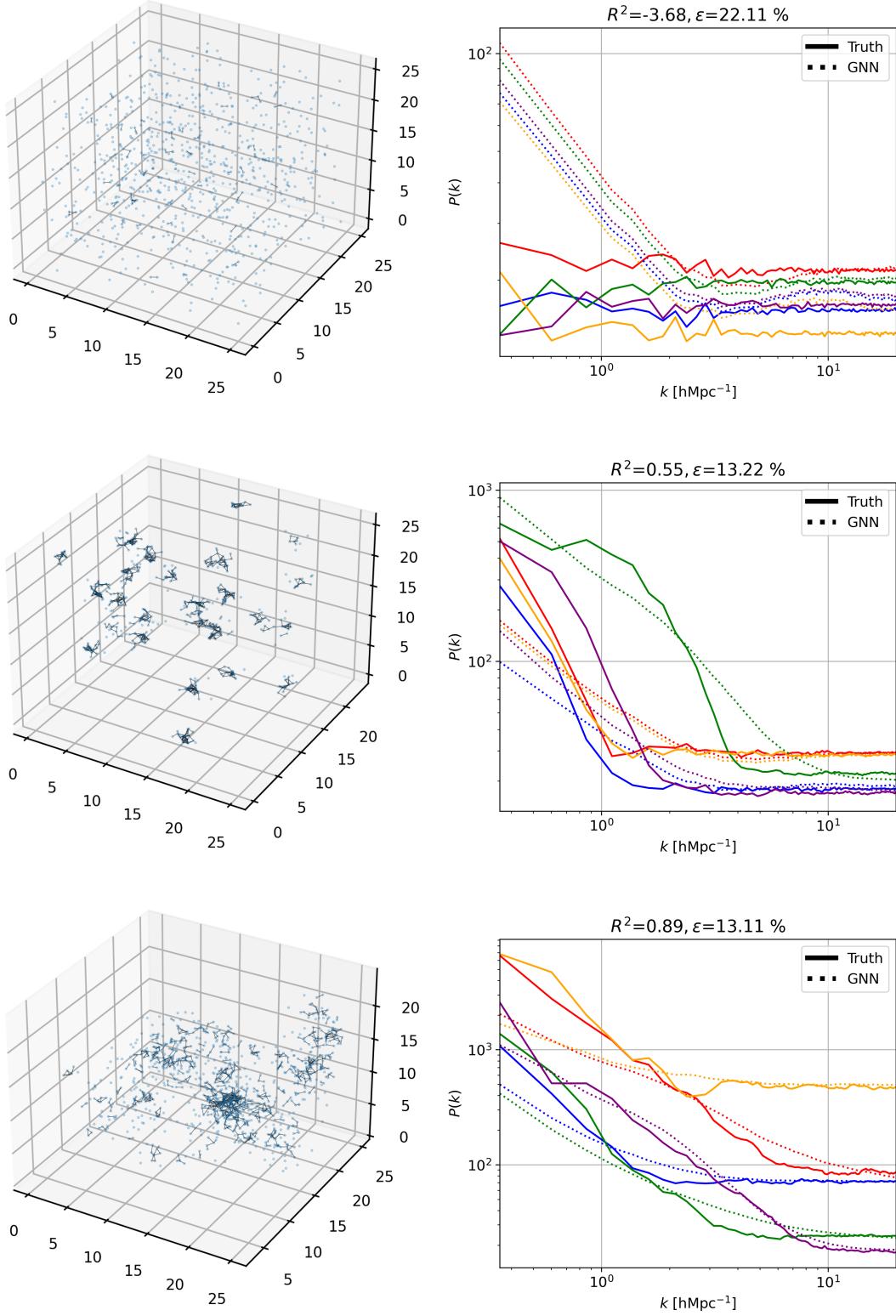
In Sec. 4 we trained a GNN using CAMELS galaxy catalogues to predict their power spectrum. We now investigate whether our model has really learned to compute the power spectrum by testing it on distributions of points with very different clustering patterns to the ones used for training. For this task we consider catalogues with three different clustering models:

- **Poisson process.** This is simply a random distribution of points within the considered volume. It is well known that such distribution exhibits a white-noise power spectrum with an amplitude equal to  $V/N$ , where  $V$  is the volume and  $N$  the number of points.
- **Neyman-Scott process** (Neyman & Scott 1952). This is a clustered distribution formed by two nested Poisson processes. A set of cluster centers are randomly spread within a box following a Poisson process. Then, a cluster of points is distributed around each center, following a given distribution, that we take as a Gaussian in this case. The correlation function can be computed analytically, resulting in a Gaussian function (Illian et al. 2008). The power spectrum can thus be derived from the correlation function, showing a Gaussian dependence on the wavenumber  $k$ .
- **Soneira-Peebles model** (Soneira & Peebles 1977, 1978; Peebles 1980). This is a fractal hierarchical distribution and is built as follows. A set of cluster centers are first randomly placed. Then,  $\eta$  daughter points are placed around each cluster center following a Gaussian distribution, with a width given by a radius  $R$ . Each of these points becomes the center of a new cluster, around which  $\eta$  new points are placed within a width  $R/\lambda$ . The process is repeated for  $L$  times, generating a multilevel structure of clusters where at a level  $l$  there is a cluster with size  $R/\lambda^l$ . Given its self-similar nature, this point process leads to a scale-free power spectrum (a power law in  $k$ ) in the range  $1/R < k < 1/(R/\lambda^L)$ .

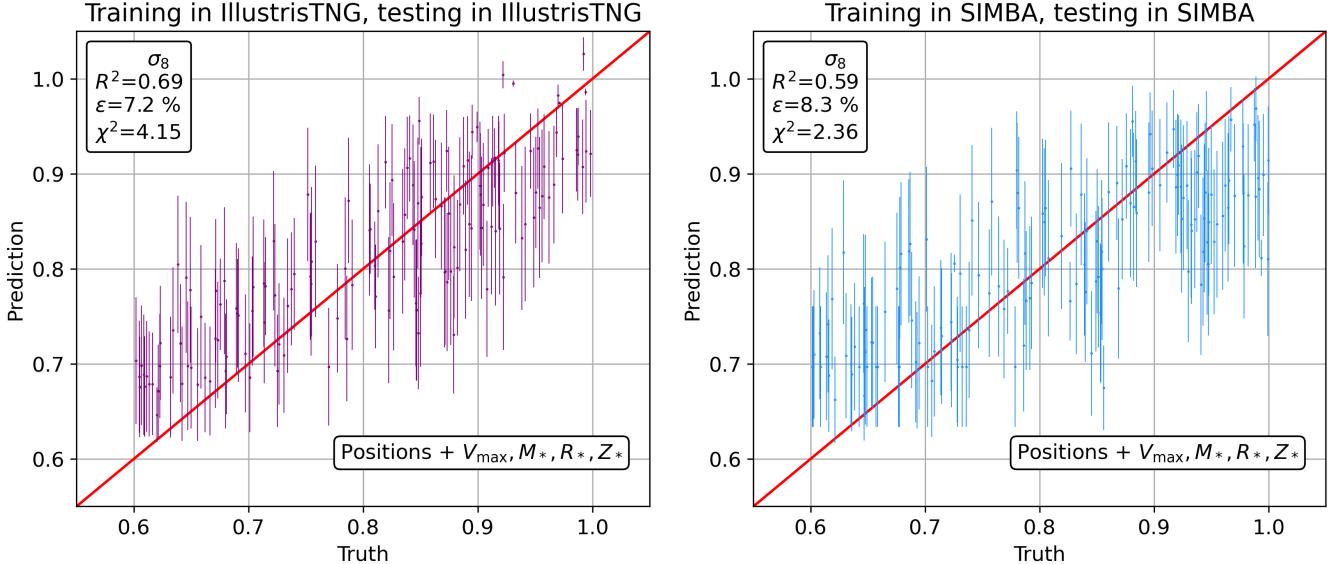
For each clustering model we generate 50 catalogues, each of them having between 700 and 1200 points, similar to the mean number density of galaxies in the CAMELS catalogues. We then feed these catalogues to the GNN, that has been trained in CAMELS galaxy catalogues, to obtain their power spectrum. We compute the true power spectrum of these point catalogues using PYLIANS and we show in Fig. 7 the comparison between the true power spectra and the network predictions. We can see that the network is partially able to predict the amplitude of the power spectrum on small scales,  $k > 5 - 10 \text{ hMpc}^{-1}$ . Even though a small offset can typically appear with respect to the true power spectrum in the Poisson scenarios, it works well for the Neyman-Scott and Soneira-Peebles catalogues.

On large scales, however, the GNN is barely able to recover the true power spectrum. There are several reasons that may explain why the network fails in this regime (at low  $k$ ). We may be tempted to justify this behaviour due to the presence of cosmic variance. However, we can see that the predicted shape of the power spectrum is pretty similar in all cases, being different to the true one. We suspect that since the model has been trained on galaxy catalogues from hydrodynamic simulations, were the volume is small and all scales may be tightly coupled, the model may be using these correlations to determine the amplitude and shape of the power spectrum on large scales. It may be possible

<sup>13</sup> <https://github.com/PabloVD/CosmoGraphNet>



**Figure 7.** We have quantified how accurate our GNN, that has been trained on galaxy catalogues to predict its power spectrum, extrapolates by using it to predict the power spectrum of point distributions with very different clustering properties: Poisson process (top), Neyman-Scott (middle), and Soneira-Peebles fractal hierarchical clustering (bottom). The left column shows examples of the graphs for these point distributions, while the right column displays the true power spectra (solid) versus the prediction of the model (dotted). As can be seen, the model is not very accurate at predicting the correct amplitude of the power spectrum on large scales. We believe that this may happen because the model has learned some correlations that are only present in the galaxy catalogues it was trained on. See text for details.



**Figure 8.** We have trained GNNs to infer the value of  $\sigma_8$  from galaxy catalogues that include galaxy positions,  $M_*$ ,  $R_*$ ,  $Z_*$  and  $V_{\max}$  on simulations from the IllustrisTNG (left) and SIMBA (right) suites. As can be seen, our models are not able to infer the value of  $\sigma_8$  accurately.

that the model is just using information from small scales to determine the amplitude of clustering on those scales and then made use of the correlations of small and large scales to infer the power spectrum on large scales.

While in Sec. 4 it has been shown how GNNs can learn to accurately compute non-trivial summary statistics from a distribution of points, the tests above illustrates how the learned procedure may not truly correspond to the traditional methods, and therefore, fails when extrapolating it. This could happen because the network is making use of information that is characteristic of the data it is trained on. If those features are not present in other datasets the model may fail. For instance, the shot noise spectrum is not well recovered in some cases. While a GNN trained on these kind of catalogs could trivially predict that regime (since it is basically given by the number of galaxies), the shot noise contribution in the training CAMELS catalogs is negligible (see Fig. 4). Training the model on data with many different clustering properties on different scales can force the network to learn more robust and close approximations to the true power spectrum estimator.

We emphasize that the purpose of this work is not to show that GNNs can learn to compute the power spectrum for generic point distributions, but that in the case of galaxy catalogues from hydrodynamic simulations, they can learn to compute non-trivial clustering properties that may be sensitive to the value of the cosmological parameters. Developing a generic power spectrum estimator would involve training in point catalogues with very different clustering properties, number of points, volumes and other considerations, and it is out of the scope of this work.

## B. CONSTRAINTS ON $\sigma_8$

Here we report the results we obtain when training GNNs to infer the value of  $\sigma_8$  using galaxy catalogues that contain both galaxy positions and galaxy properties ( $M_*$ ,  $R_*$ ,  $Z_*$ , and  $V_{\max}$ ). We show the results in Fig. 8 for the IllustrisTNG (left) and SIMBA (right) simulations. As can be seen, our models are unable to provide accurate constraints on  $\sigma_8$ . There are several reasons that could explain such bad performance. On the one hand, this could be due to degeneracies with other parameters. For instance, the cosmological effects resulting from varying  $\sigma_8$  could be mimicked by modifying one or some of the astrophysical parameters considered in CAMELS, and in such case, the network would not be able to discern the specific value. On the other hand, the volume of the simulation boxes,  $(25 h^{-1}\text{Mpc})^3$ , may be too small to properly capture the effect of  $\sigma_8$  when the field is sampled with galaxies. Training our models on simulations covering larger volumes should enable us to constrain  $\sigma_8$  with higher accuracy. We will explore this in future work.

We have checked that training only with galaxy positions leads to even worse results, where the network hardly predicts outputs other than a constant mean value with large error bars. It is worth to note that  $\sigma_8$  has been

predicted with far better accuracy from different 2D fields in CAMELS (Villaescusa-Navarro et al. 2021b,a), while Villaescusa-Navarro et al. (2022b) showed worse performance than ours using properties from individual galaxies.

## REFERENCES

- Ajani, V., Peel, A., Pettorino, V., et al. 2020, Phys. Rev. D, 102, 103531, doi: [10.1103/PhysRevD.102.103531](https://doi.org/10.1103/PhysRevD.102.103531)
- Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. 2019, arXiv e-prints, arXiv:1907.10902. <https://arxiv.org/abs/1907.10902>
- Allys, E., Marchand, T., Cardoso, J.-F., et al. 2020, Physical Review D, 102, doi: [10.1103/physrevd.102.103506](https://doi.org/10.1103/physrevd.102.103506)
- Banerjee, A., & Abel, T. 2021a, MNRAS, 500, 5479, doi: [10.1093/mnras/staa3604](https://doi.org/10.1093/mnras/staa3604)
- . 2021b, MNRAS, 504, 2911, doi: [10.1093/mnras/stab961](https://doi.org/10.1093/mnras/stab961)
- Banerjee, A., Castorina, E., Villaescusa-Navarro, F., Court, T., & Viel, M. 2020, J. Cosmology Astropart. Phys., 2020, 032, doi: [10.1088/1475-7516/2020/06/032](https://doi.org/10.1088/1475-7516/2020/06/032)
- Barreira, A., Nelson, D., Pillepich, A., et al. 2019, MNRAS, 488, 2079, doi: [10.1093/mnras/stz1807](https://doi.org/10.1093/mnras/stz1807)
- Battaglia, P. W., Hamrick, J. B., Bapst, V., et al. 2018, arXiv e-prints, arXiv:1806.01261. <https://arxiv.org/abs/1806.01261>
- Bayer, A. E., Villaescusa-Navarro, F., Massara, E., et al. 2021, Detecting neutrino mass by combining matter clustering, halos, and voids. <https://arxiv.org/abs/2102.05049>
- Beck, R., & Sadowski, P. 2019, Refined Redshift Regression in Cosmology with Graph Convolution Networks
- Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. 2011, in Advances in Neural Information Processing Systems, ed. J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, & K. Q. Weinberger, Vol. 24 (Curran Associates, Inc.). <https://proceedings.neurips.cc/paper/2011/file/86e8f7ab32cfcd12577bc2619bc635690-Paper.pdf>
- Bronstein, M. M., Bruna, J., Cohen, T., & Veličković, P. 2021, arXiv e-prints, arXiv:2104.13478. <https://arxiv.org/abs/2104.13478>
- Cheng, S., & Ménard, B. 2021, MNRAS, 507, 1012, doi: [10.1093/mnras/stab2102](https://doi.org/10.1093/mnras/stab2102)
- Corso, G., Cavalleri, L., Beaini, D., Liò, P., & Veličković, P. 2020, arXiv e-prints, arXiv:2004.05718. <https://arxiv.org/abs/2004.05718>
- Cranmer, M., Melchior, P., & Nord, B. 2021. <https://arxiv.org/abs/2106.09761>
- Cranmer, M., Sanchez-Gonzalez, A., Battaglia, P., et al. 2020. <https://arxiv.org/abs/2006.11287>
- Cranmer, M. D., Xu, R., Battaglia, P., & Ho, S. 2019, arXiv e-prints, arXiv:1909.05862. <https://arxiv.org/abs/1909.05862>
- Dai, J.-P., Verde, L., & Xia, J.-Q. 2020, J. Cosmology Astropart. Phys., 2020, 007, doi: [10.1088/1475-7516/2020/08/007](https://doi.org/10.1088/1475-7516/2020/08/007)
- Davé, R., Anglés-Alcázar, D., Narayanan, D., et al. 2019, Mon. Not. Roy. Astron. Soc., 486, 2827, doi: [10.1093/mnras/stz937](https://doi.org/10.1093/mnras/stz937)
- de la Bella, L. F., Tessore, N., & Bridle, S. 2020, arXiv e-prints, arXiv:2011.06185. <https://arxiv.org/abs/2011.06185>
- Eickenberg, M., Ally, E., Moradinezhad Dizgah, A., et al. 2022, arXiv e-prints, arXiv:2204.07646. <https://arxiv.org/abs/2204.07646>
- Fey, M., & Lenssen, J. E. 2019, Fast Graph Representation Learning with PyTorch Geometric, 2.0.2. [https://github.com/pyg-team/pytorch\\_geometric](https://github.com/pyg-team/pytorch_geometric)
- Fluri, J., Kacprzak, T., Lucchi, A., et al. 2019, Phys. Rev. D, 100, 063514, doi: [10.1103/PhysRevD.100.063514](https://doi.org/10.1103/PhysRevD.100.063514)
- Friedrich, O., Uhlemann, C., Villaescusa-Navarro, F., et al. 2020, MNRAS, 498, 464, doi: [10.1093/mnras/staa2160](https://doi.org/10.1093/mnras/staa2160)
- Gillet, N., Mesinger, A., Greig, B., Liu, A., & Ucci, G. 2019, MNRAS, 484, 282, doi: [10.1093/mnras/stz010](https://doi.org/10.1093/mnras/stz010)
- Giri, U., & Smith, K. M. 2020, arXiv e-prints, arXiv:2010.07193. <https://arxiv.org/abs/2010.07193>
- Gualdi, D., Gil-Marín, H., & Verde, L. 2021, arXiv e-prints, arXiv:2104.03976. <https://arxiv.org/abs/2104.03976>
- Gualdi, D., Novell, S., Gil-Marín, H., & Verde, L. 2020, arXiv e-prints, arXiv:2009.02290. <https://arxiv.org/abs/2009.02290>
- Gupta, A., Matilla, J. M. Z., Hsu, D., & Haiman, Z. 2018, Phys. Rev. D, 97, 103515, doi: [10.1103/PhysRevD.97.103515](https://doi.org/10.1103/PhysRevD.97.103515)
- Hahn, C., & Villaescusa-Navarro, F. 2021a, J. Cosmology Astropart. Phys., 2021, 029, doi: [10.1088/1475-7516/2021/04/029](https://doi.org/10.1088/1475-7516/2021/04/029)
- . 2021b, J. Cosmology Astropart. Phys., 2021, 029, doi: [10.1088/1475-7516/2021/04/029](https://doi.org/10.1088/1475-7516/2021/04/029)
- Hahn, C., Villaescusa-Navarro, F., Castorina, E., & Scoccimarro, R. 2020, Journal of Cosmology and Astroparticle Physics, 2020, 040–040, doi: [10.1088/1475-7516/2020/03/040](https://doi.org/10.1088/1475-7516/2020/03/040)

- Hamilton, W. L. 2020, Synthesis Lectures on Artificial Intelligence and Machine Learning, 14, 1
- Harnois-Déraps, J., Martinet, N., Castro, T., et al. 2021, MNRAS, 506, 1623, doi: [10.1093/mnras/stab1623](https://doi.org/10.1093/mnras/stab1623)
- Harnois-Déraps, J., Martinet, N., & Reischke, R. 2022, MNRAS, 509, 3868, doi: [10.1093/mnras/stab3222](https://doi.org/10.1093/mnras/stab3222)
- Hassan, S., Andrianomena, S., & Doughty, C. 2020, MNRAS, 494, 5761, doi: [10.1093/mnras/staa1151](https://doi.org/10.1093/mnras/staa1151)
- He, S., Ravanbakhsh, S., & Ho, S. 2018, Analysis of Cosmic Microwave Background with Deep Learning, ICLR 2018. <https://openreview.net/forum?id=B15uoOyvz>
- Hopkins, P. F. 2015, MNRAS, 450, 53, doi: [10.1093/mnras/stv195](https://doi.org/10.1093/mnras/stv195)
- Hortua, H. J. 2021, arXiv e-prints, arXiv:2112.11865. <https://arxiv.org/abs/2112.11865>
- Illian, J. B., Penttinen, A., Stoyan, H., & Stoyan, D. 2008, Statistical Analysis and Modelling of Spatial Point Patterns (John Wiley & Sons, Ltd)
- Jeffrey, N., Alsing, J., & Lanusse, F. 2020, arXiv e-prints, arXiv:2009.08459. <https://arxiv.org/abs/2009.08459>
- Jeffrey, N., & Wandelt, B. D. 2020, arXiv e-prints, arXiv:2011.05991. <https://arxiv.org/abs/2011.05991>
- Kuruvilla, J. 2021, arXiv e-prints, arXiv:2109.13938. <https://arxiv.org/abs/2109.13938>
- Kuruvilla, J., & Aghanim, N. 2021, arXiv e-prints, arXiv:2102.06709. <https://arxiv.org/abs/2102.06709>
- Lazanu, A. 2021, J. Cosmology Astropart. Phys., 2021, 039, doi: [10.1088/1475-7516/2021/09/039](https://doi.org/10.1088/1475-7516/2021/09/039)
- Lee, J., & Ryu, S. 2020, arXiv e-prints, arXiv:2006.14477. <https://arxiv.org/abs/2006.14477>
- Lemos, P., Jeffrey, N., Cranmer, M., Ho, S., & Battaglia, P. 2022, arXiv e-prints, arXiv:2202.02306. <https://arxiv.org/abs/2202.02306>
- Li, H., Xu, Z., Taylor, G., Studer, C., & Goldstein, T. 2017, arXiv e-prints, arXiv:1712.09913. <https://arxiv.org/abs/1712.09913>
- Li, Y., Hu, W., & Takada, M. 2014, Phys. Rev. D, 89, 083519, doi: [10.1103/PhysRevD.89.083519](https://doi.org/10.1103/PhysRevD.89.083519)
- Li, Y., Schmittfull, M., & Seljak, U. 2018, J. Cosmology Astropart. Phys., 2018, 022, doi: [10.1088/1475-7516/2018/02/022](https://doi.org/10.1088/1475-7516/2018/02/022)
- Li, Z., Liu, J., Zorrilla Matilla, J. M., & Coulton, W. R. 2019, Phys. Rev. D, 99, 063527, doi: [10.1103/PhysRevD.99.063527](https://doi.org/10.1103/PhysRevD.99.063527)
- Liu, J., & Madhavacheril, M. S. 2019, Phys. Rev. D, 99, 083508, doi: [10.1103/PhysRevD.99.083508](https://doi.org/10.1103/PhysRevD.99.083508)
- Liu, W., Jiang, A., & Fang, W. 2022, arXiv e-prints, arXiv:2204.02945. <https://arxiv.org/abs/2204.02945>
- Lu, T., Haiman, Z., & Zorrilla Matilla, J. M. 2022, MNRAS, 511, 1518, doi: [10.1093/mnras/stac161](https://doi.org/10.1093/mnras/stac161)
- Makinen, T. L., Charnock, T., Alsing, J., & Wandelt, B. D. 2021, J. Cosmology Astropart. Phys., 2021, 049, doi: [10.1088/1475-7516/2021/11/049](https://doi.org/10.1088/1475-7516/2021/11/049)
- Marques, G. A., Liu, J., Zorrilla Matilla, J. M., et al. 2019, J. Cosmology Astropart. Phys., 2019, 019, doi: [10.1088/1475-7516/2019/06/019](https://doi.org/10.1088/1475-7516/2019/06/019)
- Massara, E., Villaescusa-Navarro, F., Ho, S., Dalal, N., & Spergel, D. N. 2021, Physical Review Letters, 126, doi: [10.1103/physrevlett.126.011301](https://doi.org/10.1103/physrevlett.126.011301)
- Naidoo, K., Massara, E., & Lahav, O. 2021, arXiv e-prints, arXiv:2111.12088. <https://arxiv.org/abs/2111.12088>
- Nelson, D., Springel, V., Pillepich, A., et al. 2019, Computational Astrophysics and Cosmology, 6, 2, doi: [10.1186/s40668-019-0028-x](https://doi.org/10.1186/s40668-019-0028-x)
- Neyman, J., & Scott, E. L. 1952, ApJ, 116, 144, doi: [10.1086/145599](https://doi.org/10.1086/145599)
- Nicola, A., Villaescusa-Navarro, F., Spergel, D. N., et al. 2022, arXiv e-prints, arXiv:2201.04142. <https://arxiv.org/abs/2201.04142>
- Ntampaka, M., Eisenstein, D. J., Yuan, S., & Garrison, L. H. 2020, ApJ, 889, 151, doi: [10.3847/1538-4357/ab5f5e](https://doi.org/10.3847/1538-4357/ab5f5e)
- Peebles, P. J. E. 1980, The large-scale structure of the universe (Princeton University Press, 1980. ISBN: 978-0-691-08240-0)
- Perez, L. A., Genel, S., Villaescusa-Navarro, F., et al. 2022, arXiv e-prints, arXiv:2204.02408. <https://arxiv.org/abs/2204.02408>
- Pillepich, A., Springel, V., Nelson, D., et al. 2018, MNRAS, 473, 4077, doi: [10.1093/mnras/stx2656](https://doi.org/10.1093/mnras/stx2656)
- Ravanbakhsh, S., Oliva, J., Fromenteau, S., et al. 2017, arXiv e-prints, arXiv:1711.02033. <https://arxiv.org/abs/1711.02033>
- Ribli, D., Pataki, B. Á., Zorrilla Matilla, J. M., et al. 2019, MNRAS, 490, 1843, doi: [10.1093/mnras/stz2610](https://doi.org/10.1093/mnras/stz2610)
- Ryu, S., & Lee, J. 2020, ApJ, 894, 65, doi: [10.3847/1538-4357/ab838d](https://doi.org/10.3847/1538-4357/ab838d)
- Samushia, L., Slepian, Z., & Villaescusa-Navarro, F. 2021, arXiv e-prints, arXiv:2102.01696. <https://arxiv.org/abs/2102.01696>
- Schmelzle, J., Lucchi, A., Kacprzak, T., et al. 2017, arXiv e-prints, arXiv:1707.05167. <https://arxiv.org/abs/1707.05167>
- Shao, H., Villaescusa-Navarro, F., Genel, S., et al. 2022, ApJ, 927, 85, doi: [10.3847/1538-4357/ac4d30](https://doi.org/10.3847/1538-4357/ac4d30)
- Simpson, F., Heavens, A. F., & Heymans, C. 2013, Physical Review D, 88, doi: [10.1103/physrevd.88.083510](https://doi.org/10.1103/physrevd.88.083510)
- Simpson, F., James, J. B., Heavens, A. F., & Heymans, C. 2011, Phys. Rev. Lett., 107, 271301, doi: [10.1103/PhysRevLett.107.271301](https://doi.org/10.1103/PhysRevLett.107.271301)

- Soneira, R. M., & Peebles, P. J. E. 1977, ApJ, 211, 1, doi: [10.1086/154898](https://doi.org/10.1086/154898)
- . 1978, AJ, 83, 845, doi: [10.1086/112268](https://doi.org/10.1086/112268)
- Springel, V. 2010, MNRAS, 401, 791, doi: [10.1111/j.1365-2966.2009.15715.x](https://doi.org/10.1111/j.1365-2966.2009.15715.x)
- Springel, V., White, S. D. M., Tormen, G., & Kauffmann, G. 2001, MNRAS, 328, 726, doi: [10.1046/j.1365-8711.2001.04912.x](https://doi.org/10.1046/j.1365-8711.2001.04912.x)
- Uhlemann, C., Friedrich, O., Villaescusa-Navarro, F., Banerjee, A., & Codis, S. 2020, MNRAS, 495, 4006, doi: [10.1093/mnras/staa1155](https://doi.org/10.1093/mnras/staa1155)
- Valogiannis, G., & Dvorkin, C. 2021, arXiv e-prints, arXiv:2108.07821. <https://arxiv.org/abs/2108.07821>
- Vicinanza, M., Cardone, V. F., Maoli, R., et al. 2019, Physical Review D, 99, doi: [10.1103/physrevd.99.043534](https://doi.org/10.1103/physrevd.99.043534)
- Villaescusa-Navarro, F. 2018, Pylians: Python libraries for the analysis of numerical simulations. <http://ascl.net/1811.008>
- Villaescusa-Navarro, F., Hahn, C., Massara, E., et al. 2020a, The Astrophysical Journal Supplement Series, 250, 2, doi: [10.3847/1538-4365/ab9d82](https://doi.org/10.3847/1538-4365/ab9d82)
- Villaescusa-Navarro, F., Anglés-Alcázar, D., Genel, S., et al. 2020b. <https://arxiv.org/abs/2010.00619>
- Villaescusa-Navarro, F., Anglés-Alcázar, D., Genel, S., et al. 2021a, arXiv e-prints, arXiv:2109.09747. <https://arxiv.org/abs/2109.09747>
- Villaescusa-Navarro, F., Genel, S., Angles-Alcazar, D., et al. 2021b, arXiv e-prints, arXiv:2109.10360. <https://arxiv.org/abs/2109.10360>
- Villaescusa-Navarro, F., Genel, S., Anglés-Alcázar, D., et al. 2022a, arXiv e-prints, arXiv:2201.01300. <https://arxiv.org/abs/2201.01300>
- Villaescusa-Navarro, F., Ding, J., Genel, S., et al. 2022b, arXiv e-prints, arXiv:2201.02202. <https://arxiv.org/abs/2201.02202>
- Villaescusa-Navarro, F., Genel, S., Anglés-Alcázar, D., et al. 2022c, ApJS, 259, 61, doi: [10.3847/1538-4365/ac5ab0](https://doi.org/10.3847/1538-4365/ac5ab0)
- Villanueva-Domingo, P. 2022, PabloVD/CosmoGraphNet, v1.0, Zenodo, doi: [10.5281/zenodo.6485804](https://doi.org/10.5281/zenodo.6485804)
- Villanueva-Domingo, P., & Villaescusa-Navarro, F. 2021, ApJ, 907, 44, doi: [10.3847/1538-4357/abd245](https://doi.org/10.3847/1538-4357/abd245)
- Villanueva-Domingo, P., Villaescusa-Navarro, F., Genel, S., et al. 2021, arXiv e-prints, arXiv:2111.14874. <https://arxiv.org/abs/2111.14874>
- Villanueva-Domingo, P., Villaescusa-Navarro, F., Anglés-Alcázar, D., et al. 2022, ApJ, 935, 30, doi: [10.3847/1538-4357/ac7aa3](https://doi.org/10.3847/1538-4357/ac7aa3)
- Villar, S., Hogg, D. W., Storey-Fisher, K., Yao, W., & Blum-Smith, B. 2021, arXiv e-prints, arXiv:2106.06610. <https://arxiv.org/abs/2106.06610>
- Wadekar, D., Thiele, L., Villaescusa-Navarro, F., et al. 2022, arXiv e-prints, arXiv:2201.01305. <https://arxiv.org/abs/2201.01305>
- Weinberger, R., Springel, V., & Pakmor, R. 2020, Astrophys. J. Suppl., 248, 32, doi: [10.3847/1538-4365/ab908c](https://doi.org/10.3847/1538-4365/ab908c)
- Weinberger, R., Springel, V., Hernquist, L., et al. 2017, MNRAS, 465, 3291, doi: [10.1093/mnras/stw2944](https://doi.org/10.1093/mnras/stw2944)
- Zaheer, M., Kottur, S., Ravanbakhsh, S., et al. 2017, arXiv e-prints, arXiv:1703.06114. <https://arxiv.org/abs/1703.06114>
- Zhang, G., Li, Z., Liu, J., et al. 2020, Phys. Rev. D, 102, 083537, doi: [10.1103/PhysRevD.102.083537](https://doi.org/10.1103/PhysRevD.102.083537)
- Zorrilla Matilla, J. M., Sharma, M., Hsu, D., & Haiman, Z. 2020, arXiv e-prints, arXiv:2007.06529. <https://arxiv.org/abs/2007.06529>