Monthly Notices of the ROYAL ASTRONOMICAL SOCIETY

MNRAS **516,** 2406–2419 (2022) https://doi.org/10.1093/mnras/stac2083
Advance Access publication 2022 August 2

Galaxies and haloes on graph neural networks: Deep generative modelling scalar and vector quantities for intrinsic alignment

Yesukhei Jagvaral [1,2]★ François Lanusse [3] Sukhdeep Singh,[1,2] Rachel Mandelbaum [1,2] Siamak Ravanbakhsh[4,5] and Duncan Campbell [1,6]

[1]*McWilliams Center for Cosmology, Department of Physics, Carnegie Mellon University, Pittsburgh, PA 15213, USA*
[2]*NSF AI Planning Institute for Data-Driven Discovery in Physics, Carnegie Mellon University, Pittsburgh, PA 15213, USA*
[3]*AIM, CEA, CNRS, Université Paris-Saclay, Université Paris Diderot, Sorbonne Paris Cité, F-91191 Gif-sur-Yvette, France*
[4]*School of Computer Science, McGill University, Montreal, QC H3A 0G4, Canada*
[5]*Mila, Quebec AI Institute , Montreal, QC H2S 3H1, Canada*
[6]*Epistemix Inc., Pittsburgh, PA 15203 USA*

Accepted 2022 July 18. Received 2022 July 13; in original form 2022 April 28

ABSTRACT

In order to prepare for the upcoming wide-field cosmological surveys, large simulations of the Universe with realistic galaxy populations are required. In particular, the tendency of galaxies to naturally align towards overdensities, an effect called intrinsic alignments (IA), can be a major source of systematics in the weak lensing analysis. As the details of galaxy formation and evolution relevant to IA cannot be simulated in practice on such volumes, we propose as an alternative a Deep Generative Model. This model is trained on the IllustrisTNG-100 simulation and is capable of sampling the orientations of a population of galaxies so as to recover the correct alignments. In our approach, we model the cosmic web as a set of graphs, where the graphs are constructed for each halo, and galaxy orientations as a signal on those graphs. The generative model is implemented on a Generative Adversarial Network architecture and uses specifically designed Graph-Convolutional Networks sensitive to the relative 3D positions of the vertices. Given (sub)halo masses and tidal fields, the model is able to learn and predict scalar features such as galaxy and dark matter subhalo shapes; and more importantly, vector features such as the 3D orientation of the major axis of the ellipsoid and the complex 2D ellipticities. For correlations of 3D orientations the model is in good quantitative agreement with the measured values from the simulation, except for at very small and transition scales. For correlations of 2D ellipticities, the model is in good quantitative agreement with the measured values from the simulation on all scales. Additionally, the model is able to capture the dependence of IA on mass, morphological type, and central/satellite type.

Key words: gravitational lensing: weak – methods: numerical – galaxies: statistics – galaxies: structure – cosmology: theory.

1 INTRODUCTION

Upcoming wide-field cosmological surveys such as the Vera C. Rubin Observatory Legacy Survey of Space and Time (LSST),[1] *Roman Space Telescope*[2] High Latitude Survey (HLS) and *Euclid*[3] will provide data that can be used to answer fundamental questions on the nature of dark energy through a precise measurement of cosmic shear, the observable correlations of galaxy shapes due to the minute but coherent deformations of distant galaxy images by the gravitational influence of massive structures along the line of sight (for a review, see Kilbinger 2015). One major astrophysical contaminant that arises when trying to measure this signal comes from the tendency of galaxies to naturally align with large-scale structure, an effect called intrinsic alignments (IA) that causes coherent shape distortions that anticorrelate with lensing shape distortions, and therefore bias the cosmological analysis if not accounted for (Troxel & Ishak 2015). High resolution hydrodynamical simulations, which can simulate the formation and evolution of individual galaxies, are valuable tools to study these alignments, but remain limited to small cosmological volumes due to their computational costs. In this work, we develop a deep generative model of 2D and 3D galaxy orientations that can capture the correct correlations of galaxy alignments, with a future goal of using this model to inpaint realistic galaxy alignments in much larger *N*-body simulations at very little cost.

In order to extract robust cosmological information from weak lensing surveys, accurate and precise models of survey data are needed. Cosmological simulations are often employed to produce synthetic survey data (mock catalogues) to facilitate the design and provide a test of analysis pipelines and systematics mitigation methods. Cosmological simulations have seen rapid advances in recent decades. Dark matter only (DM-only) *N*-body simulations such as the Outer Rim simulation achieved $\sim (4\,\mathrm{Gpc})^3$ scales with unprecedented resolutions exceeding 1 trillion particles (Heitmann et al. 2019). For DM-only simulations, galaxies must be included in post-processing; there are a variety of methods for doing so. One method, semi-analytical models (SAMs) of galaxy formation and

★ E-mail: yjagvara@andrew.cmu.edu
[1]https://www.lsst.org/
[2]https://roman.gsfc.nasa.gov/
[3]https://www.euclid-ec.org/

evolution, contain a number of free tunable parameters and can produce a synthetic galaxy population along with a variety of descriptive data (Somerville et al. 2008; Guo et al. 2011). Even though SAMs have been successful in reproducing some observed galaxy properties (Somerville & Davé 2015), in the coming era of precision cosmology their simple nature and challenges in matching galaxy populations across a range of redshifts leaves some uncertainty in the synthetic galaxy components of mock catalogues.

Another natural way to incorporate synthetic galaxies in mock catalogues is to directly implement them together with the dark matter with a hydrodynamic prescription and try to capture the full physics. In the past decade, cosmological hydrodynamical simulations such as MassiveBlack-II (Khandai et al. 2015), Illustris (Vogelsberger et al. 2014), EAGLE (Schaye et al. 2015), Horizon-AGN (Dubois et al. 2016), and IllustrisTNG (Nelson et al. 2019) have had some success in producing realistic galaxies with properties that match those of observed galaxies to some degree. IA has been studied within these simulations (e.g. Tenneti et al. 2014; Chisari et al. 2015; Velliscig et al. 2015; Tenneti, Mandelbaum & Di Matteo 2016; Samuroff, Mandelbaum & Blazek 2020) in order to accurately model and constrain IA models. Still, these types of simulations are based on resolution elements and along with the hydrodynamic equations, various astrophysical processes (such as AGN and stellar feedback; gas cooling; star formation) are based on effective sub-resolution models. Despite recent progress in computational astrophysics and cosmology, an ab initio cosmological simulation is far beyond current capabilities (Vogelsberger et al. 2020). However, as survey instruments become larger and more powerful, cosmological simulations have to encompass large volumes and high-resolution to keep up with the observational data. Currently, hydrodynamical simulations cannot reach the desired Gpc scale and resolution scale for upcoming surveys. Thus the best option to produce fast robust mock catalogues is to combine *N*-body simulations and some form of galaxy model, ideally non-parametric, as done for LSST DESC in Korytov et al. (2019) and for *Euclid*[4] (based on Potter, Stadel & Teyssier 2017).

It is now well established that galaxies form in DM haloes, and thus galaxy evolution is inevitably tied to the growth and evolution of each galaxy's parent DM halo. In the literature, this interconnectedness of galaxy and haloes is dubbed the galaxy–halo connection and is typically modelled as a multivariate distribution of various galaxy and halo properties. There has been some success in predicting and modelling properties such as mass, abundance, clustering, through the use of the halo occupation distribution (HOD) and subhalo abundance matching (SHAM; Somerville & Davé 2015). However, there is mounting evidence that the full galaxy–halo connection is highly non-linear and high dimensional (see e.g. Wechsler & Tinker 2018, for a comprehensive review). For example, when modelling the above-mentioned IA, the orientation of a galaxy is a 3D vector property (though we only observe the 2D projection of the shape) that correlates with its environment and the underlying large scale structure. The halo models of IA were developed to capture the small scale orientations within the parent DM halo (Schneider & Bridle 2010; Fortuna et al. 2021). For large scale alignments, the linear alignment model (Catelan, Kamionkowski & Blandford 2001; Hirata & Seljak 2004) and the later extensions that included non-linear contributions (Bridle & King 2007; Blazek et al. 2019), were successful in capturing the large-scale alignment of elliptical galaxies, despite underestimating the alignment at intermediate and small scales. For example, the tidal alignment-tidal torquing model from Blazek et al. (2019) showed some promise for describing alignments down to $\sim$1 Mpc h^{-1} scales in Samuroff et al. (2020); with this in mind, we would like our model to work to even smaller scales. However, these analytic models include a number of tunable parameters that are challenging to physically interpret and rely on assumptions that may not be robust.

In recent years, many fields of science are undergoing an Artificial Intelligence (AI) revolution and many AI models have been used in astrophysical and cosmological frameworks (see github.com/georgestein/ml-in-cosmology for a list and Ntampaka et al. 2019 for the role of ML in cosmology). In particular, unsupervised learning methods are designed to learn and detect patterns from the data itself, whereas traditional numeric and semi-analytic approaches are based on physical laws and models that are known beforehand. One class of unsupervised deep learning methods are deep generative models (DGM), where the DGM is trained on a given data set and learns the likelihood (explicitly or implicitly) that can be used to generate new sample data. In many cases, DGMs have been shown to outperform traditional numeric and semi-analytic models in accuracy or speed (Kodi Ramanah et al. 2020; Li et al. 2021). High-resolution hydro-sims mentioned above show promise for training and testing DGMs, given that we have access to the full 3D phase space data and numerous scalar features associated with galaxies and DM haloes. Thus, in order to enable fast production of mock catalogue for future surveys, in this work we will train a DGM on hydrodynamical simulations, to capture the relevant scalar and vector features.

In this work, we aim to capture the complex relation of the density field, DM halo, and the galaxy with a deep generative model, and then sample from this model various scalar and vector features of haloes and galaxies. Galaxies in real and mock catalogues are sparsely scattered through space with no fixed pattern or regular geometry in their distribution, and do not fit with conventional approaches of fully connected layers, convolutional neural networks, recurrent neural networks to represent the data as vectors, grids (tensors), or sequences (ordered sets), respectively.

Our approach is to model the cosmic web as a set of graphs, where the graphs are constructed for each halo. Graphs are a natural data structure to capture the correlations of galaxy properties amongst neighbours (Zhou et al. 2018), given that galaxies are distributed sparsely in the Universe. Graphs are defined as sets of *vertices* (also called *nodes*) and each of the *edges* (or *links*) connecting pairs of vertices. Given this structure, we adapt a Graph-Convolutional Network (Defferrard, Bresson & Vandergheynst 2016; Kipf & Welling 2016) to be sensitive to the 3D relative positions between vertices, a key ingredient to make the model aware of the Euclidean geometry of the problem beyond the graph connectivity. Using these layers, we subsequently implement a deep Generative Adversarial Network (GAN; Goodfellow et al. 2014) for signals on graphs. As part of this work, we will explore different network architectures and different physical information content from the simulations, and identify which ones enable robust reconstruction of the large-scale galaxy alignments.

This paper is organized as follows: We begin in Section 2 by describing the deep learning methods that were used, such as our graph convolutional neural network construction and the GAN architecture. Next, we begin Section 3 by describing the simulation suite we have used in Section 3.1, after which we introduce our estimators of the tidal field, galaxy shape, and two-point statistics. Next, in Section 4 we describe the models in detail. The results are presented in Section 5 and we conclude in Section 6.

[4]https://sci.esa.int/web/euclid/-/59348-euclid-flagship-mock-galaxy-catalogue

2 DEEP LEARNING BACKGROUND

In this section, we will introduce the graph convolutional networks we have implemented and then give a brief introduction to the generative adversarial network architecture. Our end goal is to model a conditional probability density $p(\boldsymbol{y}|\boldsymbol{x})$, where $\boldsymbol{y}$ is the desired property of a galaxy that we want to model (for example, shape or orientation), and $\boldsymbol{x}$ are the features used as input to the model, such as the (sub)halo mass, position, or the tidal field.

2.1 Graph convolutional networks

2.1.1 Spectral graph convolutions

In this work, we are considering undirected and connected graphs, which can be defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, where $\mathcal{V}$ is the set of graphs vertices, with $|\mathcal{V}| = n$ the number of vertices, $\mathcal{E}$ is the set of graph edges, and $\mathbf{W} \in \mathbb{R}^{n\times n}$ is the weighted adjacency matrix.

The normalized combinatorial graph Laplacian is defined as $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2}\,\mathbf{W}\,\mathbf{D}^{-1/2}$, where $\mathbf{D} \in \mathbb{R}^{n\times n}$ is the diagonal degree matrix with $D_{ii} = \sum_j W_{ij}$. Note that this operator is positive semidefinite and therefore admits an eigenvalue decomposition defined as:

$$\mathbf{L} = \mathbf{U}\Lambda\mathbf{U}^t, \tag{1}$$

where $\mathbf{U}$ is a unitary matrix and $\Lambda = \mathrm{diag}([\lambda_0, \lambda_1, \cdots, \lambda_n])$ are the eigenvalues of the operator. By analogy with a traditional Euclidean Laplacian, this transform is called a graph Fourier transform. The columns of $\mathbf{U}$ are called graph Fourier modes and Λ is the diagonal matrix of graph Fourier frequencies. For a given signal $f \in \mathbb{R}^n$, the graph Fourier transform of f is then defined as $\hat{f} = \mathbf{U}^t f$.

Given this harmonic transform, it becomes possible to define spectral filtering on graphs by defining a convolution product on graphs as a multiplication in Fourier space:

$$f \star g = \mathbf{U}\left((\mathbf{U}^t f) \odot (\mathbf{U}^t g)\right) = \mathbf{U}\left(\hat{f} \odot \hat{g}\right), \tag{2}$$

where $\odot$ is the Hadamard product. While this expression allows for convolution operations on graphs, it is a costly operation, as it first requires a decomposition of the graph Laplacian as well as dense matrix vector multiplications.

As an efficient alternative to a full spectral graph convolution, Defferrard et al. (2016) proposed to use parametric polynomial filters g_θ, of the form:

$$g_\theta(\mathbf{L}) = \sum_{k=0}^{K-1} \theta_k \mathbf{L}^k. \tag{3}$$

where $\theta \in \mathbb{R}^K$ is a vector of polynomial coefficients with K specifying the polynomial order. Defining a filter in terms of a polynomial function of the graph Laplacian has the advantage that the filter takes the same simple expression in Fourier space:

$$g_\theta(\mathbf{L}) = \sum_{k=0}^{K-1} \theta_k (\mathbf{U}\Lambda\mathbf{U}^t)^k = \sum_{k=0}^{K-1} \theta_k \mathbf{U}\Lambda^k\mathbf{U}^t = \mathbf{U}g_\theta(\Lambda)\mathbf{U}^t \tag{4}$$

A graph convolution with such a parametric filter can be defined as $g_\theta \star f = g_\theta(\mathbf{L})f = \mathbf{U}g_\theta(\Lambda)\mathbf{U}^t f$.

Given this formalism, we find it useful to introduce Chebyshev decompositions, which for a function f are given by

$$f(x) = \sum_{k=0}^{\infty} b_k T_k(x). \tag{5}$$

Here T_k is the Chebyshev polynomial of order k. An efficient implementation of these graph convolutions can therefore be achieved by the Chebyshev approximation. Chebyshev polynomials form an orthogonal basis of $L^2([-1, 1], dy/\sqrt{1-y^2})$ and can be computed recursively as $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$ with $T_0 = 1$, $T_1 = x$. One can therefore define a spectral graph convolution in terms of a finite order Chebyshev polynomial:

$$g_\theta = \sum_{k=0}^{K} \theta_k T_k(\mathbf{L}). \tag{6}$$

Limiting this expansion to $K = 1$ (Kipf & Welling 2016) yields the following expression:

$$g_\theta \simeq \theta_0 + 2\theta_1\mathbf{L} \tag{7}$$

Contrary to the full expression of the graph convolution by filter g_θ, we see that this first-order approximation no longer requires computing the graph Fourier Transform and can be computed by a single application of the graph adjacency matrix. In the rest of this work, we will be using this first-order application to parametrize graph convolutions, which can be implemented extremely efficiently by sparse matrix vector multiplication. We define one Graph Convolutional Network layer with an activation y_i for a node i as:

$$\forall i \in \mathcal{V},\ y_i = \mathbf{b} + \mathbf{W}_0 h_i + \sum_{j\in\mathcal{N}_i} w_{i,j}\mathbf{W}_1 h_j, \tag{8}$$

where $\mathbf{b}$ represents a vector of bias terms, we denote by $\mathcal{N}_i$ the set of immediate neighbours[5] of vertex i, $\mathbf{W}_0$ are the weights that apply a linear transform to the activation vector h_i of node i (i.e. self-connection), $w_{i,j}$ are linear transforms on the activation vectors h_j of the nodes j in the neighbourhood of i, and $\mathbf{W}_1$ are the set of weights that apply to the immediate neighbours. While the expressivity of a single GCN layer is quite limited, by stacking a large number of them, complex mappings on graphs can be represented.

2.1.2 Directional convolution kernels

The GCN introduced above uses the same isotropic convolution kernels for the entire graph. However, given the nature of our signal, we expect the 3D positions of neighbouring galaxies to be relevant to their alignments, for instance we know that within a halo, satellites tend to align towards the central galaxy (Pereira, Bryan & Gill 2008). We therefore want to design graph convolutions that have some sensitivity to 3D orientations.

Based on the dynamic convolution kernels introduced in Verma, Boyer & Verbeek (2017), we propose the following direction-dependent graph convolution layer:

$$y_i = \mathbf{b} + \mathbf{W}_0 h_i + \sum_{m=1}^{M} \frac{1}{|\mathcal{N}_i|} \sum_{j\in\mathcal{N}_i} q_m(\mathbf{r}_i, \mathbf{r}_j)\mathbf{W}_m h_j. \tag{9}$$

Here $|\mathcal{N}_i|$ denotes the cardinality of the set $\mathcal{N}_i$, M is the number of directions, and $\mathbf{r}$ are the 3D Cartesian coordinates of the node. The $q_m(\mathbf{r}_i, \mathbf{r}_j)$ are normalized so that $\sum_{m=1}^{M} q_m(\mathbf{r}_i, \mathbf{r}_j) = 1$ and are defined as:

$$q_m(\mathbf{r}_i, \mathbf{r}_j) \propto \exp(-\mathbf{d}_m^t \cdot (\mathbf{r}_j - \mathbf{r}_i))\ g_\lambda(\parallel \mathbf{r}_i - \mathbf{r}_j \parallel_2^2), \tag{10}$$

where the $\{\mathbf{d}_m\}_{m\in[1,M]}$ are a set of directions we want to make the kernel sensitive to, and g_λ is a parametric function of the distance between two vertices. With this parametrization, how a given vertex i receives contributions from its neighbours will be a function of the directions to the neighbours, as well as a function of the distance.

[5]Immediate neighbours or first neighbours are neighbours that are one hop away from node i.

In this work, we chose an exponential parametrization of the form: $g_\lambda(r) = \exp(-r^2/2\lambda^2)$, where λ is fit automatically during training. Finally, when modelling 3D alignments, we use a set of 26 directions on the unit sphere, equivalent to the 27 voxels of a 3D 3×3×3 convolution kernel, minus the central voxel which in our graph neural network will correspond to node self-connections.[6]

2.2 Fitting implicit distributions

The other aspect of the problem is learning how to model, and sample from, a conditional probability density $p(\boldsymbol{y}|\boldsymbol{x})$, where $\boldsymbol{y}$ might be a particular orientation of a galaxy, and $\boldsymbol{x}$ would be quantities such as the dark matter mass or the tidal field. In this section, we briefly introduce two conditional neural density estimators used in this work to model such $p(\boldsymbol{y}|\boldsymbol{x})$: mixture density networks (MDNs; Bishop 1994) and generative adversarial networks (GANs; Goodfellow et al. 2014). As we will see in the following sections, MDNs will be used to model low-dimensional densities, whereas GANs will be used to model complex joint densities of all galaxies in a halo. We will detail in Section 4.2.1 how these two models are combined in practice.

2.2.1 Low-dimensional conditional density fitting with mixture density networks

Mixture density networks are a class of feed-forward neural networks where the outputs are conditional distributions (the posterior probability distributions) instead of point estimates. Generally, MDNs are written as the weighted sum of n_c basis PDFs, where we chose truncated normal distributions as the basis functions:

$$p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta}) = \sum_{k=1}^{n_c} r_k(\boldsymbol{x};\boldsymbol{\theta})\,\mathcal{N}\left[\boldsymbol{y} \mid \boldsymbol{\mu}_k(\boldsymbol{x};\boldsymbol{\theta}), \boldsymbol{\Sigma}_k(\boldsymbol{x};\boldsymbol{\theta}), a, b\right], \tag{11}$$

where for a given likelihood $p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta})$, the $\boldsymbol{\theta}$ are all free parameters that are learned during the training, and the weights $\{r_k(\boldsymbol{x};\boldsymbol{\theta})\}$, means $\{\boldsymbol{\mu}_k(\boldsymbol{x};\boldsymbol{\theta})\}$, and covariance matrices $\{\boldsymbol{\Sigma}_k(\boldsymbol{x};\boldsymbol{\theta})\}$ are dependent on $\boldsymbol{x}$ through some neural network. Since we are interested in predicting vector quantities that can be represented as unit vectors, we used a truncated normal distribution with $a = -1, b = 1$. To train the MDN, the output at each training step is evaluated for the training data set and we take the loss function to be the negative log-likelihood:

$$L = -\log\left[\prod_i p(\boldsymbol{y}_i|\boldsymbol{x}_i;\boldsymbol{\theta})\right] \tag{12}$$

2.2.2 High dimensional conditional density fitting by adversarial training

We now turn to a different strategy for modelling conditional distributions $p(\boldsymbol{y}|\boldsymbol{x})$ which will become necessary to model the joint distribution of multiple properties of all galaxies in a given halo – in other words, $\boldsymbol{y}$ is high dimensional and an MDN will no longer be applicable. To address this problem, we propose to learn this conditional density by adversarial training.

Given a generating function $g_\theta(z, \boldsymbol{x})$ with $z \sim \mathcal{N}(0, \boldsymbol{I})$, we aim to adjust the implicit distribution generated by g_θ to match our target distribution $p(\boldsymbol{y}|\boldsymbol{x})$. This can be done by minimizing the Wasserstein 1-distance $\mathcal{W}$ between these two distributions to find an optimal set of weights $\theta_\star$. Unfortunately, we do not have access to $\mathcal{W}$ in closed form, so we must learn it as well. We introduce a second model, such that the expression inside the parenthesis in equation (13) approximates the Wasserstein distance, and which can be trained alongside the generator by solving the following minimax optimization problem:

$$\arg\min_\theta \left(\sup_\phi \mathbb{E}_{(x,y)}\left[d_\phi(\boldsymbol{x}, \boldsymbol{y}) - \mathbb{E}_z\left[d_\phi(g_\theta(\boldsymbol{z}, \boldsymbol{x}), \boldsymbol{y})\right]\right]\right). \tag{13}$$

This is the optimization problem that the WGAN (Arjovsky, Chintala & Bottou 2017) aims to solve, modified to account for the conditional variable $\boldsymbol{x}$. To ensure that d_ϕ indeed parametrizes a Wasserstein distance, one must ensure that its Lipschitz constant remains bounded. While several approaches have been proposed, from clipping the weights of the model (Arjovsky et al. 2017), to applying a spectral norm regularization (Miyato et al. 2018), in this work we adopt the gradient constraint (Gulrajani et al. 2017) for all applications, as we find it provides good results in practice.

3 THE SIMULATION AND ANALYSIS METHODS

Here we will explain the methods we have used to measure galaxy and DM haloes shapes, intrinsic alignment, two point statistics, and the methodology we used to decompose/classify galaxies dynamically.

3.1 Simulated data

In this work, we are using the TNG100-1 run from the IllustrisTNG simulation suite (for more information, please refer to Nelson et al. 2018; Marinacci et al. 2018; Naiman et al. 2018; Pillepich et al. 2018b; Springel et al. 2018; Nelson et al. 2019). The TNG100-1 is a hydrodynamical simulation with a box side length of 75 Mpc h^{-1}. The simulation uses the moving-mesh code AREPO (Springel 2010) and contains 2×1820^3 resolution elements with a gravitational softening length of 0.7 kpc h^{-1} for dark matter and star particles. Within the simulation the dark matter particle mass is 7.46×10^6 $M_\odot$ and the star particle masses are variable. Galaxy formation and evolution were modelled using radiative gas cooling and heating; star formation in the ISM; stellar evolution with metal enrichment from supernovae; stellar, AGN, and blackhole feedback; formation and accretion of supermassive blackholes (Pillepich et al. 2018a). The haloes within the simulation were catalogued using friends-of-friends (FoF) methods (Davis et al. 1985), and the subhaloes were catalogued using the SUBFIND algorithm (Springel et al. 2001). The simulation suite includes 100 snapshots at different redshifts, and we use the latest snapshot at $z = 0$ for our analysis.

We employ a minimum stellar mass threshold of $\log_{10}(M_*/M_\odot) = 9$ for all galaxies, using their stellar mass from the SUBFIND catalogue. Using the methods described in Jagvaral et al. (2021), we quantify the disc fractions of each galaxy based on dynamics (instead of fitting Sérsic profiles) and split the sample into two morphological bins: bulge-dominated (those with disc fraction lesser than 0.5) and disc-dominated (those with disc fraction greater than or equal to 0.5).

3.2 Tidal field

In order to predict galaxy and DM halo shapes and orientations, we use the tidal field defined as the Hessian of the gravitational potential ϕ:

$$T_{ij}(\mathbf{r}) = \frac{\partial^2 \phi(\mathbf{r})}{\partial r_i \partial r_j}. \tag{14}$$

[6] In simpler terms, for a cube consisting of 3×3×3 voxels, the center voxel will have $3^3 - 1 = 26$ connections; we subtract 1 because it accounts for the self-connection

To calculate the gravitational potential ϕ, we start by computing the matter overdensity field by constructing a particle mesh. First, the simulation box is divided into smaller 3D cubic cells. Within each cell c centred at position $\mathbf{r_c}$, we can count the total mass of the particles in that cell and divided it by the average across all cells, and write the overdensity field as:

$$\delta(\mathbf{r_c}) = \frac{\rho_c}{\langle \rho_c \rangle} - 1. \tag{15}$$

The gravitational potential is related to the overdensity field via the Poisson equation:

$$\nabla^2 \phi(\mathbf{r}) = 4\pi G \bar{\rho} \delta(\mathbf{r}), \tag{16}$$

where G is Newton's gravitational constant. The solution of the Poisson equation in Fourier space is:

$$\hat{\phi}(\mathbf{k}) = -4\pi G \bar{\rho} \frac{\hat{\delta}(\mathbf{k})}{k^2}. \tag{17}$$

Plugging this result back into the Fourier transform of equation (14), we obtain the tidal tensor

$$\hat{T}_{ij}(\mathbf{k}) = 4\pi G \bar{\rho} \frac{k_i k_j}{k^2} \hat{\delta}(\mathbf{k}). \tag{18}$$

In order to smooth the small scale coarseness of the tidal field (caused by the discrete resolution elements of the simulation), we introduce a Gaussian filter with smoothing scale γ:

$$\hat{T}_{ij}(\mathbf{k}) = 4\pi G \bar{\rho} \frac{k_i k_j}{k^2} \hat{\delta}(\mathbf{k}) \;\; e^{-k^2\gamma^2/2}. \tag{19}$$

Finally, the Fourier-space tidal field from equation (19) can be converted into real space using the inverse Fourier transform. The tidal field was evaluated at the position of each galaxy, using a cloud-in-cell window kernel to interpolate between the centres of the grid points, with various values of γ: 0.1, 0.25, 0.5, 1, and 2 Mpc h^{-1} on a mesh of size 1024^3, with cell sizes given as $L_{\rm box}/1024 = 0.073$ Mpc h^{-1}.

3.3 Shapes of haloes and galaxies

To measure the shapes of galaxies and DM haloes, we utilize the mass quadrupole moments (often incorrectly referred to as the inertia tensor). We use the simple quadrupole moments I_{ij}, defined as

$$I_{ij} = \frac{\sum_n m_n r_{ni} r_{nj}}{\sum_n m_n}. \tag{20}$$

Here the summation index n runs over all particles of a given type in a given galaxy, where m_n is the mass of the nth particle and r is the distance between the galaxy or subhalo centre of mass and the nth particle, with i and j indexing the three spatial directions. We chose to use the simple mass quadrupole moment, since the reduced and the reduced iterative moments generally give very low alignment signals (Jagvaral, Singh & Mandelbaum 2022).

The three unit eigenvectors of the mass quadrupole moment, defined as $\mathbf{s}_\mu = \{s_{x,\mu}, s_{y,\mu}, s_{z,\mu}\}^\tau$ and $\mu \in \{a, b, c\}$, The half-lengths of the principal axes of the ellipsoid are given by $a \propto \sqrt{\omega_a}, b \propto \sqrt{\omega_b}$, and $c \propto \sqrt{\omega_c}$, such that $a \geq b \geq c$ and $\omega_a, \omega_b, \omega_c$ are the eigenvalues of the mass quadrupole moment.

To compute the projected alignment signals, we need to use the 3D mass quadrupole moments to define 2D projected shapes. Following Joachimi et al. (2013), we can obtain the projected 2D ellipse by solving $\boldsymbol{r}^\tau \mathbf{W}^{-1} \boldsymbol{r} = 1$, where

$$\mathbf{W}^{-1} = \sum_{\mu=1}^{3} \frac{\boldsymbol{s}_{\perp,\mu} \mathbf{s}^\tau_{\perp,\mu}}{\omega_\mu^2} - \frac{\boldsymbol{k}\boldsymbol{k}^\tau}{\alpha^2}, \tag{21}$$

and

$$\mathbf{k} = \sum_{\mu=1}^{3} \frac{s_{\parallel,\mu} \boldsymbol{s}_{\perp,\mu}}{\omega_\mu^2} \quad \text{and} \quad \alpha^2 = \sum_{\mu=1}^{3} \left(\frac{s_{\parallel,\mu}}{\omega_\mu} \right)^2. \tag{22}$$

Here, $\mathbf{s}_{\perp,\mu} = \{s_{x,\mu}, s_{y,\mu}\}^\tau$ are the eigenvectors projected along the projection axis (for which we arbitrarily choose the z-axis of the 3D simulation box). Here, we note that W and k in these equations are not the same quantities as the previously defined W's and k's in Sections 2 and 3.

Then, the two components of the galaxy ellipticity can be expressed in terms of the symmetric tensor $\mathbf{W}$

$$(e_1, e_2) = \frac{(W_{xx} - W_{yy}, 2W_{xy})}{W_{xx} + W_{yy} + 2\sqrt{\det \mathbf{W}}}. \tag{23}$$

For the special case that the s_c (smallest) axis lies perfectly along the projection axis, the absolute value of the ellipticity is $|e| = (a - b)/(a + b)$. In terms of the projected simulation box, the x, y directions correspond to the positive and negative direction of e_1 (since we projected along the z direction).

3.4 Two-point estimators

In this section, we will describe the two-point correlation functions that will be used to quantify IA.[7] The ellipticity-direction (ED) correlation captures the position and the orientation correlation angles in 3D which are useful in comparing alignments in simulations. On the other hand, the projected density-shape correlation function (w_{g+}) captures the correlation between overdensity and projected intrinsic ellipticity, as is commonly used in observational studies.

3.4.1 Orientation correlation functions in 3D

The ellipticity-direction (ED) correlation function is defined as (Lee et al. 2008):

$$\omega(r) = \langle |\hat{e}(\mathbf{x}) \cdot \hat{r}(\mathbf{x})|^2 \rangle - \frac{1}{3} = \omega^{1h}(r) + \omega^{2h}(r) \tag{24}$$

for a subhalo/galaxy at position $\mathbf{x}$ with major axis direction $\hat{e}$ and the unit vector $\hat{r}$ denoting the direction of a density tracer at a distance r. As shown, in simulations this correlation function can be decomposed into 1-halo and 2-halo terms, where the 1-halo term captures the correlation among subhaloes within the same halo, and the 2-halo term captures contributions from pairs of subhaloes belonging to different haloes. The estimator we use to compute it in the simulations is as follows:

$$\omega(r) = \sum_{i \neq j} |\hat{e}_i \cdot \hat{r}_{ij}|^2 - \frac{1}{3} \tag{25}$$

3.4.2 Density-shape correlation functions in 2D

The cross-correlation function of galaxy positions (as tracers of the galaxy overdensities) and intrinsic ellipticities is defined as:

$$\xi_{g+}(\mathbf{r}) = \langle \delta_g(\mathbf{x}) \delta_+(\mathbf{x} + \mathbf{r}) \rangle, \tag{26}$$

where $\delta_g(r)$ and $\delta_+(r)$ represent the galaxy overdensity field and the intrinsic shape field, respectively. It can be estimated using the

[7] All of the two-point statistic were measured using the HALOTOOLS package v0.7 (Hearin et al. 2017) and the supporting halotools_ia package.

method described in Mandelbaum et al. (2011) as a function of r_p and Π:

$$\xi_{g+}(r_p, \Pi) = \frac{S_+D - S_+R}{RR}. \quad (27)$$

Here, RR are counts of random–random pairs binned based on their perpendicular and line-of-sight separation;

$$S_+D \equiv \frac{1}{2}\sum_{\alpha\neq\beta} e_+(\beta|\alpha), \quad (28)$$

represent the shape correlations, where $e_+(\beta|\alpha)$ is the + component of the ellipticity of galaxy β (from the shape sample) measured relative to the direction of galaxy α (from the density tracer sample). S_+R is defined in an equivalent way, but instead of using galaxy positions we use randomly distributed positions.

The projected two-point correlation functions can be obtained by integrating over the third dimension, with the integral approximated as sums over the line-of-sight separation (Π) bins:

$$w_{g+}(r_p) = \sum_{-\Pi_{max}}^{\Pi_{max}} \Delta\Pi\, \xi_{g+}(r_p, \Pi), \quad (29)$$

where we chose a Π_{max} value of 20 Mpc h^{-1}, following Jagvaral et al. (2022).

4 NEURAL INTRINSIC ALIGNMENT MODEL

Having introduced the fundamental machine learning concepts necessary to build our model, as well as the necessary background on simulation data and intrinsic alignments, we now introduce our proposed model.

4.1 Graph construction

To construct the graph for the cosmic web (i.e. for the subhaloes and the galaxies), we first grouped all of the subhaloes and galaxies based on their parent halo. In other words, we grouped subhaloes and galaxies based on their group membership ID from the halo finder. There exist a few choices for modelling proximity graph relations between the members in a group, such as the Gabriel graph (which is a subset of the Delaunay triangulation) and the radius nearest neighbour graph (r-NNG). These different graph types differ by their connectivity, i.e. they have different adjacency matrices (Mathieson & Moscato 2019). In this study we employ the r-NNG to model the connectivity of our graphs.

Given a galaxy catalogue, an undirected graph based on the 3D positions is built by placing each galaxy on a *graph node*. Then, each node will have a list of features such as halo mass, subhalo mass, central versus satellite identification (binary column), and tidal fields smoothed on several scales. Then for a given group (i.e. within a halo) the graphs are connected. To build the graph connection, the nearest neighbours within a specified radius for a given node are connected via the *undirected edges* with *signals* on the graphs representing the alignments.

4.2 Model architecture

In Fig. 1 we outline the architecture of our models. The general idea is that we have list of features (orange box) that are relevant for capturing the dependence of intrinsic alignments within a halo (dashed red box), and the tidal fields that are relevant for capturing the dependence of IA for galaxies on matter beyond their halo (dashed purple). Then, these inputs are fed into the GAN-Generator (crimson box), which tries to learn the desired output labels (yellow box). At the end the input and the output from the GAN-Generator are fed into the GAN-Critic (blue box) to determine the performance of the GAN-Generator. In our model, the Generator has five layers each with {128,128,16,2,2} neurons, while the Critic has four layers each with {128,128,64,32} neurons followed by a mean-pooling layer and a single output neuron. For scalars, this model works as is.

However, for predicting vector quantities, we made slight modifications to the network architecture. For the 2D model, the input tidal fields are all in 3D, but the GAN-Generator outputs a 2D vector, while the inputs of the GAN-Critic get projected on to 2D. For 3D vector quantities (orientations), we had to slightly modify our architecture as we will explain below.

4.2.1 3D orientation modelling

Since satellite alignments are challenging to model, we rely on the *graph* structure to capture this alignment, whereas for the centrals we use a simple MDN. The structure of the network in this case is shown in the '3D vector' box within Fig. 1. We decided to take a 'divide-and-conquer' approach where we broke down the problem as follows:

$$p(\boldsymbol{y}|\boldsymbol{x}) = p(\boldsymbol{y}_{cen}|\boldsymbol{x}_{tid})\, p(\boldsymbol{y}_{sat}|\boldsymbol{y}_{cen}, \boldsymbol{x}_{sat}), \quad (30)$$

where $p(\boldsymbol{y}|\boldsymbol{x})$ represents the conditional probability of the 3D orientations of all galaxies given all input features; $p(\boldsymbol{y}_{cen}|\boldsymbol{x}_{tid})$ is the 3D orientation of central galaxies given the tidal fields; and $p(\boldsymbol{y}_{sat}|\boldsymbol{y}_{cen}, \boldsymbol{x}_{sat})$ is the 3D orientation of satellite galaxies given the features of satellite galaxies and orientations of central galaxies. $p(\boldsymbol{y}_{cen}|\boldsymbol{x}_{tid})$ is modelled using the tidal fields with MDN as shown in the transparent green box in the second panel of Fig. 1. Next, the outputs from the MDN are fed into the GAN, together with the other features, as listed in the red dashed box (we feed the output from the MDN to the GAN in order to capture correlations between the centrals and satellites). Since the satellite alignments appeared to be modelled effectively without using the tidal fields as an input feature, we did not include the tidal fields in the input for the GAN. Note that we initially tried training the GAN to capture both central and satellite alignments, but it was underpredicting the alignments for centrals, which motivated the approach described above.

4.3 Training

We train the model using the Adam optimizer (Kingma & Ba 2014) with a learning rate of 10^{-3} and exponential decay rates of $\beta_1 = 0$ and $\beta_2 = 0.95$. During the adversarial training we train the Generator for five steps and the Critic for one step with a batch size of 64 (one batch is set of graphs) and a leaky ReLU activation function. Due to the scarcity of high mass haloes, we balance probabilities of graphs based on group mass when processing batches by downsampling haloes of low mass and up-sampling haloes of high mass with replacement. As a result, haloes with high mass are reused in multiple different batches in the training.

In order to avoid overfitting, during each training epoch we augment the data by applying random rotations to the batches of the graphs. As a test for overfitting, we also made roughly a 50/50 train-and-test sample split, while still maintaining group membership. The results of this test are in Appendix A; we see no significant signs of overfitting. Given these findings and the limited simulation data available, in the following presentation of results, we used all of the sample to generate the output.

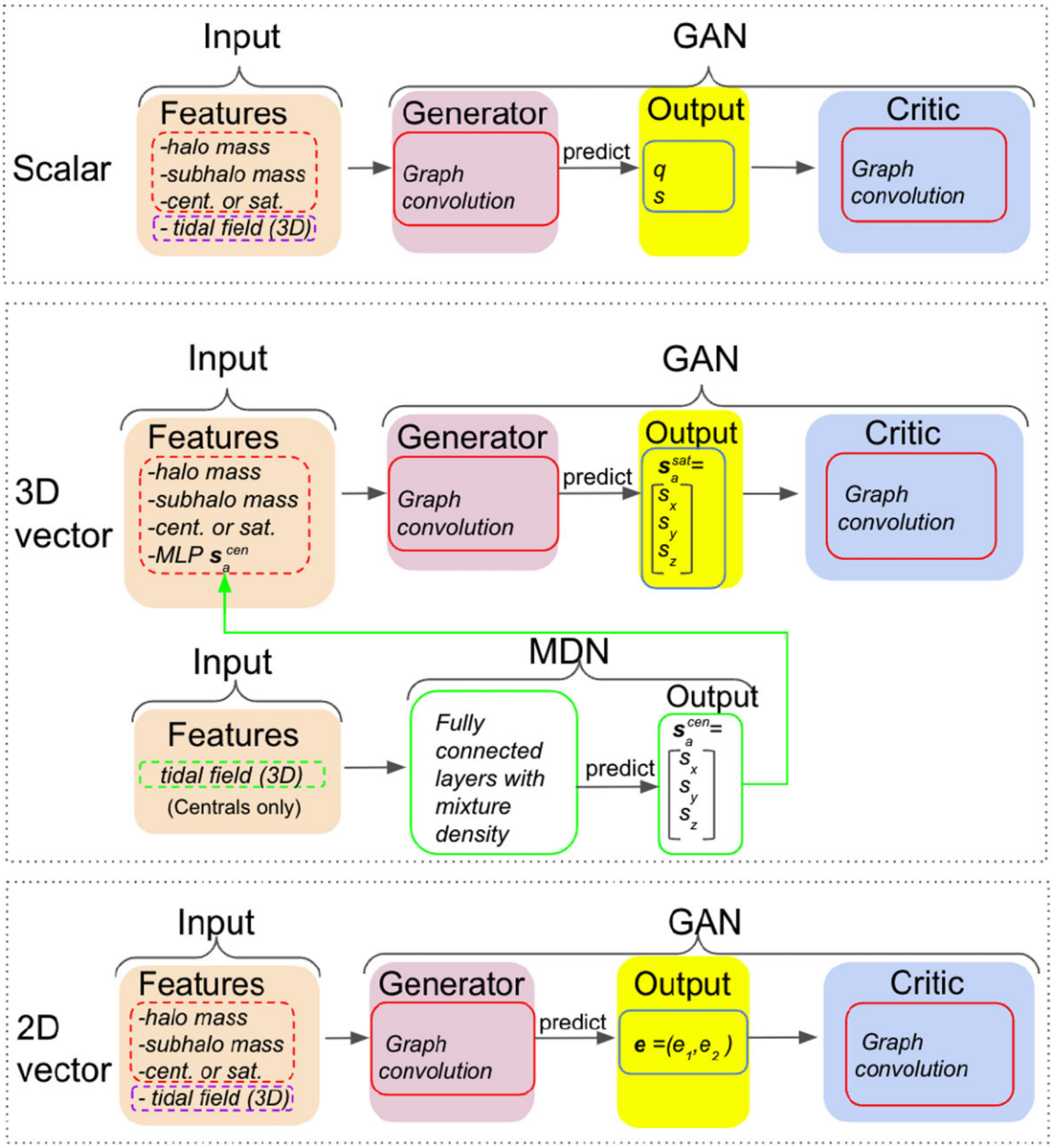

Figure 1. Architecture of the Graph convolution GAN models used.

As is common with GANs, our GAN models do not converge; we had to arbitrarily stop the training once it reached a reasonable result. For example, for the 2D model the results start to look reasonable around training step 40 000, which takes about 3–4 d on a single NVIDIA-A100 GPU. Our code is available at https://github.com/melon-lemon/GraphGAN.

5 RESULTS

In this section, we will first describe the results when using the GAN to predict scalar quantities, then 3D orientation correlation functions, and finally the 2D (projected) intrinsic alignment correlations.

Throughout the section we will refer to the sample generated from the Graph-Convolutional Network-based Generative Adversarial Networks as the *GAN* sample, and the samples from the TNG100 simulation as the *TNG* sample. In all of the below, we always trained and generated on the full sample, and tests on the subsamples were used to demonstrate the ability of the models to distinguish the variation in IA across the subpopulations. Since past work (e.g. Jagvaral et al. 2022) has shown that alignments depend primarily on mass, satellite/central status, and morphology, we explicitly check that the GAN can capture these dependencies.

5.1 Predictions of scalar quantities: shapes

In this subsection, we present the shapes of DM subhaloes and galaxies. Shapes are important scalar quantities that is used in intrinsic alignment studies. The DM subhalo and galaxy shapes were generated separately. The model for predicting DM subhalo shapes serves a dual purpose: it can be used as a sanity check and used to predict shapes of subhaloes that are not well resolved.

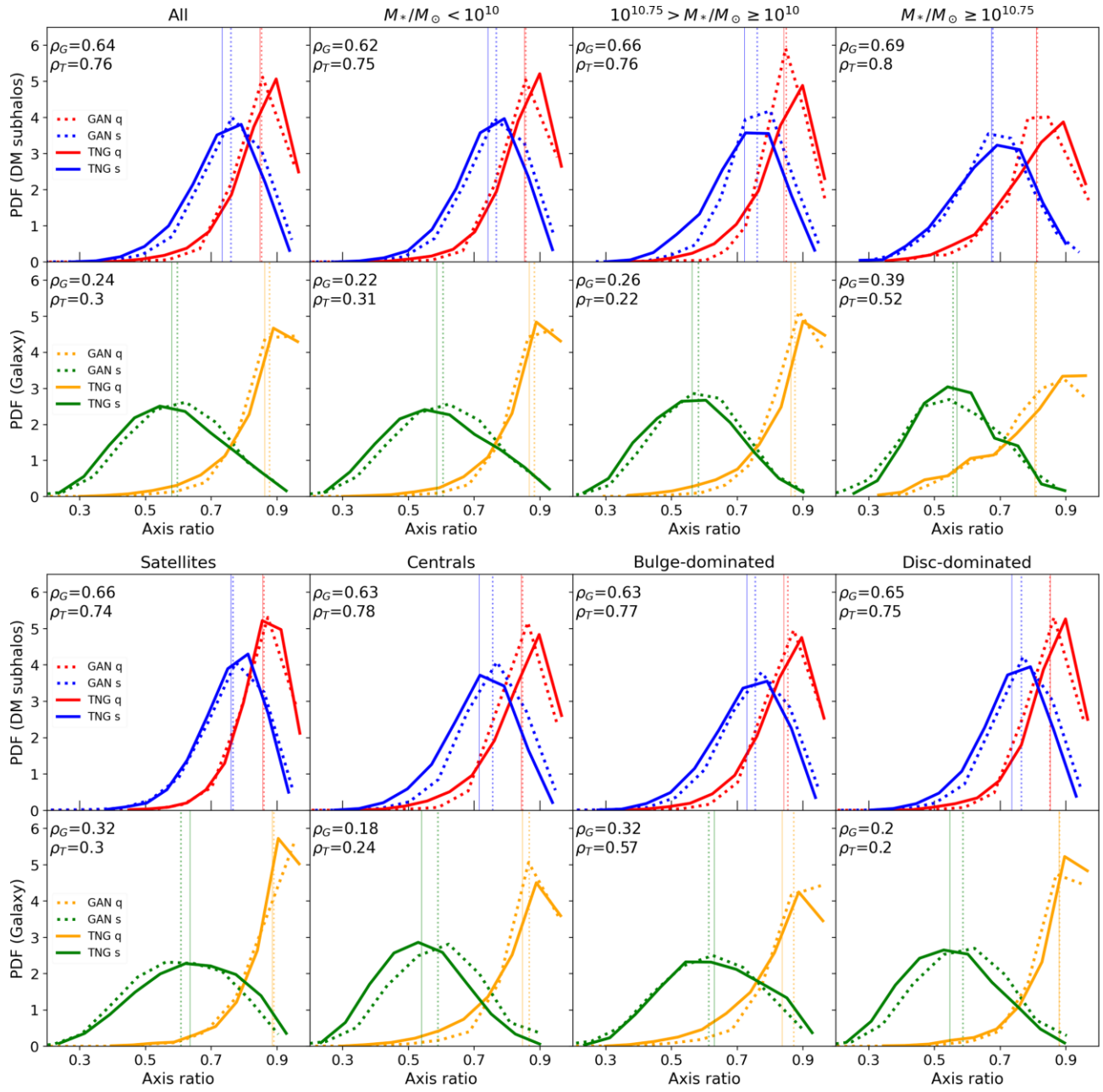

Figure 2. Distribution of shapes of DM subhaloes (top row) and galaxies (bottom row), quantified using the intermediate-to-major axial ratio q and the minor-to-major axial ratio s for the galaxy or subhalo modelled as a 3D ellipsoid. We show results for the full sample, and for subsamples determined based on stellar mass, satellite/central identification, and morphological classification. The solid lines indicate the sample from the TNG simulation and the dotted lines indicate the sample generated by the GAN. The parameters $\rho_G = \rho(\mathrm{GAN}\,q, \mathrm{GAN}\,s)$ and $\rho_T = \rho(\mathrm{TNG}\,q, \mathrm{TNG}\,s)$ in each panel indicate the Pearson-r correlation between the two shape parameters. The faint vertical lines indicate the mean of the distribution with the corresponding style and colour. Qualitatively, the GAN reproduces the shape of the distributions for the whole sample and the various subsamples. Quantitatively, the GAN captures the mean and the spread of the distribution well. However, the GAN typically underestimates the correlation between q and s by about 5–20 per cent for a given subsample. Note: for DM subhaloes we used the corresponding galaxy masses for the binning.

In Fig. 2 we compare the histograms of galaxy and DM subhalo axis ratios, defined as $q = b/a$ (intermediate-to-major axial ratio) and $s = c/a$ (minor-to-major axial ratio). Overall, the GAN captures and reproduces the distributions of the two axis ratios to a good degree, with the means of distributions agreeing within a few per cent. We also measured the correlation between the two shape parameters using the *Pearson-r* coefficients, which are displayed in each panel as ρ. These values tell us to what degree the GAN captures the distribution of 3D shapes from TNG (since the 3D shape distribution depends on correlations between q and s within the population). For the DM subhalo shapes, the GAN underestimates the correlation between the two shape parameters by about 10–15 per cent for all the subsamples. In contrast, for the galaxy shapes, the GAN underestimates the correlation between the two shape parameters by about 5–20 per cent for all subsamples, except for the satellites where it is within 2 per cent of the target correlation. All in all,

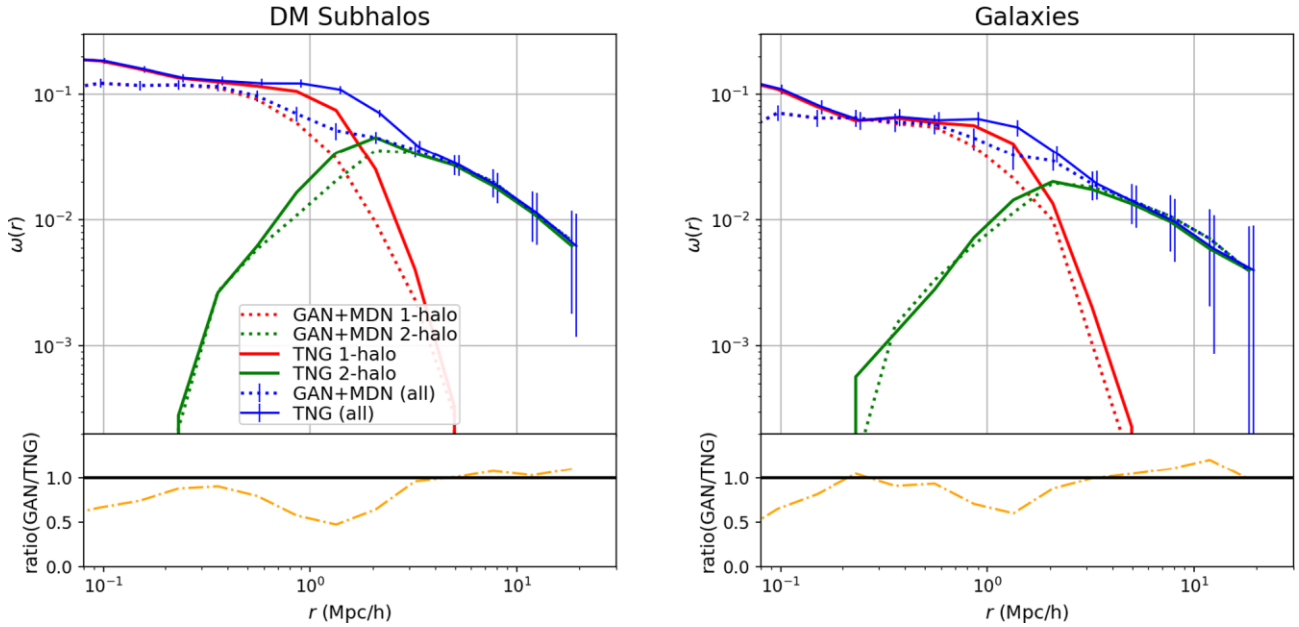

Figure 3. ED correlation function, $\omega(r)$, of the 3D major axis with galaxy positions: the solid lines show the measured values from the TNG simulation, while the dashed lines show the generated values from the GAN+MDN. The top panels show $\omega(r)$ decomposed into 1-halo (red line) and 2-halo (green line) terms along with the total (blue line), and the bottom panel shows the ratio of the total $\omega(r)$ from the GAN+MDN to that measured in TNG. The panel on the left-hand side is for DM subhaloes, whereas the right-hand panel is for galaxies. The generated values agree well with measured values on most scales, except on very small scales ($\lesssim$0.1 Mpc h^{-1}) and the 1- to 2-halo transition region ($\sim$1 to 2 Mpc h^{-1}). The GAN+MDN curve was shifted by 5 per cent to the left for visual clarity.

the distributions of shapes generated by the GAN are in good quantitative agreement with the target distributions, though slightly underestimating the correlation between the two shape parameters.

In contrast, the correlation defined as $\rho(\mathrm{GAN}\,q, \mathrm{TNG}\,q)$ – and likewise for s – is less informative, because we care about the population statistics, and not accurate predictions for individual subhaloes/galaxies. Indeed, a high value of this correlation coefficient would be a bad sign, as it might imply that the model has 'memorized' the specific shapes of galaxies/subhaloes in TNG, rather than learning about the distribution of shapes for the ensemble. We find that the measured $\rho(\mathrm{GAN}\,q, \mathrm{TNG}\,q)$ and $\rho(\mathrm{GAN}\,s, \mathrm{TNG}\,s)$ were below 15 per cent for all subsamples. These numbers are not shown on the figure.

5.2 Predictions of vector quantities

5.2.1 Predictions of ED correlation functions

Here, we present the results of the correlation functions of the galaxy major axis directions with galaxy position in 3D. In Fig. 3, we plot the ED correlation function for DM subhaloes and galaxies for the whole population generated from the GAN. The error bars were obtained using jackknife resampling. Here we remind the reader that the alignments of satellites were learned using the Graph-structure and the alignments of centrals were learned using a simple MDN, as described in Section 2.2.1. Also, the output from the MDN was given as an input to the Graph-structure so as to capture any correlation between the central and satellite terms. The satellite alignments are generally more challenging to model, since they are strongly affected by physics on non-linear distance scales with many competing physical processes that shape the alignment (Kiessling et al. 2015), and the Graph-structure learns and captures the satellite alignment correlations using only mass for both the DM subhalo and the galaxies. The alignment of centrals is easier to model, so for simplicity and efficiency we learned it using a simple MDN regressor. The generated alignment correlations provide a reasonable match to the simulation on many scales. The exception is that it underestimates the correlation (by a factor of 0.5) on $\lesssim$0.1 Mpc h^{-1} scales, and in the 1- to 2-halo transition region, $\sim$1–2 Mpc h^{-1} .

Next, in Fig. 4, we present similar quantities as in Fig. 3 for subsamples based on mass. For this purpose, we use mass bins of $M_*/\mathrm{M}_\odot < 10^{10}$, $10^{10} \leq M_*/\mathrm{M}_\odot \leq 10^{10.75}$, $M_*/\mathrm{M}_\odot > 10^{10.75}$; for DM subhaloes, we used the corresponding galaxy masses for the binning. For the low and intermediate mass bins (the first two columns), the GAN-predicted curve follows a very similar pattern as in Fig. 3, with good agreement on most scales. However, for the high mass bin (third column), the GAN underestimates the alignment correlation function by about a factor of two on all scales, for both DM subhaloes and galaxies. This underperformance may be explained by the small number of high mass galaxies in our sample, which may have caused the neural network to be undertrained for this mass bin. A similar effect has been observed in Ho et al. (2019) where the support density machine was also underperforming for high mass haloes.

5.2.2 Predictions of w_{g+} correlation functions

Moving to 2D shapes and alignments, in Fig. 5 we present the distribution of 2D complex ellipticities from the TNG simulation and the GAN. The GAN produces distributions of ellipticities that agree well quantitatively with the ones measured from the TNG simulation, with means of -0.02, 0.00, -0.01, 0.00 and standard deviations of 0.17, 0.15, 0.14, 0.14 for GAN e_1, GAN e_2, TNG e_1, TNG e_2, respectively.

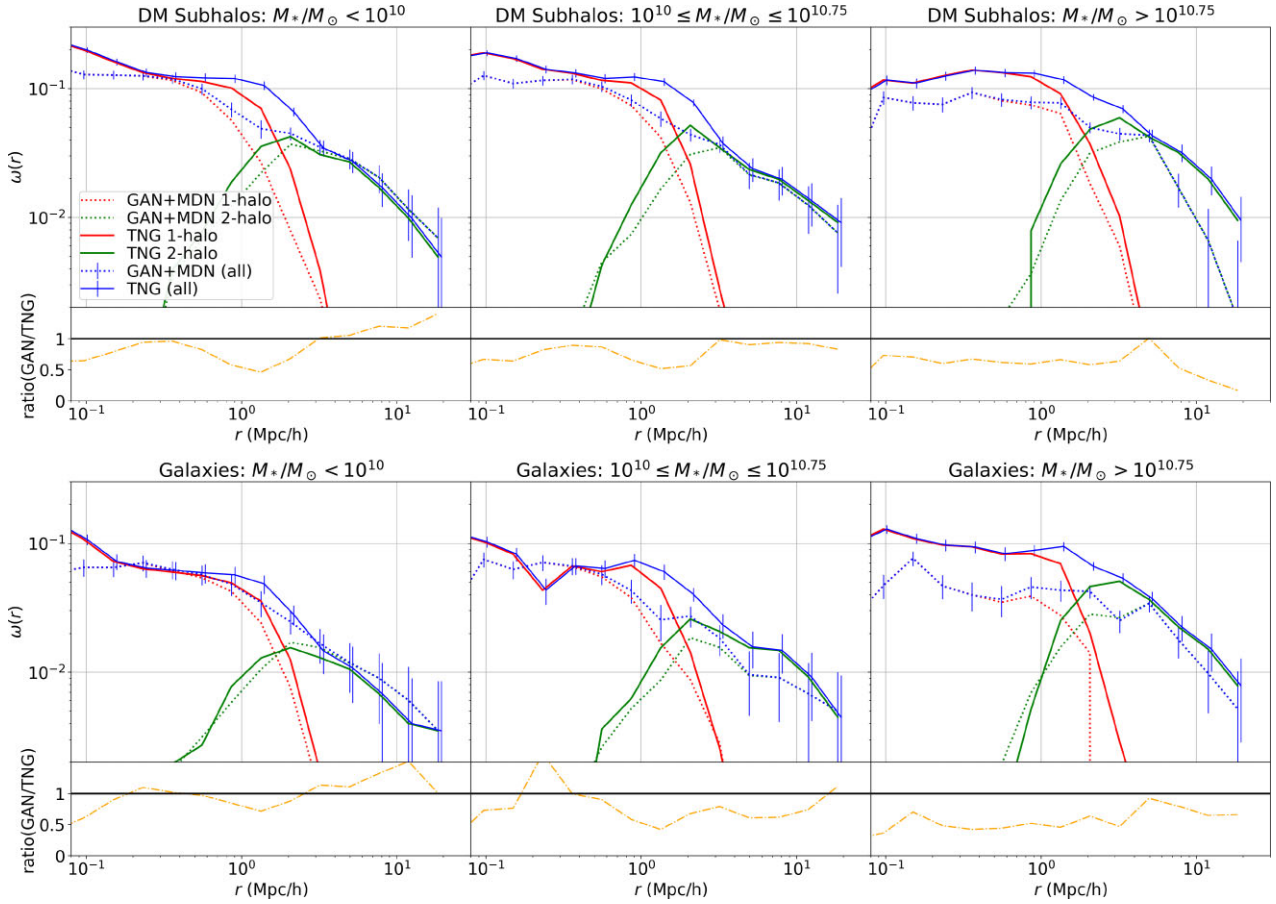

Figure 4. Same as Fig. 3, but after dividing the sample into mass bins as indicated on top of each figure. For both DM subhaloes and galaxies, for the two lower mass bins, the agreement between the generated model and TNG is good on most scales and follows the same trend as the whole population (see Fig. 3). However, for the highest mass subsample, for both DM subhaloes and galaxies, the generated values from the GAN+MDN underestimate the alignment correlation function by a factor of $\sim$1/2 on most scales. This may be due to the fact that there are far fewer high mass galaxies than low mass galaxies in our sample, resulting in an insufficient training sample for that mass range. Note: for DM subhaloes we used the corresponding galaxy masses for the binning.

Next, we examine w_{g+}, the density–shape correlation function computed using the ellipticities, as shown in Fig. 6. Compared with the 3D correlation function, $\omega(r)$, the projected 2D correlation function, w_{g+}, from the GAN agrees quantitatively with the measured one from TNG simulation. Here, the errorbars were derived from an analytic estimate of the covariance matrix, which includes Gaussian terms for noise and cosmic variance (for more details see Singh et al. 2017; Samuroff et al. 2020). We explore the robustness of the models and their stochastic uncertainties in Appendices A and B, respectively. These results show no sign of overfitting, and the stochastic uncertainty of the model is substantially smaller than the analytic errors representing astrophysical sources of statistical uncertainty (shape noise and cosmic variance).

In the second column of Fig. 6, we examine the mass dependence of the w_{g+} curve from the GAN. Again, here the agreement between the GAN and the measured correlation functions in TNG are quite good for all three mass bins. Still, on scales around 0.1–0.5 Mpc h^{-1}, the GAN overestimates w_{g+} for the low and intermediate mass bins, while underestimating w_{g+} for the high mass bin by about a factor of 2. This again may be due to the fact that there are far fewer high mass galaxies to train on for the GAN.

One thing to note here is that this plot shows the GAN model trained on the total mass of the subhaloes, instead of separate columns for DM, gas, and stellar mass. However, when we give the GAN DM, gas, and stellar mass information, the predicted mass dependence of the intrinsic alignments improves significantly, with disagreement only on the very smallest scales. We none the less chose to show the GAN model trained on total mass, since we plan to deploy this model on *N*-body simulations where that is the only information available.

Next, in the third column of Fig. 6, we explore whether the GAN captures the alignment dependence on galaxy morphology. Interestingly, the GAN is able to distinguish between morphologically selected samples, even though it was never given this information explicitly. As expected, the bulge-dominated sample shows higher alignment signal in both TNG and GAN, with good quantitative agreement. Similarly as expected, the disc-dominated population shows lower alignment, again with good quantitative agreement. Note that the bulge-dominated sample was controlled for mass effects (i.e. this sample was weighted to match the mass distribution of the disc-dominated sample), as explained in Jagvaral et al. (2022). Therefore the implicit mass differences between bulges and discs have already been accounted for in this comparison. We hypothesize that the network may be able to distinguish between morphological types based on the environments through the tidal fields, since galaxy morphology tend to depend on the environment (Tempel et al. 2011). We note that this morphological split is not available for N-body

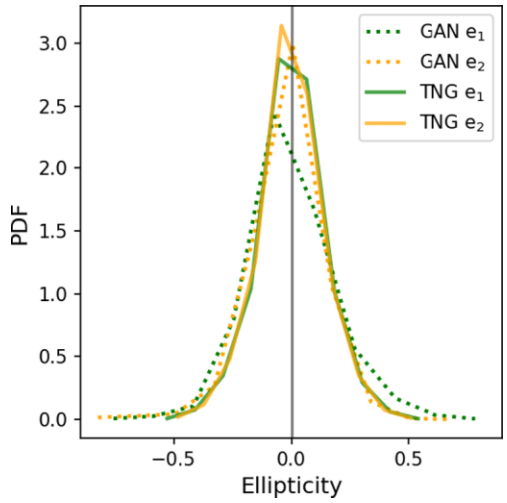

Figure 5. Distribution of individual components of the 2D complex ellipticities. The measured values from TNG100-1 are shown with solid lines, whereas the GAN-generated ellipticities are shown with dotted lines, with green denoting e_1 and yellow denoting e_2. The grey vertical line serves as a reference point of zero. The GAN produces distributions of ellipticity values that agree quantitatively with the distributions measured directly from TNG. The distributions have means of −0.02, 0.00, −0.01, 0.00 and standard deviations of 0.17, 0.15, 0.14, 0.14 for GAN e_1, GAN e_2, TNG e_1, TNG e_2, respectively.

simulations, since it is based on the kinematics of the galaxy (unless galaxy morphologies are included as part of the model).

Finally, we quantify whether the GAN captures the different intrinsic alignments of central versus satellite galaxies. In Fig. 7, we present the w_{g+} measured using different combinations of central and satellite galaxies for the density and shape samples, as indicated by the labels. The figure shows that the GAN-generated w_{g+} follow similar trends as the TNG measured w_{g+} when distinguishing between central and satellite alignments. However, the results may differ at the level of a factor of two on small scales for the All-Sat, Cent-Cent, and Sat-Cent curves, which exhibit higher alignment signals, with the GAN overestimating the alignments. For the All-Cent, Sat-Sat, and Cent-Sat curves the GAN also captures the trend, and even though it is in quantitative agreement with TNG, we note that these signals are very low and noise dominated.

6 CONCLUSIONS

In this article, we have developed a novel deep generative model for intrinsic alignments. Using the TNG100 hydrodynamical simulation from the IllustrisTNG simulation suite, we have trained the model to accurately predict 3D shapes and the 3D orientations of the major axes of both DM subhaloes and galaxies; and the projected 2D complex ellipticities of galaxies. For a simulation box of 75 Mpc h^{-1} with *20k* galaxies it takes about 3–4 d on a modern GPU to train; applying the model on a data set of equal size is very fast (less than a minute).

For the scalar quantities – the shapes of DM subhaloes and galaxies – the GAN model generated values that were in good quantitative agreement with the distributions of their actual measured counterparts from the TNG simulation. However, the correlation between the two shape parameters (intermediate-to-major and minor-to-major axial ratio) was slightly underestimated, by around 5–15 per cent depending on the galaxy or subhalo subsample.

Next, for the 3D vector quantities, the GAN generated the 3D major axis for the whole sample, and the resulting alignment correlation functions agree well for most scales except the very small and the 1- to 2-halo transition scales. When considering mass-selected

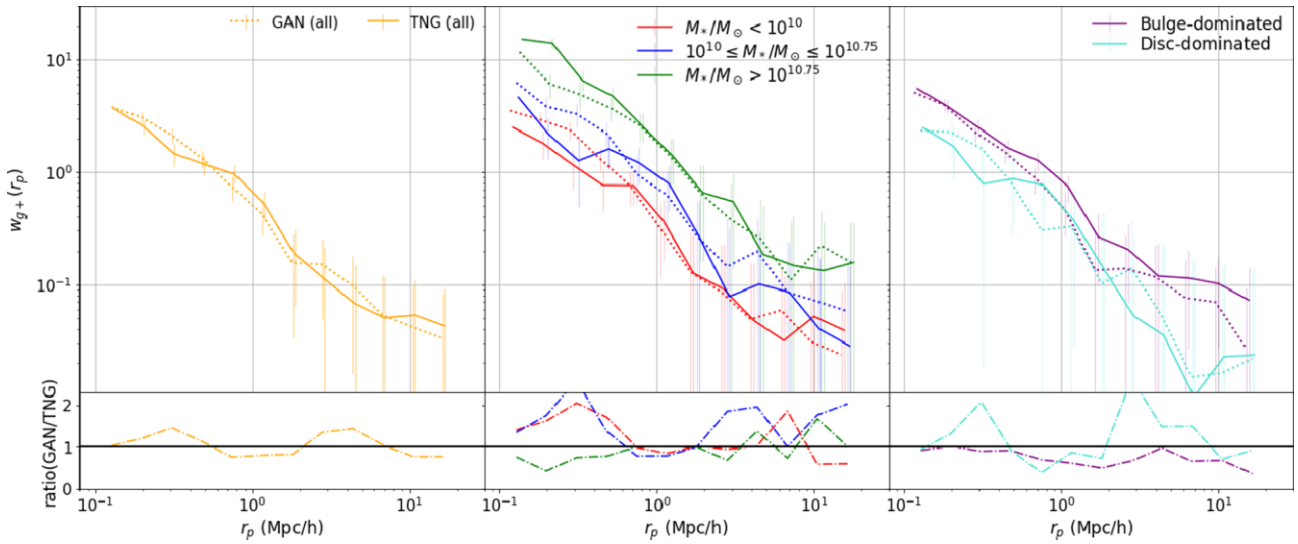

Figure 6. The projected two-point correlation functions w_{g+} of galaxy positions and the projected 2D ellipticities of all galaxies (first column), for subsamples divided by mass (second column) and for two morphological subsamples (third column), as indicated in the legends. The top rows show w_{g+} measured using data from the TNG simulation in solid lines and the data generated by the GAN in dotted lines, the bottom panel shows the ratio of the GAN curve to the TNG curve. For the first column, the two curves are in good quantitative agreement an all scales, with ratios closely following 1. For the second column, qualitatively the GAN captures the trends with mass, with higher mass bins showing higher alignment signal. Quantitatively, there is good agreement down to 0.5 Mpc h^{-1} scales. At scales below 0.5 Mpc h^{-1} the GAN underestimates the signal for the highest mass bin, whilst overestimating the signal for the lower two mass bins. For the third column, the GAN correctly reproduces the different intrinsic alignment for the two morphological types, even though this information was not explicitly given, whilst maintaining a quantitative agreement with the measured values. The errors are dominated by large-scale structure and may be correlated between the curves shown for TNG and the GAN, rather than being independent.

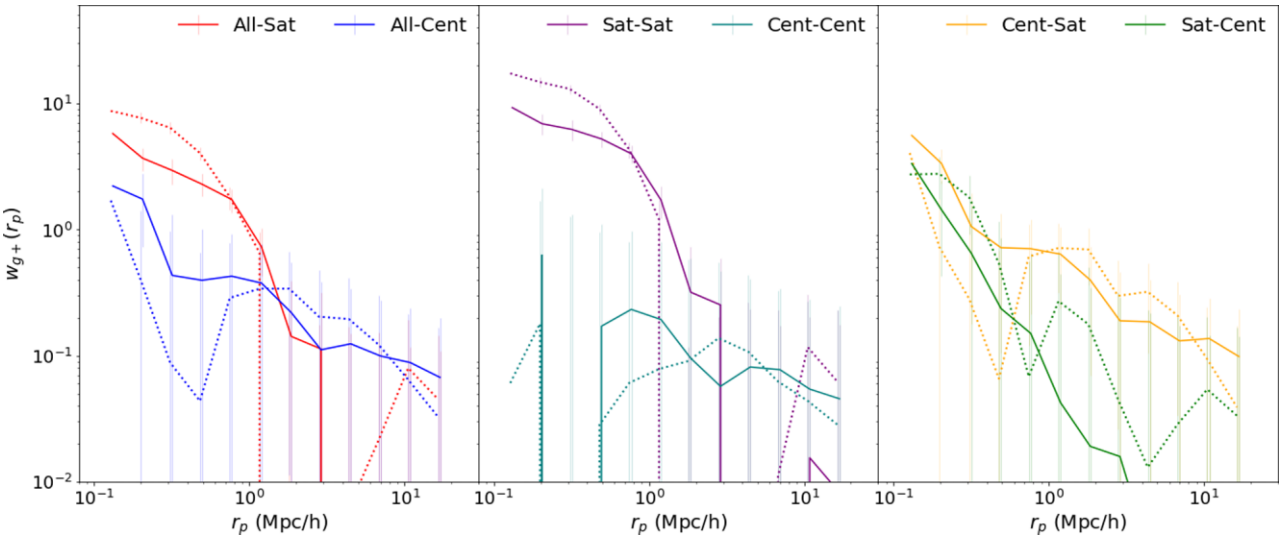

Figure 7. Projected two-point correlation function w_{g+} of galaxy positions and the projected 2D ellipticities, for subsamples defined by their central-satellite distinction. The plot shows w_{g+} measured using data from the TNG simulation in solid lines and the data generated by the GAN in dotted lines The labels specify which samples were used as the density tracers (first word) and shape samples (second word), e.g. 'All-Cent' refers to the correlation between the shapes of central galaxies and the positions of all galaxies. Qualitatively, the GAN captures the distinction between centrals and satellites. We excluded the ratio panels in this figure because some w_{g+} values were zero, and we only wanted to present the qualitative agreement.

subsamples, the results for small and medium mass subhaloes and galaxies are similar to those of the whole sample. However, for the highest mass bin, the GAN-generated values underestimate the alignment correlation function by about a factor of 2 on most scales.

Finally, for the projected 2D complex ellipticities, the projected density–shape correlation functions (w_{g+}) composed using the GAN-generated ellipticities are in excellent quantitative agreement with those from TNG100. Even when considering mass-selected subsamples, the quantitative agreement is good, except for scales below 0.5 Mpc/h where the high mass w_{g+} tends to be underestimated, while w_{g+} for intermediate and lower mass subsamples is overestimated by about a factor of 2. Also, the can qualitatively capture the IA trend for centrals and satellites, and for morphology-selected subsamples.

Overall, the Graph Convolution based Generative Adversarial network learns and generates scalar and vector quantities that have statistical properties (distributions and alignment correlations) that agree well with those of the simulation. The primary deficiencies were in high mass subsamples, perhaps due to insufficient training data in that regime.

In the future, we would like to deploy this model on a much higher volume N-body simulation with lower resolution in order to fully harness its power for upcoming weak lensing surveys. For this purpose, we will need the features listed in Fig. 1, which are usually available for *N*-body simulations. In the future, we will demonstrate the performance of the model on low-resolution, large-volume *N*-body simulations, such as the TNG300 and possibly other large volume simulations. As another area for future work, when we modelled the 3D orientation we only did so for one axis of the ellipsoid. An interesting direction for future work is to use SO(3) or E(3) equivariant neural networks for graphs (Horie et al. 2020; Satorras, Hoogeboom & Welling 2021) and point sets (Thomas et al. 2018; Villar et al. 2021). However, if one wants to use a light cone (which will need additional training across redshifts), or use 2D alignment statistics within a snapshot, then a new implementation of SO(3) equivariant neural networks is not needed. Given an *N*-body simulation with the correct features, our model can be used to include IA for very little additional computational cost.

ACKNOWLEDGEMENTS

We thank Ananth Tenneti, Tiziana DiMatteo, Barnabas Poczos, and Rupert Croft for useful discussion that informed the direction of this work. This work was supported in part by the National Science Foundation, NSF AST-1716131 and by a grant from the Simons Foundation (Simons Investigator in Astrophysics, Award ID 620789). SS is supported by a McWilliams postdoctoral fellowship at Carnegie Mellon University.

DATA AVAILABILITY

The data used in this paper is publicly available. The IllustrisTNG data can be obtained through the website at https://www.tng-project.org/data/. The catalogue data with morphological decompositions of galaxies is available at https://github.com/McWilliamsCenter/gal_decomp_paper. The software developed as part of this work is available at https://github.com/melon-lemon/GraphGAN.

REFERENCES

Arjovsky M., Chintala S., Bottou L., 2017, preprint (arXiv:1701.07875)
Bishop C., 1994, Workingpaper, Mixture Density Networks. Aston University, Birmingham
Blazek J. A., MacCrann N., Troxel M. A., Fang X., 2019, Phys. Rev. D, 100, 103506
Bridle S., King L., 2007, New J. Phys., 9, 444
Catelan P., Kamionkowski M., Blandford R. D., 2001, MNRAS, 320, L7
Chisari N. et al., 2015, MNRAS, 454, 2736
Davis M., Efstathiou G., Frenk C. S., White S. D. M., 1985, ApJ, 292, 371
Defferrard M., Bresson X., Vandergheynst P., 2016, in Proceedings of the 30th International Conference on Neural Information Processing Systems, NIPS'16. Curran Associates Inc., Red Hook, NY, p. 3844
Dubois Y., Peirani S., Pichon C., Devriendt J., Gavazzi R., Welker C., Volonteri M., 2016, MNRAS, 463, 3948

Fortuna M. C., Hoekstra H., Joachimi B., Johnston H., Chisari N. E., Georgiou C., Mahony C., 2021, MNRAS, 501, 2983
Goodfellow I. J., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A., Bengio Y., 2014, preprint (arXiv:1406.2661)
Gulrajani I., Ahmed F., Arjovsky M., Dumoulin V., Courville A., 2017, preprint (arXiv:1704.00028)
Guo Q. et al., 2011, MNRAS, 413, 101
Hearin A. P. et al., 2017, AJ, 154, 190
Heitmann K. et al., 2019, ApJS, 245, 16
Hirata C. M., Seljak U., 2004, Phys. Rev. D, 70, 063526
Ho M., Rau M. M., Ntampaka M., Farahi A., Trac H., Póczos B., 2019, ApJ, 887, 25
Horie M., Morita N., Hishinuma T., Ihara Y., Mitsume N., 2020, preprint (arXiv:2005.06316)
Jagvaral Y., Campbell D., Mandelbaum R., Rau M. M., 2021, preprint (arXiv:2105.02237)
Jagvaral Y., Singh S., Mandelbaum R., 2022, MNRAS, 514, 1021
Joachimi B., Semboloni E., Bett P. E., Hartlap J., Hilbert S., Hoekstra H., Schneider P., Schrabback T., 2013, MNRAS, 431, 477
Khandai N., Di Matteo T., Croft R., Wilkins S., Feng Y., Tucker E., DeGraf C., Liu M.-S., 2015, MNRAS, 450, 1349
Kiessling A. et al., 2015, Space Sci. Rev., 193, 67
Kilbinger M., 2015, Rep. Progr. Phys., 78, 086901
Kingma D. P., Ba J., 2014, preprint (arXiv:1412.6980)
Kipf T. N., Welling M., 2016, preprint (arXiv:1609.02907)
Kodi Ramanah D., Wojtak R., Ansari Z., Gall C., Hjorth J., 2020, MNRAS, 499, 1985
Korytov D. et al., 2019, ApJS, 245, 26
Lee J., Springel V., Pen U.-L., Lemson G., 2008, MNRAS, 389, 1266
Li Y., Ni Y., Croft R. A. C., Di Matteo T., Bird S., Feng Y., 2021, PNAS, 118, e2022038118
Mandelbaum R. et al., 2011, MNRAS, 410, 844
Marinacci F., et al., 2018, MNRAS, 480, 5113
Mathieson L., Moscato P., 2019, An Introduction to Proximity Graphs. Springer International Publishing, Cham, p. 213
Miyato T., Kataoka T., Koyama M., Yoshida Y., 2018, preprint (arXiv:1802.05957)
Naiman J. P. et al., 2018, MNRAS, 477, 1206
Nelson D., et al., 2018, MNRAS, 475, 624
Nelson D. et al., 2019, Comput. Astrophys. Cosmol., 6, 2
Ntampaka M. et al., 2019, Bull. Am. Astron. Soc., 51, 14
Pereira M. J., Bryan G. L., Gill S. P. D., 2008, ApJ, 672, 825
Pillepich A. et al., 2018a, MNRAS, 473, 4077
Pillepich A. et al., 2018b, MNRAS, 475, 648
Potter D., Stadel J., Teyssier R., 2017, Comput. Astrophys. Cosmol., 4, 2
Samuroff S., Mandelbaum R., Blazek J., 2020, preprint (arXiv:2009.10735)
Satorras V. G., Hoogeboom E., Welling M., 2021, in Meila M., Zhang T., eds, Proceedings of Machine Learning Research, Vol. 139, Proceedings of the 38th International Conference on Machine Learning. PMLR, p. 9323
Schaye J. et al., 2015, MNRAS, 446, 521
Schneider M. D., Bridle S., 2010, MNRAS, 402, 2127
Singh S., Mandelbaum R., Seljak U., Slosar A., Vazquez Gonzalez J., 2017, MNRAS, 471, 3827
Somerville R. S., Davé R., 2015, ARA&A, 53, 51
Somerville R. S., Hopkins P. F., Cox T. J., Robertson B. E., Hernquist L., 2008, MNRAS, 391, 481
Springel V., 2010, MNRAS, 401, 791
Springel V., White S. D. M., Tormen G., Kauffmann G., 2001, MNRAS, 328, 726
Springel V., et al., 2018, MNRAS, 475, 676
Tempel E., Saar E., Liivamägi L. J., Tamm A., Einasto J., Einasto M., Müller V., 2011, A&A, 529, A53
Tenneti A., Mandelbaum R., Di Matteo T., Feng Y., Khandai N., 2014, MNRAS, 441, 470
Tenneti A., Mandelbaum R., Di Matteo T., 2016, MNRAS, 462, 2668
Thomas N., Smidt T., Kearnes S., Yang L., Li L., Kohlhoff K., Riley P., 2018, preprint (arXiv:1802.08219)
Troxel M. A., Ishak M., 2015, Phys. Rep., 558, 1
Velliscig M. et al., 2015, MNRAS, 454, 3328
Verma N., Boyer E., Verbeek J., 2017, preprint (arXiv:abs/1706.05206)
Villar S., Hogg D. W., Storey-Fisher K., Yao W., Blum-Smith B., 2021, Adv. Neural Inform. Process. Syst., 34, 28848
Vogelsberger M. et al., 2014, MNRAS, 444, 1518
Vogelsberger M., Marinacci F., Torrey P., Puchwein E., 2020, Nat. Rev. Phys., 2, 42
Wechsler R. H., Tinker J. L., 2018, ARA&A, 56, 435
Zhou J. et al., 2018, preprint (arXiv:1812.08434)

APPENDIX A: TESTING THE ROBUSTNESS OF THE FITS

In this appendix, we investigate the robustness of our deep generative model. Due to scarcity of cosmological hydrodynamical simulations and the relatively limited volume available in TNG100, we used the whole sample to train and test. Therefore, in this section we test whether the model has been overfit by splitting our sample roughly 50/50, while preserving group membership of subhaloes and galaxies. In Fig. A1, we present the model performance on both train and test samples.

Additionally, we measured the *Pearson-r* correlation coefficient between the generated and the measured complex 2D ellipticities. We generated 100 different samples, each with a different random seed, measured the desired *Pearson-r* coefficient and averaged the results. These were $\langle\rho(\mathrm{GAN}\, e_1, \mathrm{TNG}\, e_1)\rangle = 0.09$ and $\langle\rho(\mathrm{GAN}\, e_2, \mathrm{TNG}\, e_2)\rangle = 0.06$. The weak correlations suggest that the model did not just simply 'memorize' the ellipticities.

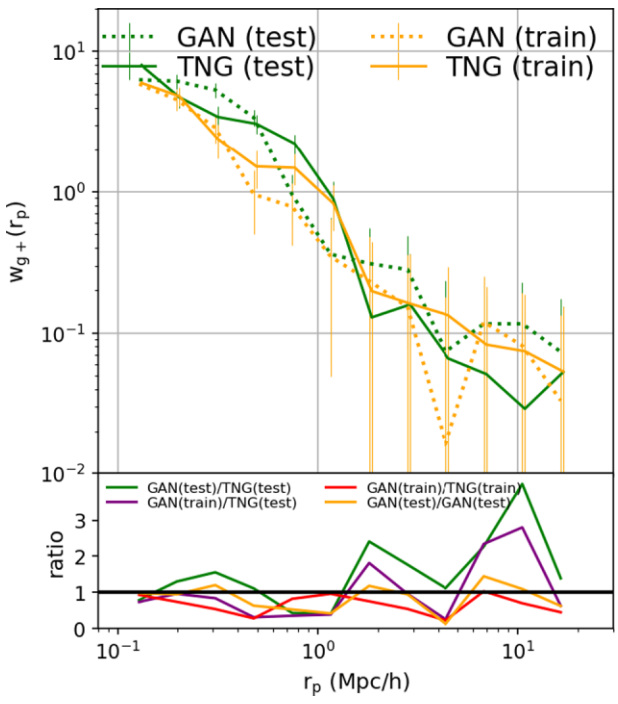

Figure A1. Projected two-point correlation functions w_{g+} of galaxy positions and the projected 2D ellipticities of all galaxies, split into roughly equal-sized training and testing samples while preserving group membership. The top panel shows w_{g+} measured using data from the TNG simulation in yellow and the data generated by the GAN in dotted green, while the bottom panel shows the ratios among the curves as indicated by the label. All four curves are in good quantitative agreement, suggesting that the GAN is not significantly overfitting.

APPENDIX B: ALEATORIC (STOCHASTIC) UNCERTAINTY

In this appendix we explore the uncertainty of the measured intrinsic alignment correlation functions due to the stochastic nature of the model implementation.[8]

In Fig. B1, we show the same quantity as in the first panel of Fig. 6, as well as its 1σ scatter obtained using 50 different random seeds from the GAN. As is evident from the plot, this scatter is substantially smaller than the analytic errors that quantity the combination of shape noise and cosmic variance. Thus, we do not propagate and show this random variance for every w_{g+} curve.

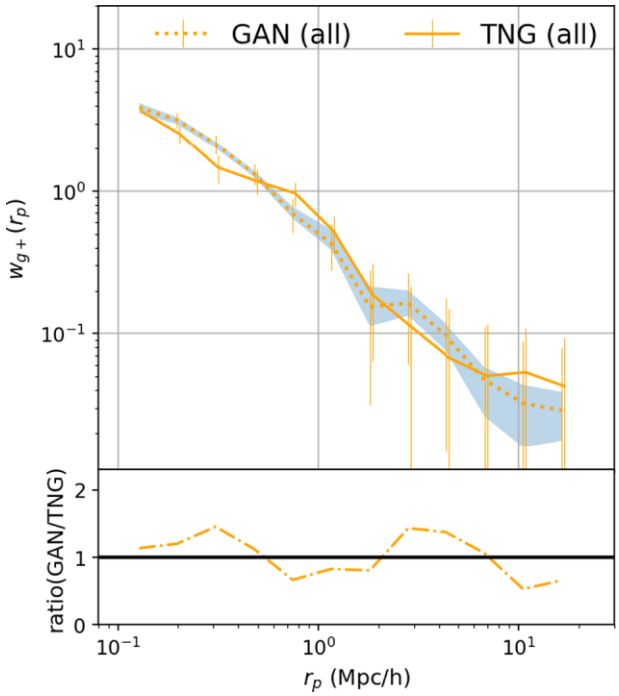

Figure B1. Projected two-point correlation function w_{g+} of galaxy positions and the projected 2D ellipticities of all galaxies. The top panel shows w_{g+} measured using data from the TNG simulation in yellow and the data generated by the GAN in dotted green, while the bottom panel shows the ratios among the curves as indicated by the label. In the blue shaded region we show the 1σ scatter of w_{g+} obtained using 50 different random seeds, as a measure of the stochastic uncertainty in the model.

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

[8]Note that due to the very complex nature of GANs and neural networks in general, it is usually very difficult to quantify model uncertainties, usually known as *epistemic uncertainty*. Here, we do not attempt to model this type of uncertainty.