Monthly Notices of the ROYAL ASTRONOMICAL SOCIETY

MNRAS **527,** 10038–10058 (2024) https://doi.org/10.1093/mnras/stad3784
Advance Access publication 2023 December 7

An observationally driven multifield approach for probing the circum-galactic medium with convolutional neural networks

Naomi Gluck,[1]★ Benjamin D. Oppenheimer,[2] Daisuke Nagai,[1] Francisco Villaescusa-Navarro[3,4] and Daniel Anglés-Alcázar[4,5]

[1]*Physics Department, Yale University, 217 Prospect Str, New Haven, CT 06511, USA*
[2]*CASA, Department of Astrophysical and Planetary Sciences, University of Colorado, 389 UCB, Boulder, CO 80309, USA*
[3]*Department of Astrophysical Sciences, Princeton University, Peyton Hall, Princeton, NJ 08544, USA*
[4]*Center for Computational Astrophysics, Flatiron Institute, 162 5th Avenue, New York, NY 10010, USA*
[5]*Department of Physics, University of Connecticut, 196 Auditorium Road, U-3046, Storrs, CT 06269, USA*

Accepted 2023 December 5. Received 2023 December 3; in original form 2023 September 12

ABSTRACT

The circum-galactic medium (CGM) can feasibly be mapped by multiwavelength surveys covering broad swaths of the sky. With multiple large data sets becoming available in the near future, we develop a likelihood-free Deep Learning technique using convolutional neural networks (CNNs) to infer broad-scale physical properties of a galaxy's CGM and its halo mass for the first time. Using CAMELS (Cosmology and Astrophysics with MachinE Learning Simulations) data, including IllustrisTNG, SIMBA, and Astrid models, we train CNNs on Soft X-ray and 21-cm (H I) radio two-dimensional maps to trace hot and cool gas, respectively, around galaxies, groups, and clusters. Our CNNs offer the unique ability to train and test on 'multifield' data sets comprised of both H I and X-ray maps, providing complementary information about physical CGM properties and improved inferences. Applying eRASS:4 survey limits shows that X-ray is not powerful enough to infer individual haloes with masses $\log(M_{\rm halo}/{\rm M}_{\odot}) < 12.5$. The multifield improves the inference for all halo masses. Generally, the CNN trained and tested on Astrid (SIMBA) can most (least) accurately infer CGM properties. Cross-simulation analysis – training on one galaxy formation model and testing on another – highlights the challenges of developing CNNs trained on a single model to marginalize over astrophysical uncertainties and perform robust inferences on real data. The next crucial step in improving the resulting inferences on the physical properties of CGM depends on our ability to interpret these deep-learning models.

Key words: software: simulations – galaxies: clusters: general – galaxies: groups: general – (galaxies:) intergalactic medium – radio lines: general – X-rays: general.

1 INTRODUCTION

New telescopes are currently engaged in comprehensive surveys across large sky areas and reaching previously unobtainable depths, aiming to map the region beyond the galactic disc but within the galaxy's virial radius: the circum-galactic medium (CGM; Tumlinson, Peeples & Werk 2017). However, these telescopes have inherent limitations in detecting emissions from gaseous haloes surrounding typical galaxies. Nevertheless, they offer an exceptional opportunity to characterize the broad properties of CGM that extend beyond their original scientific scope. The CGM contains a multiphase gas, partly accreted from the filaments of the cosmic web that is continuously being reshaped, used in star formation, and enriched by astrophysical feedback processes occurring within the galaxy (Keres et al. 2005; Christensen et al. 2016; Oppenheimer et al. 2016; Anglés-Alcázar et al. 2017b; Hafen et al. 2019).

A simple way to characterize the CGM is by temperature. The cool phase gas has a temperature of approximately $T \sim 10^4$ K and has been the focus of UV absorption line measurements (e.g. Cooksey et al. 2010; Tumlinson et al. 2013; Werk et al. 2013; Johnson et al. 2015; Keeney et al. 2018). The hot phase of the CGM, with temperatures $T > 10^6$ K, is observable via X-ray facilities (e.g. Bogdán et al. 2018; Bregman et al. 2018; Mathur et al. 2023) and can contain the majority of a galaxy's baryonic content. Understanding both the cool and hot phases of the CGM may answer questions regarding where we may find baryons (Anderson & Bregman 2011; Werk et al. 2014; Li et al. 2017; Oppenheimer et al. 2018), how galaxy quenching proceeds (Tumlinson et al. 2011; Somerville, Popping & Trager 2015), and how the metal products of stellar nucleosynthesis are distributed (Peeples et al. 2014).

New, increasingly large data sets that chart the CGM across multiple wavelengths already exist. In particular, two contrasting wavelengths map diffuse gas across nearby galaxies: the X-ray and the 21-cm (neutral hydrogen, H I) radio. First, the *eROSITA*[1] mission has conducted an all-sky X-ray survey, enabling the detection of diffuse emission from hot gas associated with groups and clusters

★ E-mail: naomi.gluck@yale.edu

[1]Although *eROSITA* is currently dormant; its data at the level we mock have already been taken.

Table 1. Definitions and global value ranges of the CGM properties to be inferred and constrained by the network. These are the global value ranges, encompassing the individual ranges of IllustrisTNG, SIMBA, and Astrid. They remain consistent throughout any combination of simulations during training and testing. Properties are further distinguished by those radially defined by R_{200c} and those by 200 kpc.

Property	Definition	Range
$M_{\rm halo}$	Logarithmic halo mass in R_{200c}	11.5–14.3
$f_{\rm cgm}$	Mass ratio of CGM gas to total mass within R_{200c}	0.0–0.23
$Z_{\rm cgm}$	Logarithmic CGM metallicity in 200 kpc	−3.6 – −1.3
$M_{\rm cgm}$	Logarithmic CGM mass in 200 kpc	8.0–12.5
$f_{\rm cool}$	Ratio of cool, low-ionized CGM gas within 200 kpc	0.0–1.0
$T_{\rm cgm}$	Logarithmic CGM temperature in 200 kpc	3.9–7.6

and potentially massive galaxies (Predehl et al. 2021). Secondly in the 21-cm radio domain, the pursuit of detecting cool gas encompasses initiatives that serve as precursors to the forthcoming Square Kilometer Array (SKA) project. Notable among these are ASKAP (Johnston et al. 2007) and MeerKAT (Jonas & MeerKAT Team 2016), both of which have already conducted comprehensive surveys of H I gas in galaxy and group environments through deep 21-cm pointings.

Cosmological simulations provide theoretical predictions of CGM maps, yet divergences arise due to varying hydrodynamic solvers and subgrid physics modules employed in galaxy formation simulations (Somerville, Popping & Trager 2015; Tumlinson, Peeples & Werk 2017; Davé et al. 2020). As a result, we see very different predictions for the circumgalactic reservoirs surrounding galaxies. Distinctively, the publicly available simulations such as IllustrisTNG (Nelson et al. 2018; Pillepich et al. 2018), SIMBA (Davé et al. 2019), Astrid (Bird et al. 2022; Ni et al. 2022), among others (e.g. Schaye et al. 2015; Hopkins et al. 2018; Wetzel et al. 2023), are valuable resources for generating CGM predictions. CAMELS[23] (Cosmology and Astrophysics with MachinE Learning Simulations) is the first publicly available suite of simulations that includes thousands of parameter and model variations designed to train machine learning models (Villaescusa-Navarro et al. 2021c, 2022). It contains four different simulations *sets* covering distinct cosmological and astrophysical parameter distributions: LH (Latin Hypercube, 1000 simulations), 1P (1-Parameter variations, 61 simulations), CV (Cosmic Variance, 27 simulations), and EX (Extreme, 4 simulations). Of these, the CV set is uniquely significant as it fixes cosmology and astrophysics to replicate the observable properties of galaxies best, providing a fiducial model. We exclude the numerous CAMELS simulations that vary cosmology and astrophysical feedback to prevent unrealistic galaxy statistics. Thus, utilizing the diverse CAMELS CV sets, we explore three universe realizations that make distinguishing predictions for the CGM.

In this study, we develop an image-based convolutional neural network (CNN) to infer CGM properties from CAMELS IllustrisTNG, SIMBA, and Astrid CV-set simulations. The definitions and ranges for all CGM properties are outlined in Table 1. Two significant and differently structured astrophysical feedback parameters that impact CGM properties, stellar and AGN feedback, remain predominantly unconstrained. The CV set does not explore the range of CAMELS feedback parameters like the other sets. However, we choose the CV set as a proof-of-concept and plan to include the much larger LH set that completely marginalizes over astrophysics (Villaescusa-Navarro et al. 2021b) in the future. The CNN is trained and tested on diverse simulations, yielding valuable insights into the CGM properties. Additionally, we apply observational multiwavelength survey limits to the CNN for each field, guiding the design and approach of new instruments and novel surveys, maximizing their scientific returns on CGM properties, and significantly advancing our understanding of galaxy formation and large-scale structure.

This paper is outlined as follows. Section 2 lays out the methods used to complete this work and includes subsections on specific simulation information (Section 2.1), data set generation (Section 2.2), CNNs (Section 2.3), and network output (Section 2.4). We begin Section 3 by presenting results using individual simulations to infer first the entire halo mass (Section 3.1), then a global CGM property, the mass of the CGM over the mass of the halo, or $f_{\rm cgm}$ (Section 3.2), and the metallicity of the CGM (Section 3.3) which exhibits large variation. We show results based on idealized soft X-ray and H I images and assess the impact of realistic observations with observational survey limits (Section 3.4). We also perform *cross simulation inference*, where one trains a CNN on one galaxy formation model or simulation and tests on another to gauge its robustness (Section 3.5). We discuss the interpretability of the cross-simulation inference analysis (Section 4.1), the applicability and limitations of CNNs applied to CGM (Section 4.2), the variance between true and inferred values for CGM properties using the idealised multifield maps (Section 4.3), and a possible avenue for future work as an expansion of this analysis (Section 4.4). Lastly, Section 5 concludes.

2 METHODS

In this section, we introduce the simulations (Section 2.1) followed by how our halo-centric 'map' data sets are generated and a description of the global properties we train the network to infer (Section 2.2). Then, Section 2.3 describes the neural network applied to these data sets. Finally, we specify the network output, including statistical measures, to evaluate the performance of CNN (Section 2.4).

We define some vocabulary and common phrases within this work. *Fields* refer to X-ray and 21-cm H I (hereafter H I), where using one field corresponds to either X-ray or H I; two fields, X-ray and H I, make up the multifield. With our CNN architecture, the number of fields is equivalent to the number of channels. *Parameters* and *hyperparameters* define the inner workings of the CNN, where the latter must be optimized. This should not be confused with parameters in the context of astrophysical feedback. *Properties* describe the attributes of the CGM that are inferred by the network: $M_{\rm halo}$, $f_{\rm cgm}$, $\log(Z_{\rm cgm})$, $M_{\rm cgm}$, $f_{\rm cool}$, and $\log(T_{\rm cgm})$. The *parameter space* reflects the range of values for the CGM properties (between the 16th and 84th percentiles) that each simulation encapsulates.

2.1 Simulations

We use the CV set from three simulation suites, each of which uses a different hydrodynamic scheme: CAMELS-IllustrisTNG (referred to as IllustrisTNG) using AREPO (Springel 2010; Weinberger, Springel & Pakmor 2020), CAMELS-SIMBA (referred to as SIMBA) utilizing GIZMO (Hopkins 2015), and CAMELS-Astrid (referred to as Astrid) utilizing MP-Gadget (Springel 2005). These simulations encompass 27 volumes spanning $(25h^{-1}{\rm Mpc})^3$ with fixed cosmological parameters ($\Omega_{\rm M} = 0.3$ and $\sigma_8 = 0.8$) with varying

[2]CAMELS Project Website: https://www.camel-simulations.org
[3]CAMELS Documentation available at https://camels.readthedocs.io/en/latest/index.html

Table 2. Outlining the number of haloes per mass bin in IllustrisTNG, SIMBA, and Astrid. The mass bins are defined as follows: Sub-$L*$ for small haloes with mass between $11.5 \leq \log(M_{\rm halo}/{\rm M}_\odot) \leq 12$, $L*$ for intermediate-sized haloes with masses ranging from $12 \leq \log(M_{\rm halo}/{\rm M}_\odot) \leq 13$, and Groups are large haloes with masses from $13 \leq \log(M_{\rm halo}/{\rm M}_\odot) \leq 14.3$.

Simulation	Sub-$L*$	$L*$	Group	Total
IllustrisTNG	3450	1812	192	5454
SIMBA	3397	1534	170	5101
Astrid	3262	1866	218	5346

random seeds for each volume's initial condition. The CAMELS astrophysical parameters for feedback are set to their fiducial values. We exclusively use the $z = 0$ snapshots for this work.

IllustrisTNG is an adaptation of the original simulation as described in Nelson et al. (2019) and Pillepich et al. (2018), using the AREPO (Springel 2010) magnetohydrodynamics code employing the *N*-body tree-particle-mesh approach for solving gravity and magnetohydrodynamics via moving-mesh methods. Like all simulation codes used here, IllustrisTNG has subgrid physics modules encompassing stellar processes (formation, evolution, and feedback) and black hole processes (seeding, accretion, and feedback). Black hole feedback uses a dual-mode approach that applies thermal feedback for high-Eddington accretion rates and kinetic feedback for low-Eddington rate accretion rates. The kinetic mode is directionally pulsed and is more mechanically efficient than the thermal mode (Weinberger et al. 2017).

SIMBA, introduced in Davé et al. (2019), uses the hydrodynamic-based 'Meshless Finite Mass' GIZMO code (Hopkins 2015, 2017), with several unique subgrid prescriptions. It includes more physically motivated implementations of (1) AGN feedback and (2) black hole growth. SIMBA's improved subgrid physics model for AGN feedback is based on observations, utilizing kinetic energy outflows for both radiative and jet feedback modes operating at high and low Eddington ratios, respectively. Additionally, it applies observationally motivated X-ray feedback to quench massive galaxies. SIMBA's black hole growth model is phase-dependent. Cool gas accretion on to BHs is achieved through a torque-limited accretion model (Anglés-Alcázar et al. 2017a), and when accreting hot gas, SIMBA transitions to Bondi accretion.

Astrid, introduced in Bird et al. (2022), adopts the Pressure–Entropy SPH hydrodynamic model that uses the MP-Gadget code (Feng et al. 2018). The original Astrid simulations focus on modelling high-redshift galaxy formation (from $z = 99$ to $z = 3$) by considering inhomogeneous hydrogen and helium reionization, metal return from massive stars, and the initial velocity offset between baryons and dark matter. It has also enhanced the modelling of black hole mergers via a dynamic friction model. The CAMELS version of Astrid (Ni et al. 2023) follows the original simulation, but slight changes in black-hole dynamics and dual-mode AGN feedback implementations were made between them.

2.2 Data set generation

To create our halo-centric map data sets, we use YT-based software (Turk et al. 2011) that allows for consistent and uniform analysis across different simulation codes. We generate maps of all haloes within the CV set with masses of at least $M_{\rm halo} = 10^{11.5}\,{\rm M}_\odot$ along the three cardinal axes. There are approximately 5000 haloes for each simulation. The highest halo mass is $10^{14.3}\,{\rm M}_\odot$, for a nearly 3 dex span in halo mass. Refer to Table 2 for additional details. We categorize all the haloes within the simulations by halo mass, where Sub-$L*$ haloes are within the range $11.5 \leq \log(M_{\rm halo}/{\rm M}_\odot) \leq 12$, $L*$ haloes are within the range $12 \leq \log(M_{\rm halo}/{\rm M}_\odot) \leq 13$, and groups are within the range $13 \leq \log(M_{\rm halo}/{\rm M}_\odot) \leq 14.3$.

The relationship between $\log(M_{\rm halo}/{\rm M}_\odot)$ and $\log(M_{\rm cgm})$, $\log(T_{\rm cgm})$, $f_{\rm cgm}$, and $\log(Z_{\rm cgm})$ for all simulations, the parameter space, is shown in Fig. 1. The mean value of each property is indicated with a solid line. The shaded regions represent the 16th−84th percentiles, and the dotted points indicate the 'statistically low' region for haloes with halo masses above $\log(M_{\rm halo}/{\rm M}_\odot) > 13.0$. In agreement with previous work (Oppenheimer et al. 2021; Delgado et al. 2023; Gebhardt et al. 2023; Ni et al. 2023), we illustrate how the properties of gas beyond the galactic disc can differ significantly between feedback implementations.

For $\log(M_{\rm cgm})$ (top left), Astrid (blue) shows little scatter below $\log(M_{\rm halo}/{\rm M}_\odot) > 12.5$, IllustrisTNG (pink) shows similar but less extreme scatter, and SIMBA (purple) has consistent scatter throughout. In $\log(T_{\rm cgm})$ (top right), Astrid again has a low scatter throughout the entire $M_{\rm halo}$ range. This scatter increases slightly for IllustrisTNG and again for SIMBA, and it is interesting to note the divergence from $\log(T_{\rm cgm}) \propto \log(M_{\rm halo}/{\rm M}_\odot)^{2/3}$. Astrid has the most scatter for $f_{\rm cgm}$ (bottom left), whereas IllustrisTNG and SIMBA display comparable scatter for lower masses, reducing for higher masses. Finally, $\log(Z_{\rm cgm})$ illustrates that all three simulations have significant and similar scatter. For $M_{\rm cgm}$, $\log(T_{\rm cgm})$, and $f_{\rm cgm}$, Astrid has higher values throughout the $M_{\rm halo}$ range, followed by IllustrisTNG and SIMBA. This is not the case in $\log(Z_{\rm cgm})$, where there is a significant overlap. The scatter in $M_{\rm halo}$ was also computed with respect to the total flux per map, corresponding to the sum of all pixel values in X-ray and H I separately. When binned by $M_{\rm halo}$, there are correlations only with IllustrisTNG and Astrid for X-ray (see Fig. A2). A more detailed discussion of map trends and pixel counts is in Appendix A.

From the snapshot data obtained from the X-ray and H I maps, we provide an equation describing the calculation of each CGM property ($M_{\rm halo}$, $f_{\rm cgm}$, $Z_{\rm cgm}$, $M_{\rm cgm}$, $f_{\rm cool}$, and $T_{\rm cgm}$):

$$M_{\rm halo} = \sum m_{\rm DM}(r < R_{\rm 200c}) + \sum m_{\rm gas}(r < R_{\rm 200c}) + \sum m_{\rm star}(r < R_{\rm 200c}) \quad (1)$$

$$f_{\rm cgm} = \frac{\sum m_{\rm cgm}(r < R_{\rm 200c})}{M_{\rm halo}} \quad (2)$$

$$Z_{\rm cgm} = \frac{\sum z_{\rm cgm}(r < 200\ {\rm kpc})}{\sum m_{\rm cgm}(r < 200\ {\rm kpc})} \quad (3)$$

$$M_{\rm cgm} = \sum m_{\rm cgm}(r < 200{\rm kpc}) \quad (4)$$

$$f_{\rm cool} = \frac{\sum m_{\rm cool}(r < 200\ {\rm kpc})}{\sum m_{\rm cgm}(r < 200\ {\rm kpc})} \quad (5)$$

$$T_{\rm cgm} = \frac{\sum t_{\rm cgm}(r < 200\ {\rm kpc})}{\sum m_{\rm cgm}(r < 200\ {\rm kpc})} \quad (6)$$

where m is the mass of dark matter (DM), gas, or stellar (star) particles enclosed within $r < 200$ kpc. The subscript 'cgm' refers to any gas that is not star-forming. $z_{\rm cgm}$ is the metallicity of the gas particle. $m_{\rm cool}$ is CGM gas with $T < 10^6$ K. $t_{\rm cgm}$ is the temperature of the gas particle. For the definitions and numerical ranges of the above CGM properties, see Table 1. To ensure our CNN is able to reproduce the scatter seen in Fig. 1, we include a comparison of mean and variance values between the input parameters and the output inference, separated by mass bin for each galaxy simulation model. This is only computed for $\log(Z_{\rm cgm})$, as this parameter does

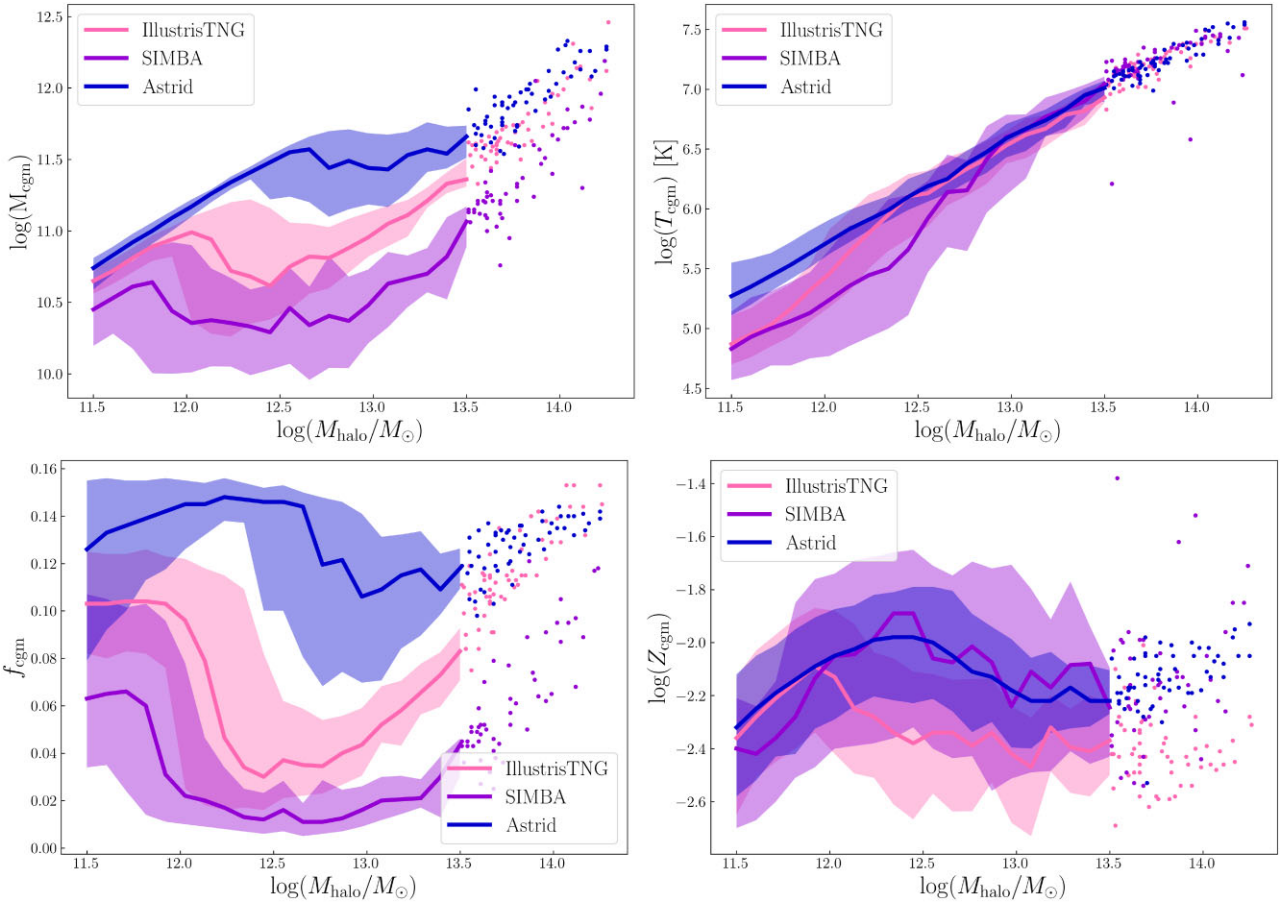

Figure 1. Relationship between different CGM properties and the halo mass. Panels specifically illustrate $M_{\rm cgm}$, $\log(T_{\rm cgm})$, $f_{\rm cgm}$, and $\log(Z_{\rm cgm})$ within each simulation (IllustrisTNG, SIMBA, and Astrid) to represent the mean distribution of the objects. The points indicate the mass bins where there are statistically fewer haloes in mass bins above $\log(M_{\rm halo}/{\rm M}_\odot) = 13.5$. The shaded regions represent the $16^{\rm th} - 84^{\rm th}$ percentiles.

not have a clear relationship with halo mass, enabling a distinction between mass relationships (seen within the other properties) and intrinsic scatter. We confirm that our CNN reproduces the scatter within the initial data sets.

We generated one channel for each field (H I or X-ray), adding them together in the multifield case (H I + X-ray). Each map utilizes values obtained through mock observation, as described below. For X-ray, we map X-ray surface brightness emission in the soft band between 0.5 and 2.0 keV. H I, or 'Radio' refers to the 21-cm emission-based measurement that returns column density maps, which is a data reduction output of 21-cm mapping techniques. Each map is 128x128 pixels, spanning 512×512 kpc^2 with a 4 kpc resolution. The depth spans $\pm$1000 kpc from the centre of the halo. Two types of maps are generated for each field: those with no observational limits, called idealized maps, and those with observed limits imposed. We first explain the generation of idealized maps. X-ray maps are created using the pyXSIM package (ZuHone & Hallman 2016). While pyXSIM can generate lists of individual photons, we use it in a non-stochastic manner to map the X-ray emission across the kernel of the fluid element. Therefore, our X-ray maps are idealized in their ability to map arbitrarily low emission levels. Radio-based H I column density maps are created using the Trident package (Hummels, Smith & Silvia 2017) where the Haardt & Madau (2012) ionization background is assumed with the self-shielding criterion of Rahmati et al. (2013) applied.

Fig. 2 depicts maps of the same massive halo in the three simulations: IllustrisTNG, SIMBA, and Astrid, from left to right, respectively. The four rows illustrate (1) idealized X-ray, (2) observationally limited X-ray, (3) idealized H I, and (4) observationally limited H I. For X-ray with observational limits, we set the surface brightness limit to 2.0×10^{-13} erg s^{-1} cm^{-2} arcmin^{-2} corresponding to the observing depth of the *eROSITA* eRASS:4 all-sky survey (Predehl et al. 2021). For H I with observational limits, we set the column density limit to $N_{\rm H\,I} = 10^{19.0}$ cm^{-2}, which is approximately the limit expected for the 21-cm H I MHONGOOSE Survey at a 15' beam size similar to the *eROSITA* survey (de Blok et al. 2016). The observational limits are implemented by setting a lower limit floor that corresponds to the detectability of the telescope. Accessing the same halo across the three simulations is possible, since the CV set shares the same initial conditions between the different simulation suites. The X-ray maps tracing the gas primarily above $T > 10^6$ K are brightest for Astrid and dimmest for SIMBA, a trend also seen when the observational limits are imposed. The H I maps, probing $T \sim 10^4$ K gas, are less centrally concentrated than X-ray and often trace gas associated with satellites.

We expand on the first column in Fig. 2 in Fig. A1, formatted similarly, for a range of halo masses within IllustrisTNG from $\log(M_{\rm halo}/{\rm M}_\odot) = 13.83$ (leftmost) to $\log(M_{\rm halo}/{\rm M}_\odot) = 11.68$ (rightmost). X-ray emission, which traces the gas with a temperature above 10^6 K, indicates a strong correlation with the halo mass. The features

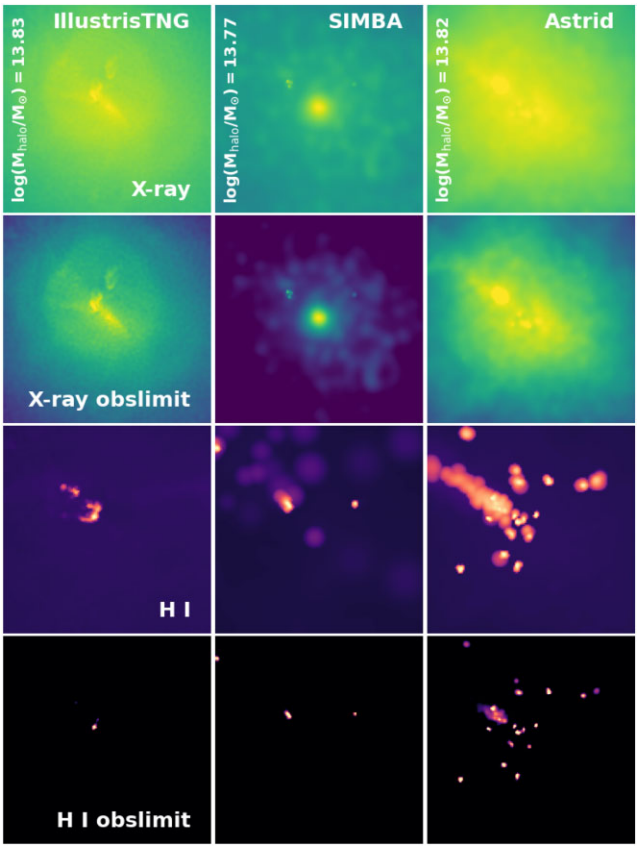

Figure 2. Each column illustrates maps for IllustrisTNG, SIMBA, and Astrid, respectively. Each row corresponds to maps for idealized X-rays, X-rays with observational limits, idealized H I, and H I with observational limits. These maps display the same halo across the CV set of the three simulations.

seen here include wind-blown bubbles (Predehl et al. 2020), satellite galaxies that create bow shocks (Kang et al. 2007; Bykov, Dolag & Durret 2008; Zinger et al. 2018; Li et al. 2022), and emissions associated with the galaxies themselves. H I does not have the same correlation with halo mass, strengthening our choice in creating the H I + X-ray multifield.

2.3 Convolutional neural network

The advantage of employing CNNs lies in their capacity to simultaneously learn multiple features from various channels or fields (X-ray and H I). Fields can be used independently or together for training, validation, and testing without modifications to the network architecture and only minor changes in the scripts whenever necessary. This work adopts likelihood-free inference methods, suitable for cases where determining a likelihood function for large and complex data sets is computationally demanding or is not attainable. Our CNN architecture is based on the architecture used with the CAMELS Multifield Data set (CMD) in Villaescusa-Navarro et al. (2021a), inferring two cosmological and four astrophysical feedback parameters. In addition to now inferring six CGM *properties*, the remaining modifications stem from replacing the LH set (Villaescusa-Navarro et al. 2022) with the CV set. We must be able to accommodate unevenly distributed discrete-valued data in the form of 'halo-centric' points, which can be seen in a property like the halo mass. This is compared to the LH-based CMD data set, which was used to infer the evenly distributed cosmological and astrophysical feedback parameters by design.

Our CNN makes two main adjustments: (1) the kernel size is changed from 4 to 2 to accommodate a smaller initial network input, and (2) the padding mode is altered from 'circular' to 'zeros'. The padding mode is crucial in guiding the network when the image dimensions decrease, as it no longer perfectly fits the original frame. Changing to 'zeros' means filling the reduced areas with zeros to maintain the network's functionality. The CNN architecture is outlined in greater detail in Appendix C in Table C1 for the main body of the CNN and Table C2 for additional functions utilized after the main body layers.

CNN also includes hyperparameters: (1) the *maximum learning rate* (also referred to as step size), which defines how the application of weights changes during training, (2) the *weight decay* as a regularization tool to prevent overfitting by reducing the model complexity, (3) the *dropout value* (for fully connected layers) as random neurone disablement to prevent overfitting, and (4) the *number of channels* in CNN (set to an integer larger than one). To optimize these hyperparameters, we employ Optuna (Akiba et al. 2019),[4] a tool that efficiently explores the parameter space and identifies the values attributed to returning the lowest validation loss, thus achieving the best performance.

We divide the full data set into a training set (60 per cent), a validation set (20 per cent), and a testing set (20 per cent). Only the training set contains the same halo along three different axis projections (setting the network parameter `split = 3`). In contrast, the latter sets include neither the axis projections of any halo nor the original image of the haloes assigned to the training set. The split is performed during each new training instance for a new combination of fields and simulations. We set the same random seed across all network operations, so as to drastically reduce the probability of overlap between training, validation, and testing sets. Without the same random seed, the data set will not be split in the same way each time, and one halo could appear in two or more sets, causing inaccurate results. This process is necessary to ensure that the network does not perform additional 'learning' in one phase.

2.4 Network outputs

Here, the 'moment' network (Jeffrey & Wandelt 2020) takes advantage of only outputting the mean, μ, and variance, σ, of each property for increased efficiency, instead of a full posterior range. The minimum and maximum values used to calculate the network error for the six CGM properties are kept the same throughout this work, regardless of which simulation is used for training. Doing so ensures that the results are comparable in the cross-simulation analysis or training on one simulation and testing on another.

We additionally include four metrics to determine the accuracy and precision of the CNN's outputs for each CGM property: the root mean squared error (RMSE), the coefficient of determination (R^2), the mean relative error (ϵ), and the reduced chi-squared (χ^2). In the formulae below, we use the subscript i to correspond to the index value of the properties [1–6], the marginal posterior mean, μ_i, and the standard deviation, σ_i. Four different statistical measurements are used to make such conclusions, and $\mathrm{TRUE_i}$ is used to denote the true value of any given CGM property with respect to simulation-based maps.

[4] https://github.com/pfnet/optuna/

Root mean squared error, RMSE:

$$\mathrm{RMSE_i} = \sqrt{\langle(\mathrm{TRUE_i} - \mu_\mathrm{i})^2\rangle}, \tag{7}$$

where smaller RMSE values can be interpreted as increased model accuracy in units that can be related to the measured property.

Coefficient of determination, R^2:

$$R_i^2 = 1 - \frac{\Sigma_i(\mathrm{TRUE_i} - \mu_\mathrm{i})^2}{\Sigma_i(\mathrm{TRUE_i} - \overline{\mathrm{TRUE_i}})^2}, \tag{8}$$

representing the scale-free version of the RMSE, where the closer R^2 is to one, the more accurate the model.

Mean relative error, ϵ:

$$\epsilon_i = \left\langle \frac{\sigma_i}{\mu_i} \right\rangle, \tag{9}$$

where smaller ϵ_i values can be interpreted as increased model precision. This is also the type of error predicted by CNN.

Reduced chi-squared, χ^2:

$$\chi_i^2 = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\mathrm{TRUE_i} - \mu_i}{\sigma_i}\right)^2, \tag{10}$$

where this quantifies how 'trustworthy' the posterior standard deviation is, such that values close to one indicate a properly quantified error and that the model is well trained. Values greater than one indicate that the errors are underestimated, and those smaller than one are overestimated. We do not expect deviations far from one when analysing inferences from CNNs trained and tested on the same simulation. However, values much higher than one are expected if network training and testing occur in different simulations, as outliers may have a large contribution. The variation in parameter spaces between simulations can be seen in Fig. 1. If χ^2 values become very large, two hypotheses can be presented. First, either the CNN is not powerful enough to output the correct inference from the provided maps, or second, there is not enough of the correct information within the data set to produce a good inference. Distinguishing one hypothesis from the other, along with a physical interpretation of values that deviate from one, is not possible without the ability to interpret deep learning models. Resolving these issues is the focus of our future endeavours.

It is also important to note that the values reported in the subsequent figures correspond to the subset of the data that has been plotted, not the entire set, unless otherwise noted. To achieve such a reduced set, we randomly select a fraction of data points that vary with halo mass – for example, approximately (1/30)th of $\log(M_\mathrm{halo}/\mathrm{M}_\odot) = 11.5$ haloes, but all haloes above $\log(M_\mathrm{halo}/\mathrm{M}_\odot) = 13.0$ are plotted.

3 RESULTS

We present our main results in this section. First, we discuss training and testing on one idealized field at a time for the same simulation (single-simulation analysis), focusing on three inferred properties: (1) halo mass in R_200c (M_halo in Section 3.1), (2) the mass ratio of CGM gas to the total mass inside R_200c [f_cgm in Section 3.2), and (3) the metallicity of the CGM inside 200 kpc ($\log(Z_\mathrm{cgm})$ in Section 3.3]. We do not display the case of a multifield here, as the results do not indicate a significant improvement. Following this, we show the results for the observationally limited case (for properties M_halo and M_cgm), strengthening the motivation for using a multifield (Section 3.4). We also organize network errors (RMSEs) by mass bin (see Table 2) and by simulation (training and testing on the same simulation) for the multifield case with observational limits (Section 3.4.1). Finally, we provide the results of a cross-simulation analysis encompassing all three simulations, with and without observational limits for comparison (Section 3.5).

We utilize Truth–Inference scatter plots to display inferences on the CGM properties. Each plot consists of multiple panels distinguished by field, simulation, or both. The panels visualize the true value, $\mathrm{TRUE_i}$, on the x-axis and the inferred posterior mean, μ_i, on the y-axis, with error bars corresponding to the posterior standard deviation, σ_i. Four statistics (for the subset of data plotted and *not* the entire data set) are also provided for each panel: the RMSE (equation 7), coefficient of determination values (R^2, equation 8), mean relative error (ϵ, equation 9), and reduced χ^2 values (equation 10). The definitions and equations for each are given in Section 2.4. The black diagonal line also represents a 'perfect inference' one-to-one line between the true and inferred values.

3.1 Inferred halo mass

The halo mass emerges as a readily interpretable property, directly deducible from the network, owing to its clear expectations: 'true' high-mass haloes should yield correspondingly high 'inferred' halo masses, regardless of the simulation used for training and testing. Fig. 3 illustrates the Inference–Truth plots for M_halo across all three simulations for a subset of the data. We define the halo mass in equation (1) as the sum of dark matter, gas, and stars within $r < R_\mathrm{200c}$.

The top row corresponds to the results using idealized X-ray maps to infer M_halo, and similarly for the bottom row using H I maps. The columns are ordered by the simulation used for training and testing: IllustrisTNG (left), SIMBA (middle), and Astrid (right). The points are coloured by halo mass throughout. We examine the CNN with input X-ray maps first. In the first panel, we train and test on IllustrisTNG and obtain inferences that indicate a relatively well-constrained monotonic relationship. Some points remain further away from the 'truth' values set by the black diagonal line at the low-mass end, suggesting that X-ray may not be the best probe for these low-mass haloes. The next panel visualizes the training and testing results on SIMBA, with a slight improvement in the higher mass range and an overall relatively well-constrained, monotonic trend with a few outliers that stray far from the black line. This is exactly what is expected and is the same across simulations and fields. There are a few more outliers than IllustrisTNG and slightly larger error bars across the entire mass range. Finally, the third panel demonstrates that the CNN trained on and tested with Astrid has excellent inference power, as indicated by smaller error bars throughout the mass range.

We can now look at the results obtained using the H I input maps. In the first panel, with training and testing on IllustrisTNG, we obtain a clear and well-constrained monotonic relationship with relatively little scatter. There is a slight improvement in error predictions when using H I instead of X-ray, as indicated by the change in the χ^2 value from 0.918 with X-ray to 0.938 with H I. Visually, we can see the improvement in the lower mass range, as there is less scatter. The middle panel shows training and testing on SIMBA, where the inference made is significantly worse than it is with X-ray throughout the entire parameter space, especially intermediate to low masses with increased scatter and larger error bars. The last panel shows training and testing on Astrid. Training and testing the CNN on Astrid with H I input maps yields the best inference of the three galaxy formation models, indicated by the highest R^2 and lowest ϵ. However, the inference made with H I is outperformed by that made with X-ray in the intermediate- to high-mass range.

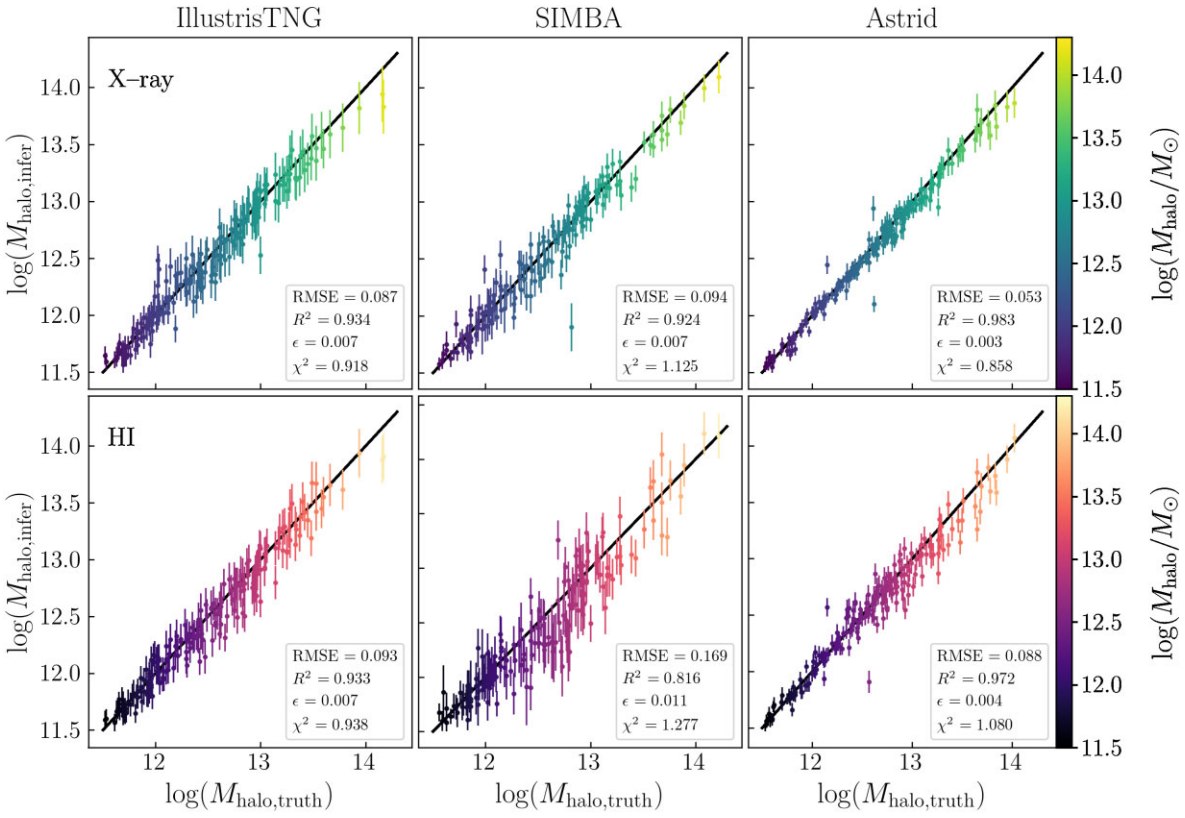

Figure 3. The Truth–Inference plots for $M_{\rm halo}$ when training and testing on the same simulation using idealized single-field data. IllustrisTNG, SIMBA, and Astrid are shown from left to right, and X-ray and H I are shown in the upper and lower rows, respectively. The data are at $z = 0.0$. We plot a mass-dependent fractional sample of haloes from the testing set.

Quantitatively, Astrid provides the most accurate and precise inference for both fields following the RMSE and ϵ values, respectively. It also has the highest R^2 score, indicating that a CNN trained and tested on Astrid can best explain the variability in the data. SIMBA has the lowest R^2 value overall with H I input maps, making it the least accurate in this case. Investigating the χ^2 values, CNNs training and testing on (1) IllustrisTNG consistently overestimate the error ($\chi^2 < 1$), (2) SIMBA consistently underestimate the error ($\chi^2 > 1$), and (3) Astrid overestimate the X-ray error to a greater degree than IllustrisTNG but slightly underestimate the H I error to a lesser degree than SIMBA.

To interpret the meaning of the χ^2 values reported in Fig. 3, we determine the percentage of σ_i errors of individual data points that overlap the line of perfect inference. If the errors are truly capturing the Gaussian behaviour, as in $\chi^2 = 1$, we would expect $1 - \sigma$ or 68 per cent errors to overlap. Briefly, we find that the percentage of overlapping points is 78.3 per cent for the overall lowest $\chi^2 = 0.858$ for Astrid on idealized X-ray maps, and 65.0 per cent for the overall highest $\chi^2 = 1.277$ for SIMBA for idealized H I maps. In the case of SIMBA X-ray, we find an overlapping percentage of 68.9 per cent for $\chi^2 = 1.125$, which indicates a slight non-Gaussian behaviour for a χ^2 just under one. Note that data points with underestimated errors generally overcontribute, especially this χ^2 value. However, it is encouraging to see that the χ^2 values scale as expected, meaning that they have a diagnostic value, but that the inferred errors do not completely follow Gaussian statistics.

3.2 Inferred CGM gas fraction

Fig. 4 shows the Truth–Inference plots for $f_{\rm cgm}$ in the same format as Fig. 3, with the colour bar still indicating $M_{\rm halo}$. We see that $f_{\rm cgm}$ does not have a monotonic trend, seen explicitly in Fig. 1. Higher masses tend to be more constrained, illustrated by a less deviation from the black line and smaller errors than those of lower mass haloes. However, this is likely due to having fewer higher mass haloes for the network to learn from. We define $f_{\rm cgm}$ in equation (2), as the sum of non-star-forming gas within a radius of $r < 200$ kpc divided by the halo mass.

CNN performs poorly with IllustrisTNG on idealized X-ray maps, resulting in scattered points with large error bars. The network underestimates the error bars, as indicated by a χ^2 value greater than one. The next panel shows the results with SIMBA, for which there is better agreement and less scatter toward the higher and intermediate halo masses. However, for the low-mass haloes, there is no distinctive trend, though the network can predict the values well overall but with somewhat large error bars (also underestimated). SIMBA also has slightly lower $f_{\rm cgm}$ values than IllustrisTNG (c.f. Fig. 1). Finally, a CNN with Astrid provides excellent inference for $f_{\rm cgm}$ and accurately estimates the network error. The values are systematically larger, matching Fig. 1.

Similarly, we display H I in the bottom row, with overall trends matching those seen with X-ray. However, H I offers tighter constraints at lower mass haloes (higher $f_{\rm cgm}$ values). This indicates that H I is a slightly better probe for $f_{\rm cgm}$ than X-ray. Interestingly, SIMBA

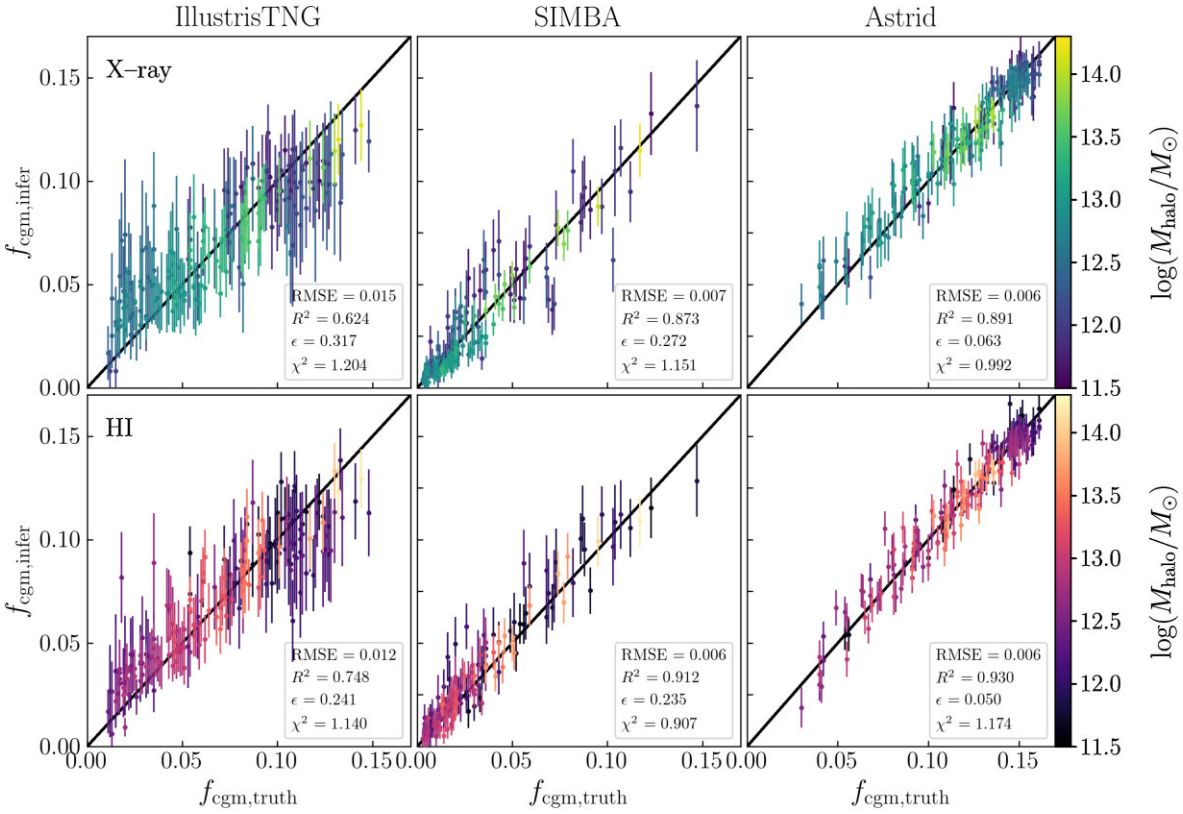

Figure 4. The Truth–Inference plots for $f_{\rm cgm}$, with idealized X-ray (top) and idealized H I (bottom), where the colour bar still represents halo mass. Astrid performs the best with the tightest constraints and the smallest errors, while IllustrisTNG performs the worst, likely due to the sharp increase of $f_{\rm cgm}$ at low mass.

now overestimates the network error (χ^2 value less than one), while IllustrisTNG and Astrid underestimate the errors. In general, $f_{\rm cgm}$ performs worse than $M_{\rm halo}$, but CNN is learning to infer this property using a single idealized field.

It does not appear that the quality of inference by the CNN depends on where the range of $f_{\rm cgm}$ lies with respect to the entire value space spanned by all three simulations – IllustrisTNG returns the worst performance but has intermediate $f_{\rm cgm}$ values, with an underestimate of the error. Astrid yields the most accurate and precise inferences for X-ray and H I fields, with lower scatter and error values for predicting $M_{\rm halo}$ compared to IllustrisTNG and SIMBA. While SIMBA generally performs worse, it exhibits relatively good results in this case, especially with H I.

Following the results of Davies et al. (2020) using IllustrisTNG-100, we see similar non-monotonic trends using CAMELS-IllustrisTNG in $f_{\rm cgm}$ as a function of halo mass. Low-mass haloes ($\log(M_{\rm halo}/{\rm M}_\odot) < 12$) show high $f_{\rm cgm}$ values. When the halo mass is slightly increased, there is a decline in the values of $f_{\rm cgm}$, until approximately $\log(M_{\rm halo}/{\rm M}_\odot) \approx 12.5$ as a threshold mass, after which the monotonic trend with the halo mass returns. Star-forming feedback processes below this threshold mass are dominant and incapable of clearing the CGM. At the threshold mass, these star-forming feedback processes become stronger. Instead of learning the CGM of its gas, the AGN feedback is shut down as early black hole formation is limited (Delgado et al. 2023). We then see a dramatic increase due to turning on jet-mode feedback. Even for cluster-mass objects, AGN feedback cannot overcome deep potential wells, so that we again see high values of $f_{\rm cgm}$. SIMBA has the strongest feedback implementation of the galaxy formation models considered, resulting in lower overall $f_{\rm cgm}$ values throughout the entire mass range. Additionally, we note that although SIMBA has the largest scatter in $f_{\rm cgm}$, this is not simply a reflection of larger statistical fluctuations, as it has a comparable amount of sub-$L*$ objects to IllustrisTNG (see Table 2). Astrid has the weakest feedback, resulting in higher overall $f_{\rm cgm}$ values across the mass range Ni et al. (2023). Further analysis is needed to more concretely establish the relationships between feedback and halo mass such that these results are robust to observational data.

3.3 Inferred metallicity

Fig. 5 shows the truth–inference plots for metallicity, plotted as the logarithm of the absolute value of Z (note $\log(Z_\odot) = 1.87$, Asplund et al. (2009) on this scale. Metallicity presents an interesting challenge to our CNN, as there are often $\sim$1 dex of scatter in Z at the same halo mass with no obvious trend (see Fig. 1). When training and testing on IllustrisTNG (top left), we see that higher mass haloes are slightly better constrained than low-mass haloes, which are more scattered and have larger (and overall underestimated) error bars. We define the metallicity of the CGM in equation (3) as the sum of the metallicity of the gas particles times the mass of non-star-forming gas within a radius of $r < 200$kpc.

Training and testing on SIMBA results in significant scatter across the entire mass range, with larger and underestimated error bars. L^* and group haloes have higher metallicity values overall than in the previous panel. The last panel shows training and testing with Astrid, returning the best overall inference in $\log(Z_{\rm cgm})$ across the entire mass range. Although the error is underestimated, Astrid has much

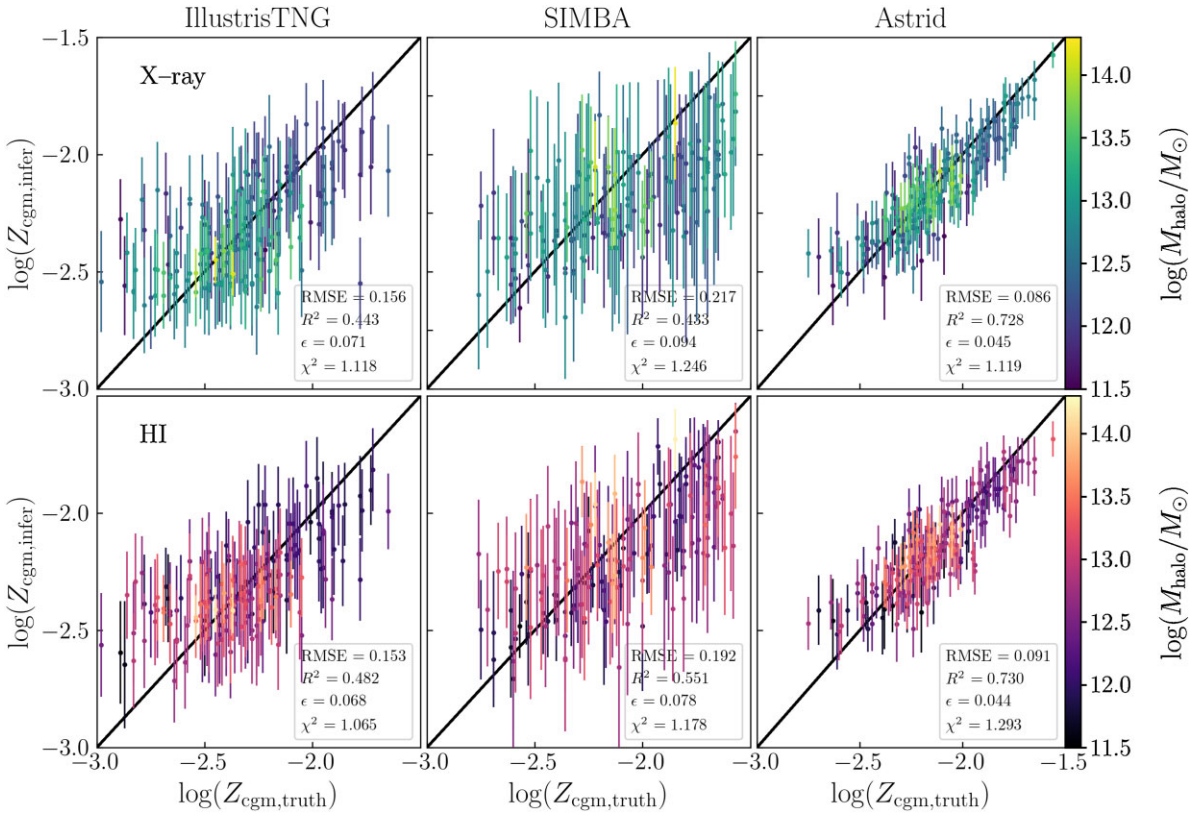

Figure 5. The Truth–Inference plots for metallicity, with idealized X-ray (top) and idealized H I (bottom), where the colour bar still represents halo mass. Astrid performs the best, while SIMBA performs the worst, as it has the most varied Z-values across the mass range, while Astrid has the most confined values.

higher accuracy and precision based on RMSE, R^2, and ϵ values. We argue that this is quite an impressive demonstration of our CNN's ability to predict a value with significant scatter at a single halo mass.

The bottom row illustrates this same inference, but now using H I, where we see similar trends as with X-ray, though slightly more constrained in the case of IllustrisTNG and SIMBA and slightly less constrained in the case of the low-mass end of Astrid. The same upward shift for L^* and group haloes is seen with SIMBA, alluding to SIMBA's strong astrophysical feedback prescriptions that impact higher mass haloes. This is also seen in the changes in lower (higher) χ^2 values for IllustrisTNG and SIMBA (Astrid). We conclude that neither X-ray nor H I is powerful enough on its own to infer log ($Z_{\rm cgm}$). Surprisingly, the entire metallicity of the CGM can be well inferred using H I, especially in the case of Astrid, despite being a small fraction of overall hydrogen, which itself is a primordial element. We do not attempt to provide a physical interpretation of the metallicity of the CGM, as it is quite complex and will be a good topic to focus on for our future work applying interpretative deep learning techniques.

3.4 Observational limits and multifield constraints

Simulations must consider the limitations of current and future observational multiwavelength surveys, such that a one-to-one correlation between them and the developing models can exist. The specific limits used in this work come from the *eROSITA* eRASS:4 X-ray luminosity of 2×10^{-13} erg s^{-1} cm^{-2} arcmin^{-2}, and the typical radio telescope column density for measurements of H I as 10^{19} cm^{-2}. Again, we include the RMSE values R^2, ϵ, and χ^2, which are especially important to distinguish between single and multifield inferences.

The top row of Fig. 6 displays the Truth–Inference plots, highlighting the power of using multiple fields to infer $M_{\rm halo}$ by training and testing a CNN with the IllustrisTNG observationally limited data sets. Utilizing the X-ray (top left), it is clear that we cannot make an inference towards lower halo masses (Sub-$L*$). This is expected, given the *eROSITA*-inspired limits, which show X-ray emission strongly correlating with the halo mass, following Fig. A1. The inability to make a clear inference in this mass regime despite providing the CNN with the most information (nearly 3500 separate Sub-$L*$ haloes, see Table 2) reiterates the weaknesses of X-ray. The X-ray inference improves in the $L*$ range, but is still highly scattered. Chadayammuri et al. (2022) targeted $L*$ galaxies by stacking *eROSITA* haloes and found a weak signal, which appears to be supported by the assessment here. The groups provide much better inference for $M_{\rm halo}$ since these objects should be easily detectable via *eROSITA*. In the middle panel, we explore H I with observational limits to infer $M_{\rm halo}$. Interestingly, H I does a much better job for sub-$L*$ haloes, as these are robustly detected in the 21-cm mapping (see Fig. A1). The inference worsens for $L*$ haloes and for much of the Group range. H I thus far shows improvements via a lower RMSE value, a R^2 value closer to 1, and a lower ϵ value. It also indicates that the network predicts a greater error underestimation due to a higher χ^2 value.

As neither X-ray nor H I is robust enough to infer $M_{\rm halo}$ alone properly, we now train and test the network on combined H I + X-ray 'multifield'. The multifield approach is specifically used when one field alone may not be enough to constrain a property fully or only constrain a property within a certain range of values. The secondary or tertiary fields would then be able to fill in some gaps or tighten the constraints within the inference. Additionally, with

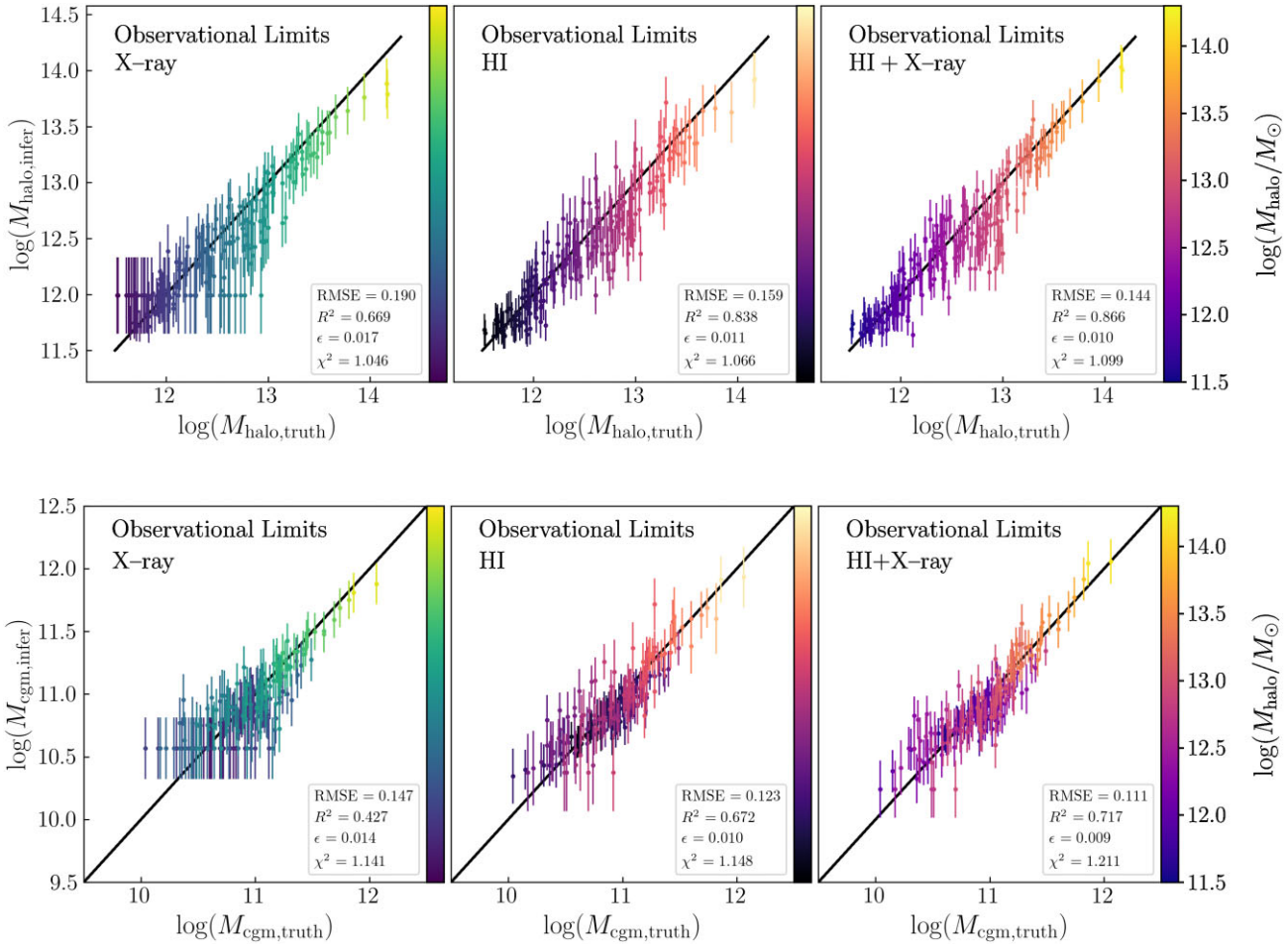

Figure 6. Truth–Inference figures for X-ray (left), H I (middle), and H I + X-ray (right) for IllustrisTNG with observational limits imposed on $M_{\rm halo}$ (*top row*) and $M_{\rm cgm}$ (*bottom row*) using IllustrisTNG. X-ray provides poor inference, especially for lower-mass galaxies, as there are very few, sometimes no emission lines detected if they are too faint. On the other hand, the inference produced from H I results in more uniform errors throughout the mass range, since H I is detected around both low- and high-mass haloes. Combined with their observational limits, the inference is enhanced by tighter constraints at all mass scales.

the ability to adjust the network based on current observations, we form computational counterparts to future surveys to aid in its construction. We achieve stronger constraints throughout the entire mass range, even with observational limits from both X-ray and H I. X-ray probes the $L*$ and Group mass range well, while H I probes the sub-$L*$ mass well, alleviating the previously unresolved noise of the left panel. We see a quantitative improvement in the multifield approach in lower values for RMSE, R^2, and ϵ, which comes at the price of increased underestimation of errors seen with a slight increase in χ^2 value.

The bottom row of Fig. 6 provides similar results and trends for $M_{\rm cgm}$ (defined in equation 4) via IllustrisTNG with X-ray, H I, and multifield using observational limits. X-ray here is also not powerful enough as a probe to infer this property, especially in the low halo mass region. We then look at H I, where there is a better overall inference in the low halo-mass region. However, H I produces more scatter towards the high halo masses than X-ray. The last panel displays results from the H I + X-ray multifield, which is an overall improvement compared to either field alone. The constraints are tighter overall, and the scatter is reduced, as seen in the RMSE values, R^2, ϵ, and χ^2. Additional truth–inference multifield plots with observational limits for the remaining CGM properties can be found in the Appendix B.

3.4.1 Visualizing the CNN error

To quantify the CNN error across all six CGM properties [$M_{\rm halo}$, $f_{\rm cgm}$, $\log(Z_{\rm cgm})$, $M_{\rm cgm}$, $f_{\rm cool}$, and $\log(T_{\rm cgm})$], we plot the error in each property binned by the halo mass. In the left panel of Fig. 7, we plot the error (neural network error, or mean relative error) for each property when considering the observational limits on H I, X-ray, and multifield H I + X-ray for a CNN that is trained and tested on IllustrisTNG. Panels are separated by halo mass, where we use the full data set instead of the subset in the truth–inference plots.

We outline the general trends of this figure and point out interesting features. In the sub-$L*$ panel, X-ray maps alone provide the highest error, followed by multifield, and then H I with the lowest error, to be expected. Note that there is an infinitesimal difference between multifield and using H I alone. In the second panel ($L*$), the margin of error between X-ray and H I is decreasing, meaning that X-ray is becoming increasingly more important in the intermediate halo mass range. Multifield development is also strictly improving with the use of H I alone. With Groups, the multifield offers a greater improvement over either field alone, except for $\log(M_{\rm cgm})$ where the X-ray has a slightly lower error.

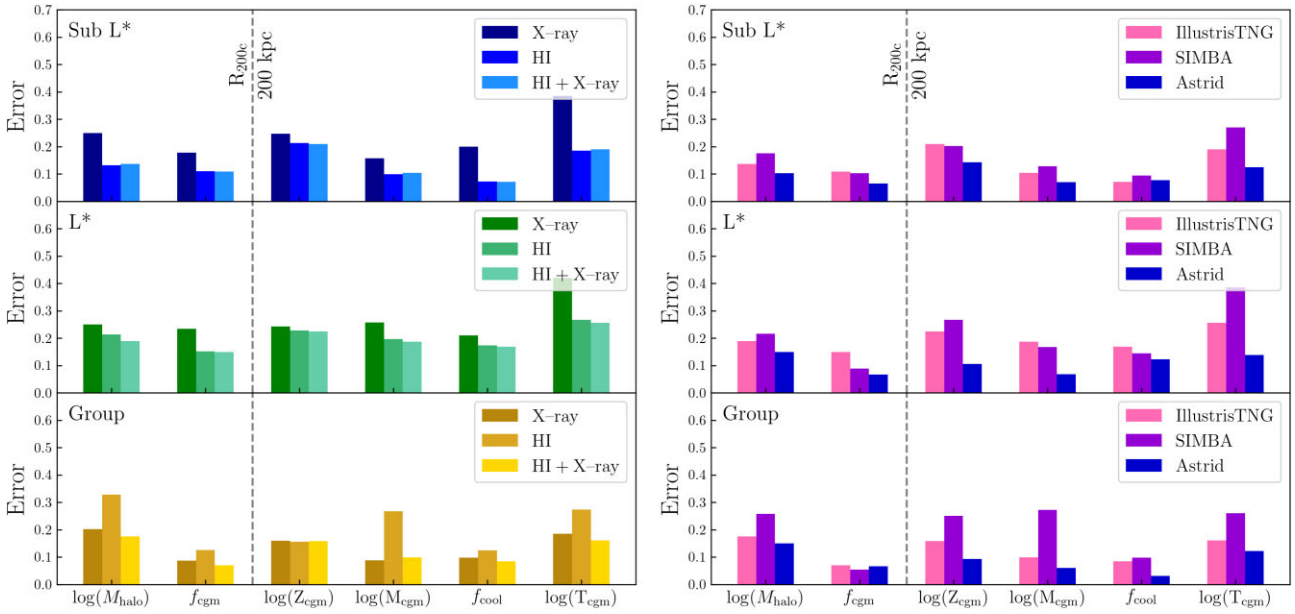

Figure 7. *Left:* Average RMSE values split by halo category for training and testing on IllustrisTNG, with fields X-ray, H I, and the multifield H I + X-ray with observational limits, for all six properties. These bars are representative of the *full* data set. We provide a dashed vertical line to distinguish between properties that are radially bound by R_{200c} and those by 200 kpc. *Right:* Average RMSE values split by simulation (training and testing on IllustrisTNG, SIMBA, or Astrid), with H I + X-ray and observational limits, for all six properties. These bars are representative of the *full* data set. Neither panel is entirely comparable to the Truth–Inference plots, as these categorize errors by halo mass and are for the full data set.

Focusing on f_{cgm}, the errors are generally smaller than those of M_{halo}, but this may reflect the quantity range that is inferred as f_{cgm} is mainly between 0.0 and 0.16 while M_{halo} varies between 11.5 and 14.3. Meanwhile, log (Z_{cgm}) has similar error levels between H I and X-ray, with a small improvement for multifield for sub-$L*$ and $L*$. The errors in log (Z_{cgm}) vary between 0.16 and 0.24, so measuring metallicity at this level of accuracy is promising, but distinguishing high values of metallicity from low ones is disappointing for IllustrisTNG (see Fig. B2).

The last three sets of properties, log (M_{cgm}), f_{cool}, and log (T), have not been previously illustrated as Truth–Inference plots. They depict similar trends and show general multifield improvement. H I infers sub-$L*$ the best, while X-ray infers Groups the best. The multifield is most important for $L*$ haloes, and across all six properties, there is a significant improvement in the inference. Other halo categories usually do not result in as much improvement; in some cases, the multifield performs slightly worse. We note that inference of f_{cool} for groups is a significant improvement, from 0.102 (X-ray) and 0.125 (H I) to 0.084 (multifield), reflecting that CNN integrates observations of both cool gas (H I) and hot gas (X-ray) in this fraction.

The right panel of Fig. 7 outlines the errors in IllustrisTNG, SIMBA, and Astrid for the multifield H I + X-ray with observational limits for all six properties. The halo mass again separates the three panels. Generally, a CNN trained and tested on SIMBA has the highest error over the entire mass range, while a CNN trained and tested on Astrid returns a better inference. f_{cgm} breaks this trend, as it is significantly worse for L* mass haloes when using IllustrisTNG, which is directly due to the drastic inflection point seen in Fig. 1. Additionally, Astrid can infer log (Z_{cgm}) remarkably well for $L*$ mass haloes, compared to the high error when using IllustrisTNG. This can be seen in Fig. B2 where IllustrisTNG has much more scatter across the entire mass range, while Astrid shows little scatter.

3.5 Cross simulation inference

Until now, each truth–inference plot has been created by training and testing on the same simulation. In this section, we provide the results obtained when training on one simulation or galaxy formation model and testing on another to prove the degree of robustness across any particular simulation. We do this for both an X-ray with observational limits and a multifield with observational limits.

In Fig. 8, we demonstrate the cross-simulation inference between IllustrisTNG, SIMBA, and Astrid, using X-ray with observational limits only on the M_{halo} property. The diagonal plots correspond to the training and testing in IllustrisTNG, SIMBA, and Astrid from upper left to lower right (repeated from the upper panels of Fig. 3).

The top row refers to CNNs trained on IllustrisTNG, where each panel from left to right has been tested on IllustrisTNG, SIMBA, and Astrid, respectively. When tested on SIMBA, most points are close to the black line, but with significantly more scatter. When tested on Astrid, we can only recover good constraints for the high-halo mass range. There is much more scatter in the low-mass range, as a majority of them are overestimated, except for a few outliers, most likely resulting from the inability of X-rays to probe the low-halo mass range.

When training on SIMBA, but then testing on IllustrisTNG, there is still quite a bit of scatter in the low-mass haloes, and the high-mass haloes are now overestimated. This matches the expectations from the brightness differences between IllustrisTNG (brighter) and SIMBA (dimmer). When testing on Astrid, all points are shifted up and overestimate halo mass.

Finally, training on Astrid and testing on IllustrisTNG cannot recover any of the results. There is a lot of scatter for the low-halo-mass range with large error bars, with points that do not follow the expected trends in IllustrisTNG for intermediate and high masses. Astrid underestimates the majority of the halo masses. When testing on SIMBA, the results cannot be recovered either, as most

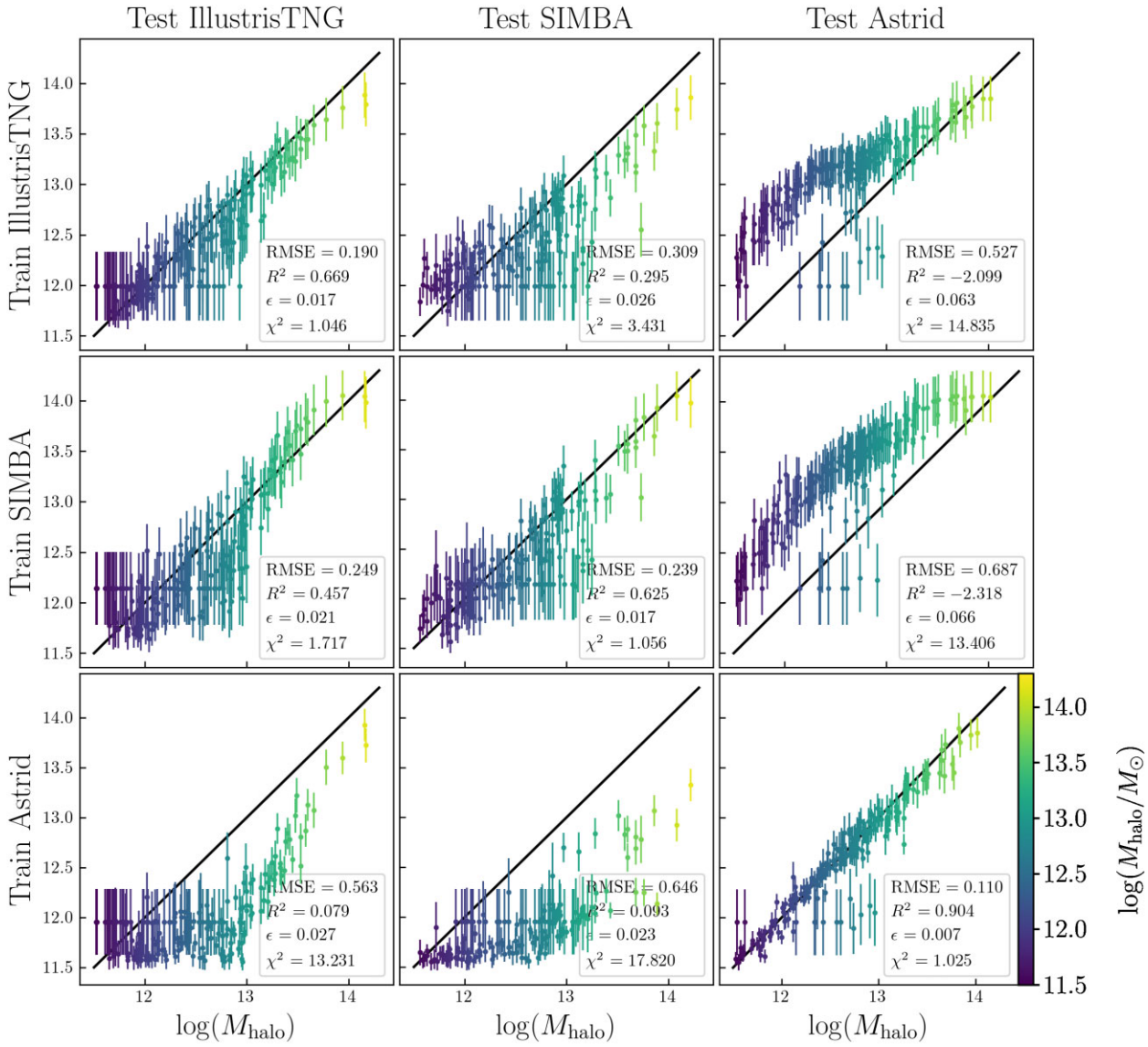

Figure 8. Cross-simulation results for IllustrisTNG, SIMBA, and Astrid on X-ray for $M_{\rm halo}$, with observational limits. The x-axis of each panel corresponds to the true values of $M_{\rm halo}$, and the y-axis corresponds to the inference values of $M_{\rm halo}$, as before. The y-axis labels indicate that the panels in the top row were trained on IllustrisTNG, the middle row on SIMBA, and the bottom row by Astrid. The columns are labelled such that the panels in the first column were tested on IllustrisTNG, the second column's panels on SIMBA, and the third on Astrid. The diagonal panels are the result of training and testing on the same simulation. Training and testing on Astrid provide the tightest constraints and the best inference. These points are a fraction of the full data set.

points underestimate the halo mass. Although training and testing on Astrid seem to provide the best constraints on halo mass with X-ray observational limits, it is the least robust simulation out of the three, as measured by its ability to be applied to other simulations as a training set. In contrast, other models trained on the Astrid LH set (Ni et al. 2023; de Santi et al. 2023) are the most robust, as the parameter variations produce the widest variation in galaxy properties, in turn making ML models more robust to changes in baryonic physics. IllustrisTNG is the most robust in this case, as it returns the results of the other two simulations with the least amount of scatter.

One oddity in the statistical measurements produced comes from training on either IllustrisTNG or SIMBA and testing on Astrid, which results in a negative R^2 value, indicating a significant mismatch in the models. Another unusual statistic is in the extremely high χ^2 values from three cases: (1) training on IllustrisTNG and testing on Astrid, (2) training on SIMBA and testing on Astrid, and (3) training on Astrid and testing on either IllustrisTNG or SIMBA. Each reiterates the lack of robust results that can be achieved with Astrid.

Fig. 9 illustrates the cross-simulation results on $M_{\rm halo}$ with observational limits on the multifield H I + X-ray for IllustrisTNG, SIMBA, and Astrid. The top left panel shows this multifield, trained on and tested with IllustrisTNG, where overall, $M_{\rm halo}$ can somewhat be constrained throughout the entire parameter space. The second panel on the diagonal corresponds to the same multifield but is now trained on and tested with SIMBA. The constraints here are weaker throughout the entire parameter space as there is more overall scatter, though the trend is the same as expected. The last panel on the diagonal shows the network trained on and tested with Astrid, where we can obtain the tightest constraints overall, especially in the higher halo mass range. The few outliers towards the mid ($L*$) to low (Sub-$L*$) mass range with larger error bars may need further investigation.

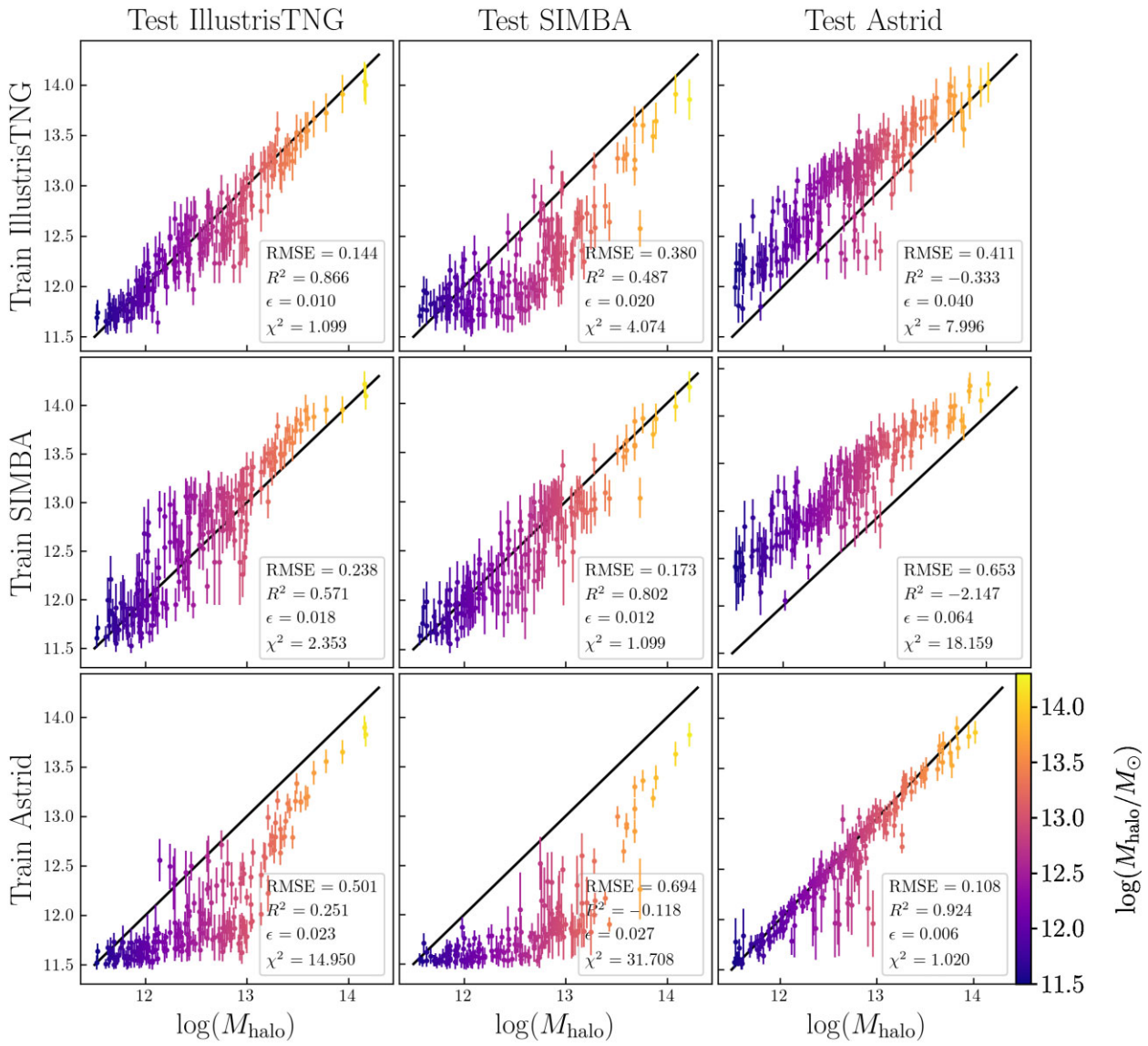

Figure 9. Cross-simulation results to infer $M_{\rm halo}$ using the multifield with observational limits for IllustrisTNG, SIMBA, and Astrid. The layout is the same as in Fig. 8. Even with the observational limits of H I and X-ray, training and testing on Astrid have the best overall inference for $M_{\rm halo}$.

The top row shows training with IllustrisTNG and testing on IllustrisTNG, SIMBA, and Astrid, respectively. When training on IllustrisTNG and testing on SIMBA, we expect that for a given mass halo in IllustrisTNG, that same halo will look dimmer and, therefore, less massive in SIMBA. This is seen here, as most haloes are below the black line. When the network is now tested on Astrid, a similar but opposite expectation is met. With the knowledge that for a given halo mass in IllustrisTNG, that same halo will look brighter and, therefore, more massive in Astrid, this trend also makes sense, as we see a large majority of the points shifted above the black line. We can conclude that with observational limits in the multifield, training on IllustrisTNG can return the trends in SIMBA and Astrid, but there is an offset in recovered $M_{\rm halo}$ explainable by the shift in observables.

The middle row shows training with SIMBA and testing on IllustrisTNG, SIMBA, and Astrid, respectively. When the network trains on IllustrisTNG, it can recover the inference and achieve good constraints. The same halo in SIMBA will appear brighter in IllustrisTNG, so the shift in most points upward above the black line is, therefore, as expected. When testing on Astrid, we still recover the inference and achieve good constraints, but we see the same shift as we saw when training on IllustrisTNG and testing on Astrid. This also aligns with the expectations, as the haloes in Astrid will seem much brighter than those in SIMBA. We can conclude that with observational limits in the multifield, SIMBA is also robust enough to recover inference and constraints for $M_{\rm halo}$.

The bottom row shows training with Astrid and testing with IllustrisTNG, SIMBA, and Astrid, respectively. When the network tests on IllustrisTNG, we can recover the general trend with slightly less strong constraints. We can recover the general trend with slightly less strong constraints when the network is tested on SIMBA. The haloes in Astrid will be brighter than the same haloes in IllustrisTNG and SIMBA, so the majority of the points are below the black line when testing on IllustrisTNG and SIMBA. We can conclude that a CNN trained on Astrid cannot recover the inference and constraints for $M_{\rm halo}$. We see the same statistical nuances as in the

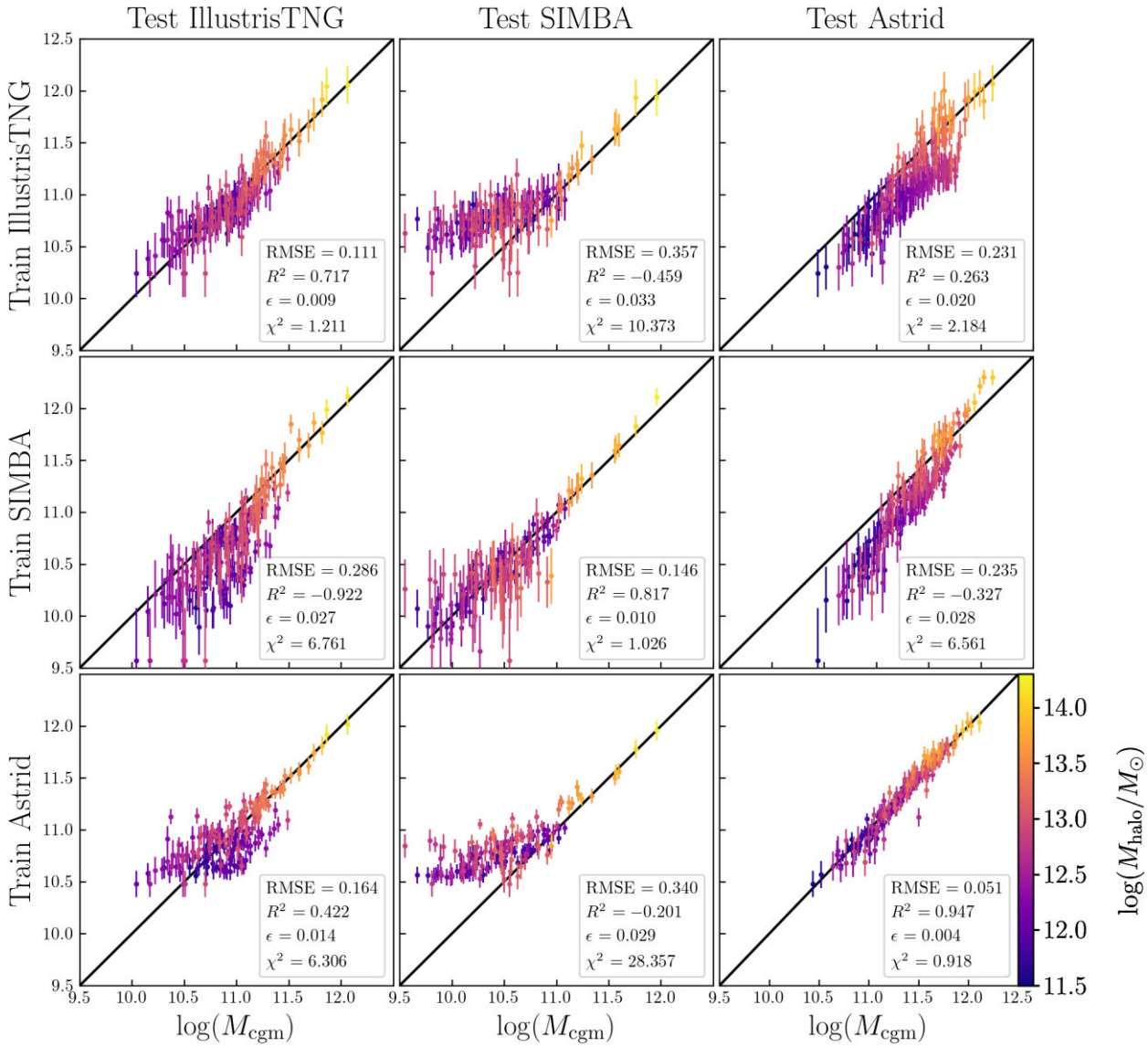

Figure 10. Cross-simulation results to infer $M_{\rm cgm}$ using the multifield with observational limits for IllustrisTNG, SIMBA, and Astrid. The layout is the same as in Fig. 8.

previous figure: negative R^2 values and large χ^2 values in the same configurations.

By adding observational constraints for both H I and X-rays, the simulations gain a further level of similarity, which enhances their constraining power in the cross-simulation analysis. Fig. 10 shows the results of using the multifield (H I + X-ray) approach with observational limits on $M_{\rm cgm}$, with observational limits. The layout of the plot is analogous to that of Fig. 9. Training on IllustrisTNG (top row) overpredicts the results for intermediate- and low-mass haloes when testing on SIMBA and underpredicts the same results when testing on Astrid. This aligns with the expectations in the bottom left panel of Fig. 1, which describes the relationship between the halo mass and $M_{\rm cgm}$. Training on SIMBA (middle row) underpredicts intermediate- and low-mass haloes results when testing on IllustrisTNG and Astrid. Note that there is much more scatter when testing on IllustrisTNG, especially for objects with low $M_{\rm cgm}$ values. Training on Astrid (bottom row) does reasonably well when testing on IllustrisTNG with some scatter in the intermediate- and low-mass haloes. However, it overpredicts these intermediate- and low-mass haloes when tested on SIMBA.

Although able to return similar trends, cross-simulation training and testing display offsets related to different CGM properties in all simulations. However, it is enlightening to see that cross-simulation inference improves when more bands are included, which indicates that broad properties like $M_{\rm halo}$ and $M_{\rm cgm}$ are more robustly characterized by observing in multiple bands. We make a deliberate choice to show the cross-simulation analysis results for $M_{\rm halo}$ and $M_{\rm cgm}$, not $f_{\rm cgm}$, as it is a ratio of the gas mass to the halo mass throughout the halo (not within 200 kpc), leading to a more complex trend that is not as easily interpretable. Cross-simulation analysis can offer a way to understand the direction and magnitude of systematic offsets and the variations between feedback implementations qualitatively and the feedback energy as a function of redshift. This is entirely contingent on our ability to create physically motivated deep-learning models that are interpretable, which is the focus of our future work.

4 DISCUSSIONS

In this section, we discuss the interpretation of cross-simulation analysis (Section 4.1), assess the applications and limitations of CNNs when applied to CGM (Section 4.2), compare the variance between true and inferred values for $\log(M_{\rm halo})$, $\log(M_{\rm cgm})$, and $\log(Z_{\rm cgm})$ using the idealized multifield maps (Section 4.3), and expand on an intriguing direction for future work (Section 4.4).

4.1 Cross-simulation interpretability

In Section 3.5, we explore the robustness of simulations by examining cross-simulation inference with and without observational limits. Fig. 9 presents cross-simulation inferences for multifield H I + X-ray with observational limits on $M_{\rm halo}$. Upon initial inspection, training, and testing on Astrid offer the tightest constraints across the entire mass bin. In general, a test simulation will overpredict (underpredict) properties when trained on a simulation with CGM observables that are dimmer (brighter). Among the three simulations, a CNN trained on IllustrisTNG is the most robust, as it accurately captures the differences between halo mass measurements when trained on SIMBA and Astrid. However, more work must be done to show that a CNN trained on IllustrisTNG will produce the most robust predictions when applied to real observational data. A novel aspect to further explore is training and testing on multiple simulations, varying the feedback parameters such that the CNN would marginalize over the uncertainties in baryonic physics.

The effort to train and test on different simulations mimics training on a simulation and predicting real observational data. Although it is disappointing to see such deviations in the results of the cross-simulation analysis, we know that some simulations offer better representations given the specific scope of this work than others. Using observational limits that resemble the ranges of detection of current instruments as a simulation constraint, we can begin directly comparing simulations and observations. We note that the simulations are unconstrained by available observations in the CGM. The fiducial prescriptions for IllustrisTNG and SIMBA are calibrated to match the available data of the groups with varying success (Oppenheimer et al. 2021), but Astrid with its higher $f_{\rm gas}$ values has not been calibrated similarly. Importantly, no simulation is a perfect representation of the real universe, but it is crucial to develop CNNs that can adapt to a wide range of mock haloes generated using multiple galaxy formation codes that aim to simulate these systems with realistic physical prescriptions.

Robustness quantification, or how well a network trained on one simulation can infer a given quantity when tested on another simulation within any set of simulations and machine learning algorithms, including the CAMELS suites, is crucial to further their development (Villaescusa-Navarro et al. 2021b; Villanueva-Domingo & Villaescusa-Navarro 2022; Echeverri et al. 2023; de Santi et al. 2023). The lack of robustness can be due to either (1) differences between simulations, (2) networks learning from numerical effects or artefacts, or (3) lack of overlapping between simulations in the high-dimensional parameter space. These reasons are not surprising, because of the use of the CV set within CAMELS, and there could be slight variations in feedback that are unaccounted for. Using the LH set instead would improve the results obtained in this work. Additionally, precision (smaller error bars) without accuracy (recovering the 'true' values) is meaningless. Therefore, although Astrid generally has the smallest error bars, this alone shows strong biases when tested on other models. Future work can be done to address the inability to obtain robust constraints while performing cross-simulation analysis. One avenue is through domain adaptation (Ganin et al. 2015), which allows a smoother transition between training and testing on different simulations such that we obtain robust results.

4.2 Applicability and limitations of CNNs applied to the CGM

We have applied a CNN following the structural format used by Villaescusa-Navarro et al. (2022) and modified it to infer underlying *properties* of the CGM of individual haloes with fixed cosmology and astrophysics within the CAMELS CV set. The former CNN infers six independent parameters (two cosmological and four astrophysical feedback) *by the design of the LH simulation set*. Our trained CGM CNN learns to predict properties with high co-dependencies (e.g. $\log(M_{\rm halo})$ and $\log(T_{\rm cgm})$) and related quantities ($f_{\rm cgm}$ and $M_{\rm cgm}$). In the latter case, there are two different ways to quantify CGM mass in two distinct apertures– $M_{\rm cgm}$ is the CGM mass inside 200 kpc, and $f_{\rm cgm}$ is the mass of CGM over the total mass inside $R_{\rm 200c}$.

We attempted to infer one property at a time instead of all six and found only a marginal improvement. CNN implemented in this work, classified as a moment network (Jeffrey & Wandelt 2020), has the flexibility to infer multiple properties simultaneously, but requires a rigorous hyperparameter search, as detailed in Section 2.3.

A concern that often appears with any simulation-based approach is the possibility of biases seeping into the result, generally due to incomplete modelling of physical processes. We aim to alleviate this concern first by using the CV set within the CAMELS simulations, where the values of cosmological and astrophysical feedback parameters are fixed to their fiducial values. The LH set, which was not used in this work (but could easily be integrated as part of future efforts), increases the chances of successful cross-simulation analysis as the astrophysical dependencies are completely marginalized. From this standpoint, the CV set is not best suited to produce robust cross-simulation analysis. Using the CV set, we gain valuable insight into the distinctions among simulations and their effects on the results of the CGM properties in this study. In addition to using the LH set, we can explore training and testing on more than one simulation or performing a similar analysis on the broader parameter space of TNG-SB28 (Ni et al. 2023).

We apply CNNs to the CGM data sets to (1) determine the degree to which physical properties of the CGM can be inferred given a combination of fields and simulations, and (2) examine different observing strategies to determine how combining different wavebands can infer underlying CGM properties.

We demonstrate the feasibility of applying a CNN to observational data sets and return values and errors for the CGM properties, including $M_{\rm halo}$ and $M_{\rm cgm}$. Additionally, the inference of $M_{\rm halo}$ is more robustly determined when another field, along with its associated observational limits, is added. However, training on one simulation and testing on another support the notion that predictions can produce significantly divergent results compared to the true values, as seen in Figs 8 and 9. As mentioned in Section 4.1, although IllustrisTNG, SIMBA, and Astrid have been tuned to reproduce galaxies' and some gas properties, they make varied predictions for gaseous haloes. In future efforts to improve this work, the LH set would replace the CV set, under the expectation of improvement, as all astrophysics is marginalized. Should this not be the case, domain adaptation is the longer term solution to help bridge the many gaps between different subgrid physics models. Another interesting future direction would include training and testing on combinations of simulations, though this is ideally performed with the LH set.

Table 3. The variance of $M_{\rm cgm}$ and $\log(Z_{\rm cgm})$ compared between the input from CAMELS (truth) and the values from the idealized multifield (H I + X-ray) inference.

$\log(M_{\rm cgm})$	sub-$L*$		$L*$	
	True	Infer	True	Infer
IllustrisTNG	0.024	0.013	0.127	0.074
SIMBA	0.099	0.094	0.129	0.111
Astrid	0.026	0.024	0.038	0.038
$\log(Z_{\rm cgm})$	sub-$L*$		$L*$	
	True	Infer	True	Infer
IllustrisTNG	0.072	0.041	0.078	0.035
SIMBA	0.086	0.060	0.108	0.052
Astrid	0.053	0.033	0.042	0.035

4.3 Multifield variance comparison

As an additional test, we check if our CNN-inferred values can reproduce the original dispersion of a CGM data set. Even if a CNN can reproduce the mean value of a CGM parameter, can it also reproduce the spread of values? Fig. 1 shows the shaded $\pm1\sigma$ dispersions in addition to the medians. We, therefore, calculate the dispersion for $\log(M_{\rm cgm})$ and $\log(Z_{\rm cgm})$ for sub-$L*$ and $L*$ galaxies across the three simulations to explore our CNN's ability to reproduce this scatter in relatively flat $M_{\rm halo}$ bins. The values are displayed in Table 3. On a positive note, it does appear that sub-$L*$ and $L*$ dispersions for $\log(M_{\rm cgm})$ are well reproduced in SIMBA and Astrid. However, the dispersions are severely underestimated, often by a factor of 2, for $\log(Z_{\rm cgm})$ and $\log(M_{\rm cgm})$, but, notably, R^2 measures and the performance of the CNN is poor for these cases. In particular, with IllustrisTNG, $M_{\rm CGM}$, and $f_{\rm CGM}$ as a closely related quantity, show worse performance due to rapidly changing gas fractions in response to feedback, as we discuss in Section 3.2. In this case, the CNN is unable to adequately learn signatures of reduced CGM mass at a fixed halo mass. This test presents a crucial challenge for future machine learning and deep learning methods in reproducing the spread of a given property for objects that are otherwise alike.

4.4 Future work

In expanding the scope of this work to additional wavelengths in the future, we also aim to advance our understanding of where the CNN extracts important information from within a given map. We can use the information gained from this type of analysis, which has not been applied to CGM data before this work, to inform future observational surveys on how best to achieve the greatest scientific returns given wavelength, survey depth, and other specifications. Additionally, this type of analysis will be necessary to determine machine learning verification and validation. To achieve this, we hypothesize that moving towards higher resolution simulations, including IllustrisTNG-100, EAGLE, and others along with a more physically motivated deep-learning model, will have a significant impact across a wide range of scales, especially in the case of observational limits.

5 CONCLUSIONS

In this study, we use CNNs trained and tested on CAMELS simulations based on the IllustrisTNG, SIMBA, and Astrid galaxy formation models to infer six broad-scale properties of the circum-galactic medium (CGM). We focus on the halo mass, the CGM mass, the metallicity, the temperature, and the cool gas fraction. We simulate two observational fields, X-ray and 21-cm H I radio, which can represent the broad temperature range of the CGM. We tested our CNN on data sets with and without (idealized) observational limits. Our key findings include the following.

(i) When training and testing the CNN on the same simulation:

(a) By comparing all the CGM properties the CNN is trained to infer, it performs the best overall on $M_{\rm halo}$ and $M_{\rm cgm}$, both with and without observational limits. For IllustrisTNG with observational limits, the RMSE values returned for $M_{\rm halo}$ are $\sim$0.14 dex, and $M_{\rm cgm}$ are $\sim$0.11 dex when combining X-ray and H I data.

(b) The 'multifield' CNN trained simultaneously on X-ray and H I data with observational limits allows for the best inference across the entire mass range using the same inpts without the discontinuities seen when trained only on one field. Obtaining interpretable inferences on the halo mass for the continuous range of $11.5 \leq \log(M_{\rm halo}/{\rm M}_\odot) \leq 14.5$ requires a multifield, although various combinations may be better over smaller mass bins than others. Sub-L^* haloes ($M_{\rm halo} = 10^{11.5-12}\,{\rm M}_\odot$) are only marginally better inferred with H I than multifield. Moving to L^* haloes ($M_{\rm halo} = 10^{12-13}\,{\rm M}_\odot$) and the more massive groups ($M_{\rm halo} > 10^{13}\,{\rm M}_\odot$), there is a drastic improvement when using multifield over X-ray and H I alone. Our exploration demonstrates that CNN-fed multiple observational fields with detectable signals can continuously improve the inference of CGM properties over a large mass range given the same input maps.

(c) When adding observational limits to the multifield CNN, the inference accuracy declines, but still returns RMSE values indicating success. Recovering total mass from observations appears to be feasible with our CNN. H I mapping is especially critical for recovering CGM properties of sub-$L*$ and $L*$ galaxies.

(ii) For CNN cross-simulation analysis (training on one simulation and testing on another):

(a) When applying cross-simulation analysis by training on one simulation and testing on another, the inferred values generally correlate with the true physical properties. Still, they are frequently offset, indicating strong biases and overall poor statistical performance.

(b) Interestingly, the cross-simulation analysis reveals that using the H I + X-ray multifield with observational limits improves the halo mass inference compared to that from X-ray maps alone. In the process of adding constraints in this case, the difference between the individual simulation parameter spaces becomes smaller and acts as tighter boundary conditions for the network.

Our results have broader implications for applying deep learning algorithms to the CGM than those outlined here. First, performing a cross-simulation analysis and determining that the CNN is robust opens the possibility of replacing one of the simulations with real data to infer the actual physical properties of observed systems. Second, the addition of more wavelengths is easily implemented within image-based neural networks. To continue making connections to current and future multiwavelength surveys, we can expand the number of fields used in this architecture beyond X-ray and H I, including image-based CGM probes like the Dragonfly Telescope that can map the CGM in optical ions, like Hα and N II (Lokhorst et al. 2022), and UV emission from ground- or space-based probes (Johnson et al. 2014; Burchett et al. 2018; Johnson et al. 2018; Péroux & Howk 2020). Most importantly, this method

would allow simulation differences to be marginalized, while still obtaining correlations and constraints. We can overcome the current challenges of cross-simulation analysis by training our CNN on multiple CAMELS simulations and parameter variations existing and in production [including expanding to EAGLE (Schaye et al. 2015), RAMSES (Teyssier 2010), Enzo (Bryan et al. 2014), and Magneticum[5]] while integrating additional wavebands. It is crucial to identify the primary source of information for CNN to increase the number of simulations and wavelengths used as input. Future work includes performing saliency analysis with integrated gradients to determine the most important pixels on a given map. It allows for more targeted and efficient adjustments to improve inferences. This can reveal which underlying physical properties are universally recoverable and robustly predictable in observations.

The CGM demarcates a region of space defined by nebulous boundaries, which poses a unique challenge to traditional analysis techniques like principal component analysis. In addition, there are no established methods to characteristically analyse CGM. The phrase 'characteristically analysing' implies distinctly categorizing entities. For instance, traditional analysis can be used with galaxies to classify them into various categories based on their unique evolutionary traits, as evidenced by Lotz, Primack & Madau (2004). However, the CGM refers to the area surrounding the galactic disc until the accretion shock radius, where neither boundary is precisely defined as they cannot be directly observed. Applying the same traditional analysis approach to a CGM data set would require a rigid pipeline, making it difficult to incorporate new simulations or wavelengths without extensive reconfiguration. Deep learning offers a more flexible and versatile approach as a solution.

ACKNOWLEDGEMENTS

We thank Shy Genel and Matthew Ho for valuable feedback and suggestions for the paper. The CAMELS simulations were performed on the supercomputing facilities of the Flatiron Institute, which is supported by the Simons Foundation. This work is supported by the NSF grant AST 2206055 and the Yale Center for Research Computing facilities and staff. The work of FVN is supported by the Simons Foundation. The CAMELS project is supported by the Simons Foundation and the National Science Foundation (NSF) grant AST 2108078. DAA acknowledges support by NSF grants AST-2009687 and AST-2108944, CXO grant TM2-23006X, Simons Foundation Award CCA-1018464, and Cottrell Scholar Award CS-CSA-2023-028 by the Research Corporation for Science Advancement.

DATA AVAILABILITY

CAMELS data are publicly available at https://camels.readthedocs.io/en/latest/. Original data is available from the authors upon request by emailing naomi.gluck@yale.edu.

REFERENCES

Akiba T., Sano S., Yanase T., Ohta T., Koyama M., 2019, preprint (arXiv:1907.10902)
Anderson M. E., Bregman J. N., 2011, ApJ, 737, 22
Anglés-Alcázar D., Davé R., Faucher-Giguère C.-A., Özel F., Hopkins P. F., 2017a, MNRAS, 464, 2840
Anglés-Alcázar D., Faucher-Giguère C.-A., Kereš D., Hopkins P. F., Quataert E., Murray N., 2017b, MNRAS, 470, 4698
Asplund M., Grevesse N., Sauval A. J., Scott P., 2009, ARA&A, 47, 481
Bird S., Ni Y., Di Matteo T., Croft R., Feng Y., Chen N., 2022, MNRAS, 512, 3703
Bogdán Á., Lovisari L., Volonteri M., Dubois Y., 2018, ApJ, 852, 131
Bregman J. N., Anderson M. E., Miller M. J., Hodges-Kluck E., Dai X., Li J.-T., Li Y., Qu Z., 2018, ApJ, 862, 3
Bryan G. L. et al., 2014, ApJS, 211, 19
Burchett J. N., Tripp T. M., Wang Q. D., Willmer C. N. A., Bowen D. V., Jenkins E. B., 2018, MNRAS, 475, 2067
Bykov A. M., Dolag K., Durret F., 2008, Space Sci. Rev., 134, 119
Chadayammuri U., Bogdán Á., Oppenheimer B. D., Kraft R. P., Forman W. R., Jones C., 2022, ApJ, 936, L15
Christensen C. R., Davé R., Governato F., Pontzen A., Brooks A., Munshi F., Quinn T., Wadsley J., 2016, ApJ, 824, 57
Cooksey K. L., Thom C., Prochaska J. X., Chen H.-W., 2010, ApJ, 708, 868
Davies J. J., Crain R. A., Oppenheimer B. D., Schaye J., 2020, MNRAS, 491, 4462
Davé R., Crain R. A., Stevens A. R. H., Narayanan D., Saintonge A., Catinella B., Cortese L., 2020, MNRAS, 497, 146
Davé R., Anglés-Alcázar D., Narayanan D., Li Q., Rafieferantsoa M. H., Appleby S., 2019, MNRAS, 486, 2827
de Blok W. J. G. et al., 2016, in MeerKAT Science: On the Pathway to the SKA. p. 7, preprint (arXiv:1709.08458)
Delgado A. M. et al., 2023, MNRAS, doi:
de Santi N. S. M. et al., 2023, ApJ, 952, 69
Echeverri N. et al., 2023, ApJ, 954, 125
Feng Y., Bird S., Anderson L., Font-Ribera A., Pedersen C., 2018, MP-Gadget/MP-Gadget: A tag for getting a DOI
Ganin Y., Ustinova E., Ajakan H., Germain P., Larochelle H., Laviolette F., Marchand M., Lempitsky V., 2015, preprint (arXiv:1505.07818)
Gebhardt M. et al., 2023, preprint (arXiv:2307.11832)
Haardt F., Madau P., 2012, ApJ, 746, 125
Hafen Z. et al., 2019, MNRAS, 488, 1248
Hopkins P. F., 2015, MNRAS, 450, 53
Hopkins P. F., 2017, preprint (arXiv:1712.01294)
Hopkins P. F. et al., 2018, MNRAS, 480, 800
Hummels C. B., Smith B. D., Silvia D. W., 2017, ApJ, 847, 59
Jeffrey N., Wandelt B. D., 2020, preprint (arXiv:2011.05991)
Johnson S. D., Chen H.-W., Mulchaey J. S., Tripp T. M., Prochaska J. X., Werk J. K., 2014, MNRAS, 438, 3039
Johnson L. C. et al., 2015, ApJ, 802, 127
Johnson S. D. et al., 2018, ApJ, 869, L1
Johnston S. et al., 2007, PASA, 24, 174
Jonas J., MeerKAT Team, 2016, in MeerKAT Science: On the Pathway to the SKA. p. 1
Kang H., Ryu D., Cen R., Ostriker J. P., 2007, ApJ, 669, 729
Keeney B. A. et al., 2018, ApJS, 237, 11
Keres D., Katz N., Weinberg D. H., Dave R., 2005, MNRAS, 363, 2
Li J.-T., Bregman J. N., Wang Q. D., Crain R. A., Anderson M. E., Zhang S., 2017, ApJS, 233, 20
Li R. et al., 2022, ApJ, 936, 11
Lokhorst D. et al., 2022, ApJ, 927, 136
Lotz J. M., Primack J., Madau P., 2004, AJ, 128, 163
Mathur S., Das S., Gupta A., Krongold Y., 2023, MNRAS, 525, L11
Nelson D. et al., 2018, MNRAS, 475, 624
Nelson D. et al., 2019, CompAC, 6, 2
Ni Y. et al., 2022, MNRAS, 513, 670
Ni Y. et al., 2023, ApJ, 959, 136
Oppenheimer B. D. et al., 2016, MNRAS, 460, 2157
Oppenheimer B. D., Schaye J., Crain R. A., Werk J. K., Richings A. J., 2018, MNRAS, 481, 835
Oppenheimer B. D., Babul A., Bahé Y., Butsky I. S., McCarthy I. G., 2021, Universe, 7, 209
Paszke A. et al., 2019, preprint (arXiv:1912.01703)
Peeples M. S., Werk J. K., Tumlinson J., Oppenheimer B. D., Prochaska J. X., Katz N., Weinberg D. H., 2014, ApJ, 786, 54

[5]http://www.magneticum.org/

Péroux C., Howk J. C., 2020, ARA&A, 58, 363
Pillepich A. et al., 2018, MNRAS, 473, 4077
Predehl P. et al., 2020, Nature, 588, 227
Predehl P. et al., 2021, A&A, 647, A1
Rahmati A., Pawlik A. H., Raičević M., Schaye J., 2013, MNRAS, 430, 2427
Schaye J. et al., 2015, MNRAS, 446, 521
Somerville R. S., Popping G., Trager S. C., 2015, MNRAS, 453, 4337
Springel V., 2005, MNRAS, 364, 1105
Springel V., 2010, MNRAS, 401, 791
Teyssier R., 2010, Astrophysics Source Code Library, record ascl:1011.007
Tumlinson J. et al., 2011, Science, 334, 948
Tumlinson J. et al., 2013, ApJ, 777, 59
Tumlinson J., Peeples M. S., Werk J. K., 2017, ARA&A, 55, 389
Turk M. J., Smith B. D., Oishi J. S., Skory S., Skillman S. W., Abel T., Norman M. L., 2011, ApJS, 192, 9
Villaescusa-Navarro F. et al., 2021a, preprint (arXiv:2109.09747)
Villaescusa-Navarro F. et al., 2021b, preprint (arXiv:2109.10360)
Villaescusa-Navarro F. et al., 2021c, ApJ, 915, 71
Villaescusa-Navarro F. et al., 2022, ApJS, 259, 61
Villanueva-Domingo P., Villaescusa-Navarro F., 2022, ApJ, 937, 115
Weinberger R. et al., 2017, MNRAS, 465, 3291
Weinberger R., Springel V., Pakmor R., 2020, ApJS, 248, 32
Werk J. K., Prochaska J. X., Thom C., Tumlinson J., Tripp T. M., O'Meara J. M., Peeples M. S., 2013, ApJS, 204, 17
Werk J. K. et al., 2014, ApJ, 792, 8
Wetzel A. et al., 2023, ApJS, 265, 44
Zinger E., Dekel A., Birnboim Y., Nagai D., Lau E., Kravtsov A. V., 2018, MNRAS, 476, 56
ZuHone J. A., Hallman E. J., 2016, Astrophysics Source Code Library, record ascl:1608.002.

APPENDIX A: ADDITIONAL PLOTS OF MOCK DATA SETS

In this appendix, we provide additional maps and plots, including the scatter of $M_{\rm halo}$ with total pixel counts per map in X-ray and H I (analogous to Fig. 1), and Truth–Inference plots for inferred $f_{\rm cgm}$, $\log(Z_{\rm cgm})$, $f_{\rm cool}$, and $\log(T_{\rm cgm})$ for the H I + X-ray multifield with observational limits. We omit $M_{\rm halo}$ or $M_{\rm cgm}$ here, as similar panels are shown in the diagonal panels of Figs 9 and 10. A summary of these properties and their trends with halo mass is shown in Fig. 7.

Fig. A1 is an expanded version of Fig. 2 for IllustrisTNG maps in X-ray (with and without observational limits, first and second rows, respectively) and H I (with and without observational limits, third and fourth rows, respectively) across most of the halo mass range explored in our analysis. Each column indicates four variations of the same halo.

Fig. A2 illustrates the scatter of $\log(M_{\rm halo}/{\rm M}_\odot)$, where each coloured point represents the total pixel value of each map along with the respective halo mass. 'Pixel counts', as the total flux (X-ray) or the total column density (H I), are the sum of the pixels in each map (log-scaled). We only include one image axis (even though our CNN training set uses three rotations of the same halo along the three axes) so that the same halo does not appear more than once. The black points represent the average trends in each halo mass bin (see the definitions of the mass bin in Table 2), and the error bars are the 16th–84th percentiles. Dashed grey vertical lines indicate the observational limits of each field, such that to the left of this line reside objects that would be too faint to observe with current instruments. The top row

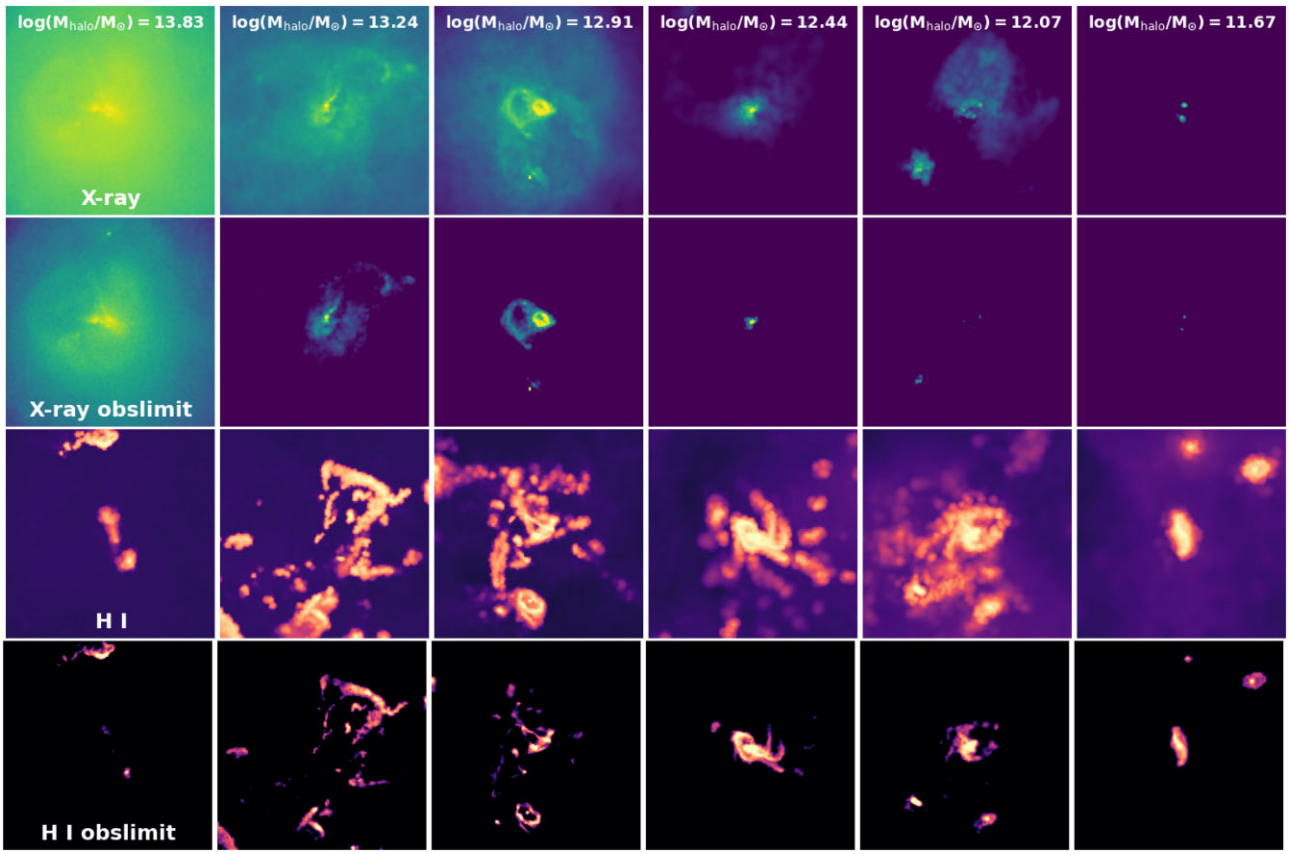

Figure A1. Maps of idealized X-ray (row 1), X-ray with observational limits (row 2), idealized H I (row 3), and H I with observational limits (row 4), as seen with IllustrisTNG. Moving across the row are haloes of decreasing mass (approximately 0.5 dex), where columns correspond to the same halo map and hence the same mass.

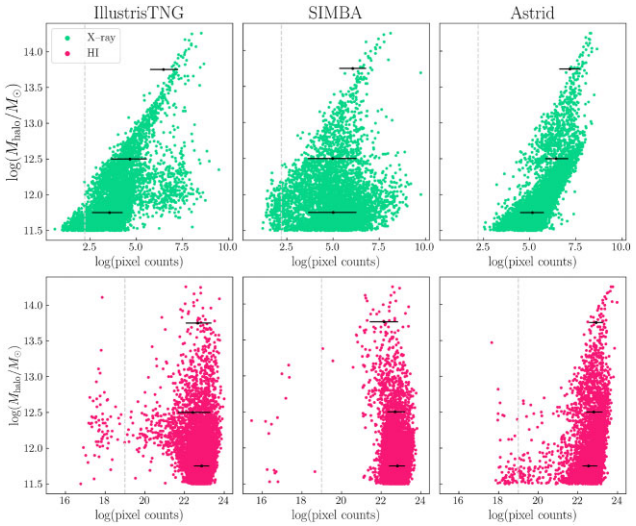

Figure A2. Halo mass as a function of the spatially integrated (total) flux in X-ray (top; green) and H I (bottom; pink) for *all maps* available from the IllustrisTNG (left), SIMBA (middle), and Astrid (right) simulations. The vertical dashed line represents the observational limit of each field, and the black points represent the average value in each mass bin. The error bars represent the average 16–84 percentile in total flux for different halo mass bins. We see correlations only for IllustrisTNG and Astrid in X-ray.

shows this scatter in X-ray, where there is a clear correlation with the halo mass for IllustrisTNG and Astrid. The bottom row shows the scatter in H I, similarly formatted. In this case, the correlations with halo mass for all simulations are either too weak or non-existent. Both fields match the expected trends from the visualization of the maps in Fig. A1. This exercise aims to see if the halo mass can be predicted solely with total flux. Since the vertical scatter is not in the same order and is much larger than the network error, we cannot conclude that the halo mass is based only on the total flux.

APPENDIX B: ADDITIONAL MULTIFIELD TRUTH–INFERENCE PLOTS

Fig. B1 shows the Truth–Inference plots for f_{cgm} with the multifield H I + X-ray and observational limits. We see a significant scatter throughout the mass range when training and testing on IllustrisTNG (left). Training and testing on SIMBA (middle) shows intermediate-mass haloes clustered at low f_{cgm} values, and low-mass haloes scattered throughout. Training and testing on Astrid (right) shows a relatively low amount of scatter with small error bars, resulting in the lowest ϵ and highest R^2 value. Also, note that Astrid shifts the entire trend towards higher f_{cgm} values and has its highest concentration of points towards higher values of f_{cgm}.

Fig. B2 shows the Truth–Inference plots for $\log(Z_{cgm})$ with the H I + X-ray multifield and observational limits. Training and testing

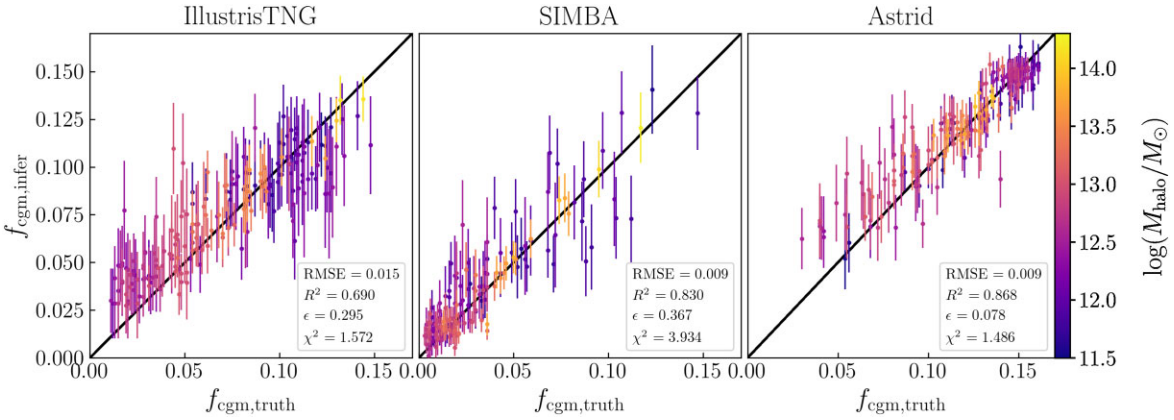

Figure B1. Truth–Inference plots for f_{cgm} using H I + X-ray with observational limits for IllustrisTNG, SIMBA, and Astrid. These points are a fraction of the full data set.

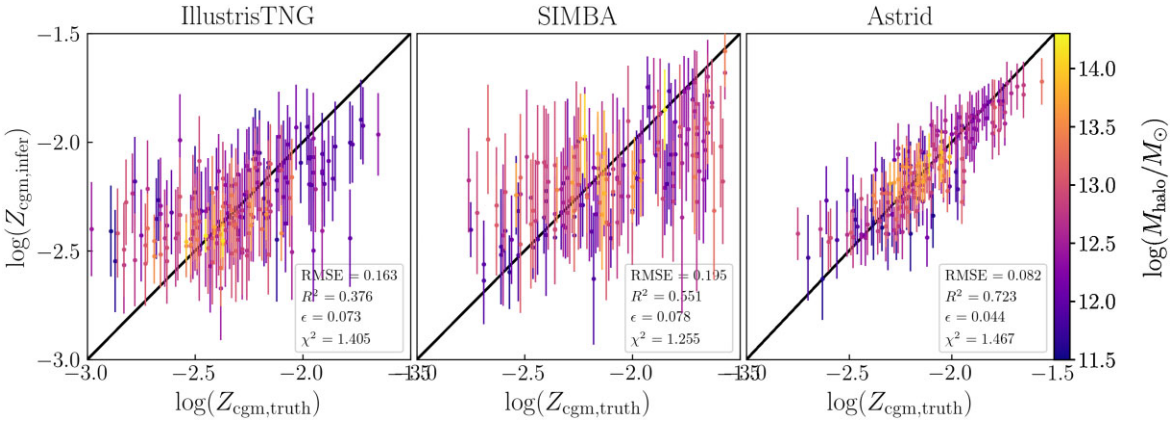

Figure B2. Truth–Inference plots for $\log(Z_{cgm})$ using H I + X-ray with observational limits for IllustrisTNG, SIMBA, and Astrid. These points are a fraction of the full data set.

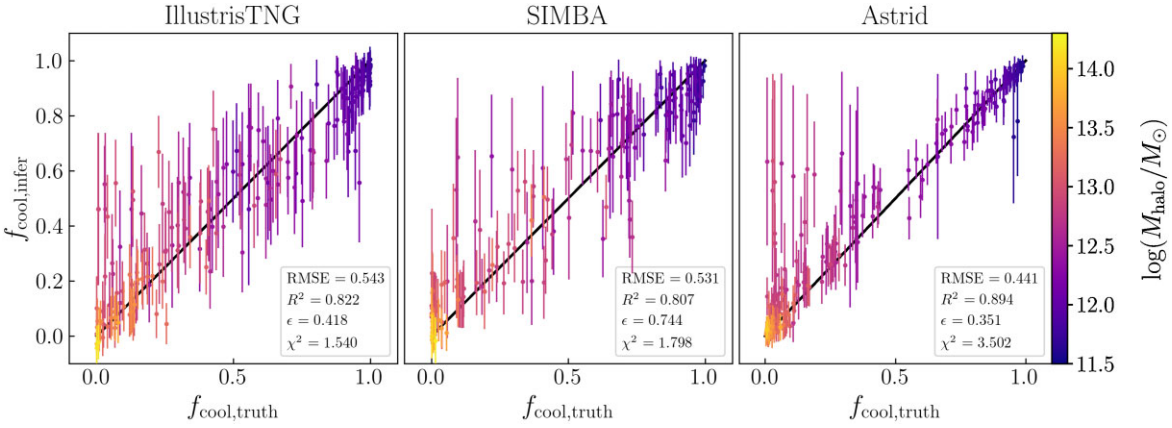

Figure B3. Truth–Inference plots for $f_{\rm cool}$ (defined in equation 5) using H I + X-ray with observational limits for IllustrisTNG, SIMBA, and Astrid. These points are a fraction of the full data set.

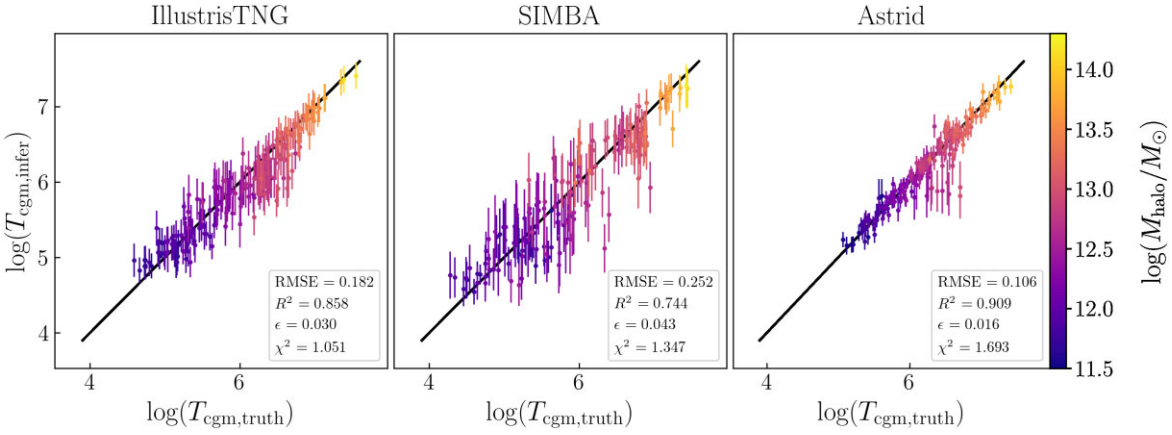

Figure B4. Truth–Inference plots for $\log(T_{\rm cgm})$ (defined in equation 6) using H I + X-ray with observational limits for IllustrisTNG, SIMBA, and Astrid. These points are a fraction of the full data set.

on IllustrisTNG (left) or SIMBA (middle) results in significant scatter. Training and testing on Astrid shows a relatively low amount of scatter with small error bars. These trends look relatively similar to those with idealized maps of X-rays and H I (Fig. 5).

Fig. B3 shows the Truth–Inference plots for $f_{\rm cool}$ with the multifield H I + X-ray and observational limits. All three simulations have inferences with significant scatter and large error bars. The multifield greatly improves the inference results for this property, which is difficult to constrain within the scope of this work.

Finally, Fig. B4 shows the Truth–Inference plots for $\log(T_{\rm cgm})$ with the H I + X-ray multifield and observational limits. Training and testing on IllustrisTNG provide relatively good inference, with increased scatter for intermediate-mass haloes. Training and testing on SIMBA show the largest scatter across the mass range. Training and testing on Astrid provides the least amount of scatter, with the smallest error bars and an overall impressive inference.

APPENDIX C: CNN ARCHITECTURE

Initially, a similar CNN was applied to the CAMELS Multifield Data set (CMD; Villaescusa-Navarro et al. 2021c) as continuous 2D maps with the aim of constraining two cosmological parameters (σ_8 and Ω_M), and four astrophysical feedback parameters ($A_{\rm SN1}$, $A_{\rm SN2}$, $A_{\rm AGN1}$, $A_{\rm AGN2}$) whose definitions change depending on the simulation used. Note that 3D maps are also available and can be reduced to obtain the existing 2D maps, but are not used for this analysis. A multifield allows the combination of fields to determine which singular or multiple fields return the tightest and most accurate constraints on any given parameter. The parameters currently available in the original network for the CMD are gas properties (density, velocity, temperature, pressure, metallicity), neutral hydrogen density, electron number density, magnetic fields, magnesium-ion fraction, dark matter density, and velocity, stellar mass density, and the total matter density.

We now define the variables used for the CNN used for this work in Table C1. The names of layers beginning with C refer to `Conv2d`, and B refers to `BatchNorm2d`. Each type of layer has different input variables as described in the first mention of the layer type, with more details in the Paszke et al. (2019) documentation for `PyTorch`. Subsequent layers of these two types do not have headings, but the numbers in the columns refer to the variable names and definitions when they are first mentioned.

For the `Conv2d` layers, `Input` and `Output` are the size of the image produced as it passes through each layer. `Kernel` refers to the size of the kernel or the grid space in any particular layer. `Stride` is the number of rows and columns that have passed through each 'slide' or translation between layers. If computational efficiency is not an

Table C1. Table outlining the main body of the CNN architecture used.

Layer	Input	Output	Kernel	Stride	Padding
C01	1	12	(3,3)	(1,1)	(1,1)
C02	12	12	(3,3)	(1,1)	(1,1)
C03	12	12	(2,2)	(2,2)	
Layer	Size	ϵ	Momentum	Affine	Tracking
B01	12	1e−5	0.1	True	True
B02	12	1e−5	0.1	True	True
B03	12	1e−5	0.1	True	True
C11	12	24	(3,3)	(1,1)	(1,1)
C12	24	24	(3,3)	(1,1)	(1,1)
C13	24	24	(2,2)	(2,2)	–
B11	24	1e−5	0.1	True	True
B12	24	1e−5	0.1	True	True
B13	24	1e−5	0.1	True	True
C21	24	48	(3,3)	(1,1)	(1,1)
C22	48	48	(3,3)	(1,1)	(1,1)
C23	48	48	(2,2)	(2,2)	–
B21	48	1e−5	0.1	True	True
B22	48	1e−5	0.1	True	True
B23	48	1e−5	0.1	True	True
C31	48	96	(3,3)	(1,1)	(1,1)
C32	96	96	(3,3)	(1,1)	(1,1)
C33	96	96	(2,2)	(2,2)	–
B31	96	1e−5	0.1	True	True
B32	96	1e−5	0.1	True	True
B33	96	1e−5	0.1	True	True
C41	96	192	(3,3)	(1,1)	(1,1)
C42	192	192	(3,3)	(1,1)	(1,1)
C43	192	192	(2,2)	(2,2)	–
B41	192	1e−5	0.1	True	True
B42	192	1e−5	0.1	True	True
B43	192	1e−5	0.1	True	True
C51	192	384	(3,3)	(1,1)	(1,1)
C52	384	394	(3,3)	(1,1)	(1,1)
C53	384	384	(2,2)	(2,2)	–
B51	384	1e−5	0.1	True	True
B52	384	1e−5	0.1	True	True
B53	384	1e−5	0.1	True	True
C61	384	768	(2,2)	(1,1)	–
B61	768	1e−5	0.1	True	True

issue, in some cases, it can be more accurate to slide one element at a time. However, cutting out the intermediate steps and increasing the stride for larger data sets like the one used here is more efficient. `Padding` refers to filling the kernel's edges after each layer. As the dimensions of the image decrease and eventually reach 1×1, we need to fill the space left after each dimensional reduction. One padding mode is the 'zeros', where values of 0 are used as a filler as the image is processed through the network. Another common mode is 'circular', where the grid is filled with the value at the boundary of the image in the current stage.

For the `BatchNorm2d` layers, where `Size` refers to the number of features based on some expected input size from the previous layer. ϵ is added to the denominator of any value to ensure the stability of the pipeline and the results. `Momentum` can be set to None if a cumulative moving average (simple average) is being computed, but the default value is 0.1 for the running mean and running variance computations. Note that this argument is slightly different from what is generally used in optimizer classes. `Affine`, if set to True, allows weights and biases to be defined, which are γ and β, respectively,

Table C2. A continuation of the neural network architecture, following Table C1.

Layer	Type	Kernel size	Stride	Padding
P0	AvgPool2d	2	2	0
Layer	Type	`Feat. In`	`Feat. Out`	Bias
FC1	Linear	768	384	True
FC2	Linear	384	12	True
Layer	Function	*p*-value	in-place	Slope
Dropout	`Dropout()`	0.3522	False	
ReLU	`ReLU()`			
LeakyReLU	`LeakyReLU()`			−0.2
tanh	`Tanh()`			

within the documentation. `Tracking`, if set to True as the default, tracks the mean and variance. If set to False, statistics buffers are initialized such that `running-mean` and `running-var` is set to None, and the module uses only batch statistics for training and testing modes.

In Table C2, the functions mentioned directly follow those of Table C1. The `P0: AvgPool2d` layer includes the `kernel size` for the window size. `Stride` and `Padding` have similar definitions as before, where `Stride` now refers to the stride of the window, where it defaults to the same value as the kernel size, and `Padding` is defaulted to 'zeros' mode, discussed previously.

A linear transformation is applied for both Fully Connected (FC) layers, where the `Feat. In` and `Feat. Out` refers to the size of each input and output sample, respectively. Setting the `bias` to True (as the default) allows the activation function to be shifted by some constant amount, known as the bias, to the layer input. The dropout layers randomly disengage some neurons with some probability, `p-value`, or just as `p`, to discourage some neurons from being favoured over others. The default *p*-value is 0.5. Finally, if set to True, `inplace` will randomly set the neurones to zero in place. The default value for this parameter is False, where the results of the dropout layer are saved to a separate variable to be potentially used later.

The `ReLU()`, or the rectified linear activation function (linear, piece-wise), takes this form:

$$\mathrm{ReLU(x)} = (\mathrm{x})^{+} = \max(0, \mathrm{x}). \tag{C1}$$

The input is returned directly if positive and will be set to zero otherwise. The `LeakyReLU()` (Leaky rectified linear unit) is defined as

$$\mathrm{LeakyReLU(x)} = \begin{cases} x, & \text{if} \mathrm{x} \geq 0 \\ \text{negative\_slope} \times \mathrm{x}, & \text{otherwise} \end{cases}, \tag{C2}$$

where the `negative_slope` controls the slope angle specifically used for negative input values. The default value is 0.001, such that instead of a flat slope for negative values, it has a small slope, determined before training begins, and is not a result of the training process. The `Tanh()` function is defined as,

$$\mathrm{Tanh(x)} = \frac{\exp(\mathrm{x}) - \exp(-\mathrm{x})}{\exp(\mathrm{x}) + \exp(-\mathrm{x})}, \tag{C3}$$

where it is used in place of the sigmoid function, as it is more computationally efficient for networks with multiple layers.

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