A&A, 687, A1 (2024)
https://doi.org/10.1051/0004-6361/202346683
© The Authors 2024

Astronomy & Astrophysics

Cosmology with galaxy cluster properties using machine learning

Lanlan Qiu[1,2], Nicola R. Napolitano[1,2,3], Stefano Borgani[4,5,6,7,8], Fucheng Zhong[1,2], Xiaodong Li[1,2], Mario Radovich[9], Weipeng Lin[1,2], Klaus Dolag[10,11], Crescenzo Tortora[12], Yang Wang[2,13], Rhea-Silvia Remus[10], Sirui Wu[1,2], and Giuseppe Longo[3]

[1] School of Physics and Astronomy, Sun Yat-sen University Zhuhai Campus, 2 Daxue Road, Tangjia, Zhuhai 519082, PR China
e-mail: zhongfch@mail2.sysu.edu.cn; napolitano@mail.sysu.edu.cn
[2] CSST Science Center for Guangdong-Hong Kong-Macau Great Bay Area, Zhuhai 519082, PR China
[3] Department of Physics E. Pancini, University Federico II, Via Cinthia 6, 80126 Naples, Italy
[4] Astronomy Unit, Department of Physics, University of Trieste, via Tiepolo 11, 34131 Trieste, Italy
[5] INAF-Osservatorio Astronomico di Trieste, via G. B. Tiepolo 11, 34143 Trieste, Italy
[6] IFPU, Institute for Fundamental Physics of the Universe, Via Beirut 2, 34014 Trieste, Italy
[7] INFN, Instituto Nazionale di Fisica Nucleare, Via Valerio 2, 34127 Trieste, Italy
[8] ICSC – Italian Research Center on High Performance Computing, Big Data, and Quantum Computing, Via Magnanelli 2, Bologna, Italy
[9] INAF – Osservatorio Astronomico di Padova, via dell'Osservatorio 5, 35122 Padova, Italy
[10] Universitäts-Sternwarte, Fakultät für Physik, Ludwig-Maximilians-Universität München, Scheinerstr.1, 81679 München, Germany
[11] Max-Planck-Institut für Astrophysik, Karl-Schwarzschild-Straße 1, 85741 Garching, Germany
[12] INAF – Osservatorio Astronomico di Capodimonte, Salita Moiariello 16, 80131 Napoli, Italy
[13] Peng Cheng Laboratory, No. 2, Xingke 1st Street, Shenzhen, 518000, PR China

Received 17 April 2023 / Accepted 12 February 2024

ABSTRACT

Context. Galaxy clusters are the largest gravitating structures in the universe, and their mass assembly is sensitive to the underlying cosmology. Their mass function, baryon fraction, and mass distribution have been used to infer cosmological parameters despite the presence of systematics. However, the complexity of the scaling relations among galaxy cluster properties has never been fully exploited, limiting their potential as a cosmological probe.

Aims. We propose the first machine learning (ML) method using galaxy cluster properties from hydrodynamical simulations in different cosmologies to predict cosmological parameters combining a series of canonical cluster observables, such as gas mass, gas bolometric luminosity, gas temperature, stellar mass, cluster radius, total mass, and velocity dispersion at different redshifts.

Methods. The ML model was trained on mock "measurements" of these observable quantities from Magneticum multi-cosmology simulations to derive unbiased constraints on a set of cosmological parameters. These include the mass density parameter, Ω_m, the power spectrum normalization, σ_8, the baryonic density parameter, Ω_b, and the reduced Hubble constant, h_0.

Results. We tested the ML model on catalogs of a few hundred clusters taken, in turn, from each simulation and found that the ML model can correctly predict the cosmology from where they have been picked. The cumulative accuracy depends on the cosmology, ranging from 21% to 75%. We demonstrate that this is sufficient to derive unbiased constraints on the main cosmological parameters with errors on the order of ~14% for Ω_m, ~8% for σ_8, ~6% for Ω_b, and ~3% for h_0.

Conclusions. This proof-of-concept analysis, though based on a limited variety of multi-cosmology simulations, shows that ML can efficiently map the correlations in the multidimensional space of the observed quantities to the cosmological parameter space and narrow down the probability that a given sample belongs to a given cosmological parameter combination. More large-volume, mid-resolution, multi-cosmology hydro-simulations need to be produced to expand the applicability to a wider cosmological parameter range. However, this first test is exceptionally promising, as it shows that these ML tools can be applied to cluster samples from multiwavelength observations from surveys such as *Rubin*/LSST, CSST, *Euclid*, and *Roman* in optical and near-infrared bands, and eROSITA in X-rays, to the constrain cosmology and effect of baryonic feedback.

Key words. methods: numerical – galaxies: clusters: general – galaxies: luminosity function, mass function – cosmological parameters – X-rays: galaxies – X-rays: galaxies: clusters

1. Introduction

According to the hierarchical clustering scenario, galaxy clusters are the largest and the most massive collapsed objects in the universe, typically residing in the nodes of the cosmic web. The virial mass of a typical rich cluster is about 10^{14}–$10^{15} M_\odot$, consisting of approximately 2% galaxies, 12% hot gas, and 86% dark matter. Due to their spatial distribution in the universe and specific mass composition, they have been widely investigated, both as an effective cosmological probe and a natural astrophysical laboratory (Allen et al. 2011; Kravtsov & Borgani 2012; Lesci et al. 2022a,b; Ingoglia et al. 2022).

With respect to their cosmological application, cluster masses and, in particular, the cluster mass function can be used to constrain both the universe mean matter density Ω_m and the density fluctuation amplitude σ_8. However, their constraining

Open Access article, published by EDP Sciences, under the terms of the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.
This article is published in open access under the Subscribe to Open model. Subscribe to A&A to support open access publication.

capacity is inevitably limited by the difficulty of deriving accurate mass estimates from observations (Pratt et al. 2019). The most precise mass estimates come from weak gravitational lensing. This has been widely exploited to calibrate mass estimations from other methods, but the cluster triaxiality and projection effects of lensing measurements limit the precision of individual cluster mass to about 5% (e.g., Hoekstra et al. 2015; Umetsu et al. 2016; Hildebrandt et al. 2017; Melchior et al. 2017; Henson et al. 2017; Euclid Collaboration 2024). Besides, weak lensing is also observationally difficult to perform, and yet today there is a rather limited statistics of clusters having accurate weak lensing mass (e.g., Sereno & Umetsu 2011; Sereno 2015; Umetsu et al. 2020; Giocoli et al. 2021). Other direct mass estimates are obtained through the virial theorem, i.e., by measuring the velocity field of galaxy members (e.g., Abdullah et al. 2020), or via Jeans analysis (e.g., Łokas et al. 2006; Falco et al. 2013; Biviano et al. 2013; Munari et al. 2014). However, the application of the virial theorem and Jeans analysis is also limited by the difficulty in measuring a large number of redshifts in individual clusters and the presence of systematics such as outliers and underlying modeling assumptions that are hard to control. Customarily, to overcome at least the observational difficulties, cluster masses are widely estimated indirectly by various means. For instance, some multi-band integrated observables of galaxy clusters are generally expected to scale with cluster masses and be used as mass proxies. Typical observables may come from the X-ray emission (e.g., Borgani & Guzzo 2001; Vikhlinin et al. 2009b; Mantz et al. 2010; Chiu et al. 2023), optical richness (e.g., Borgani et al. 1999; Rykoff et al. 2016; Maturi et al. 2019; Abbott et al. 2020), and millimeter-wave thermal Sunyaev-Zel'dovich signal (e.g., Bleem et al. 2015; Planck Collaboration XXIV 2016; Bocquet et al. 2019; Hilton et al. 2021). However, the scaling relations connecting these quantities with mass are generally very noisy and not bias-free (Mantz et al. 2016; Dietrich et al. 2019; Bahar et al. 2022). In general, cluster masses based on various methods tend to be rather scattered, leaving the constraints based on these systems under-exploited, despite the large potential (Abdullah et al. 2020; Lesci et al. 2022a).

Recent studies have shown the potential of using artificial intelligence (AI) based methods to cluster science, e.g., for mass estimation using tools trained on simulations. These studies have used a variety of cluster features, such as the velocity distribution of the cluster members (Ntampaka et al. 2015), the velocity distribution along with mock X-ray and weak-lensing analyses (Armitage et al. 2019), richness, velocity distribution, and other simulated multiwavelength measurements (Cohn & Battaglia 2020), or directly emulating the richness-mass relation (Ragagnin et al. 2023). Other studies have also considered the cluster phase space distribution (e.g., Ho et al. 2019; Kodi Ramanah et al. 2020, 2021), and stellar mass, X-ray flux, or the Compton y parameter (e.g., Yan et al. 2020; de Andres et al. 2022). These simulation-based AI schemes have been found very promising as alternatives to classical methods of cluster mass estimation.

Despite these many efforts to enhance the cosmological application of galaxy clusters by improving the accuracy of mass estimates, very little has been done to exploit the potential of all other direct observables connected to the baryonic components, which, being tightly correlated with masses, can also keep significant cosmological information. The one-to-one correlations among some typical observables, such as the stellar mass, gas mass, and X-ray flux, i.e., the so-called scaling relations, represent a viable approach to constrain cosmology (Singh et al. 2020). In principle, to fully exploit the cosmological potential of the cluster properties, one could combine the information encoded in all of the existing scaling relations among various mass-related quantities. ML is the ideal tool to extract valuable scientific information and execute a joint analysis out of such a multidimensional feature space and help establish internal links between these features and their environmental information. To be linked to cosmology and baryonic physics, these need to be trained using realistic mock data samples for which the ground truths are given. Cosmological simulations can provide such training samples as they have currently reached a rather advanced technological and theoretical level to predict the effect of cosmology (and feedback) on the baryonic + dark scaling relations over different scales, from galaxies to clusters (see e.g., Wechsler & Tinker 2018 for a review). Modern hydrodynamical simulations can capture most of this physics with a fair accuracy and study the effect of the complex baryon processes over the dark matter distribution (e.g., Borgani et al. 2004; Dolag et al. 2009; Cui et al. 2012; Vogelsberger et al. 2014; Remus et al. 2017; Pillepich et al. 2018b), though, they mostly focus on one single cosmological model.

On the other hand, multi-cosmology hydrodynamical simulations would be of paramount importance to combine cosmology and baryonic physics and possibly solve the degeneracies coming from the interplay of the dark and baryonic components (Wechsler & Tinker 2018; Villaescusa-Navarro et al. 2022). An effective strategy is to fully explore the multidimension parameter space where, on one side, one can change the cosmology, meaning the cosmological parameters and the dark matter (DM) flavors, and, on the other side, one can explore different galaxy formation models, including the stellar initial mass function, the duration, power, and location of star formation, the stellar feedback including the supernova explosions, the AGN effect, etc.

By combining Machine Learning and multi-cosmology hydrodynamical simulations, we have the possibility to build a new effective model to predict the cosmology and the formation scenario from catalogs of astronomical observables. Among the first attempts to collect predictions from a different combination of cosmology and baryonic physics scenarios, the CAMELS project[1] (Villaescusa-Navarro et al. 2021) is designed for galaxy scales while Magneticum project[2] (Singh et al. 2020) is tailored for galaxy cluster scales. The bottleneck of these applications is the availability of sufficiently large volume simulations with enough mass resolution to investigate the widest range of the systems under exams. For galaxy scales, simulation samples are sufficient to directly test the application to mock galaxy samples (e.g., Villaescusa-Navarro et al. 2022; Chawak et al. 2023; Echeverri-Rojas et al. 2023). For cluster scales, on the other hand, there are still limited multi-cosmological samples to use. One way to expand the simulation library can be the adoption of emulators or generative models, that have been already used to reproduce cosmological statistics such as galaxy clustering (e.g., Storey-Fisher et al. 2024), galaxy power spectrum (e.g., Kobayashi et al. 2022) and halo mass function (e.g., Bocquet et al. 2020).

In this first article, we start by testing the predictive power encoded in the galaxy clusters' multiwavelength and spectroscopic data of next-generation surveys to constrain the cosmology testing a suite of ML tools on Magneticum multi-cosmology simulations (Singh et al. 2020). We postpone the constraints of the feedback in this analysis because of the limited variety of

[1] https://www.camel-simulations.org/
[2] http://magneticum.org/

Table 1. Cosmological parameter values for 13 cosmological models.

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13
Ω_m	0.200	0.204	0.222	0.232	0.268	0.272	0.301	0.304	0.342	0.363	0.400	0.406	0.428
σ_8	0.850	0.739	0.793	0.687	0.721	0.809	0.824	0.886	0.834	0.884	0.650	0.867	0.830
h_0	0.730	0.689	0.676	0.670	0.699	0.704	0.707	0.740	0.708	0.729	0.675	0.712	0.732
Ω_b	0.0415	0.0437	0.0421	0.413	0.0449	0.0456	0.0460	0.0504	0.0462	0.0490	0.0485	0.0466	0.0492

feedback models currently available for these simulations. The observables available in simulations are gas mass, gas bolometric luminosity, gas temperature, stellar mass, cluster size, total mass, and velocity dispersion at different redshifts. In particular, we aim to demonstrate that ML can be trained on multi-cosmology simulations to recognize the correct universe a given cluster catalog belongs to. Then, by defining the probability for each cluster of being drawn by a cosmology with a series of cosmological parameters, we will derive the posterior probability distribution of any given cosmological parameter. Albeit we make this proof-of-concept experiment realistic enough, by including observationally motivated measurement errors, this remains a "toy model" approach. To move to real data applications, it will need a more methodical derivation of fiducial observables from simulations, in order to minimize the systematics due to the "observational realism." The inclusions of these aspects, as well as the study of the impact of other sources of systematics that can be introduced by simulation set-ups (e.g., resolution, numerical methods, etc.), are beyond the scope of this paper and will be only touched here but fully addressed in the second phase of the project, where we will investigate the application to real cluster catalogs.

This paper is organized as follows. Section 2 introduces the data we use to check this idea and the algorithm for getting the preprocessed data and preparing training and test samples. Section 3 illustrates all the ML algorithms and evaluation metrics involved to quantify the constraining power of each experiment. Section 4 lists all the results about the proper classifier, the classification of cosmological models, and the cosmological parameter inferences. In Sect. 5, we discuss the robustness of our results and some sources of systematics. Finally, we draw conclusions and outline future perspectives in Sect. 6.

2. Data

In the previous section, we have anticipated that the main aim of this work is to demonstrate the ability of a ML method to predict cosmological parameters, starting from the observables of a set of galaxy clusters. In this section, we introduce the set of multi-cosmology simulations adopted to train such a tool. The galaxy cluster catalogs derived from these simulations represent the "observational-like" data (the features) to start from, to first train the ML method and then test the predictions of the cosmological parameters (the targets). In particular, we explain how we define the training and the test samples used to train and evaluate the performances of the proposed ML tool. We also briefly discuss the limitations of the current simulation set and the need to expand the coverage of the cosmological parameter space for real applications.

2.1. Multi-cosmology simulations

Magneticum simulations are based on the N-body code P-GADGET3, which is the successor of the code P-GADGET2 (Springel et al. 2005b; Springel 2005; Boylan-Kolchin et al. 2009), from which it differs for a space-filling curve aware neighbor search (Ragagnin et al. 2016) and an improved Smoothed Particle Hydrodynamics (SPH) solver (Beck et al. 2016). The physics of these simulations are presented in a series of separate method papers: for example, Springel et al. (2005a) discusses the treatment of radiative cooling, heating, ultraviolet (UV) background, star formation, and stellar feedback processes; Tornatore et al. (2007) describes in details the chemical evolution and enrichment model, while Fabjan et al. (2010); and Hirschmann et al. (2014) present the prescriptions for the black hole growth and active galactic nuclei (AGNs) feedback.

Haloes are identified using the friends-of-friends (FOF) algorithm with linking length $b = 0.16$. The spherical overdensity (SO) virial masses (Bryan & Norman 1998) are computed using the SUBFIND algorithm (Springel et al. 2001; Dolag et al. 2009).

In this paper, we focus on the multi-cosmology simulations of the Magneticum project (Dolag et al. 2016; Singh et al. 2020, S+20 hereafter). The original simulation set includes 15 flat ΛCDM cosmological models (C1, C2, ..., C15) that run with the same initial conditions, and same feedback circumstances, but different configurations of four cosmological parameters, namely, the mass density parameter Ω_m, the power spectrum normalization σ_8, the "reduced" Hubble constant h_0, defined as $H_0/100$ km s^{-1} Mpc^{-1}, and the baryon density parameter Ω_b (see S+20, Table 1). Each simulation uses a large size (~896 h_0^{-1} Mpc) box, containing 1512^3 dark matter particles and an equal number of gas particles. The mass of the dark matter particles is $1.3 \times 10^{10}\ h^{-1} M_\odot$ and the initial mass of gas particles is $2.6 \times 10^9\ h^{-1} M_\odot$.

For each simulation, only haloes with $M_{vir} > 2 \times 10^{14}\ M_\odot$ are selected to avoid spurious detections due to resolution and other numerical effects. The catalogs of the selected clusters are obtained for different redshift snapshots, i.e., $z = 0.00, 0.14, 0.29, 0.47, 0.67, 0.90$. Taken as a whole, the numbers of identified haloes vary significantly among these 15 cosmological models due to different configurations of cosmological parameters (see S+20, Table 2). Considering that the identified haloes generated by C1 and C2 are too few (i.e., 1245 and 4810, respectively) to construct an informative sample for the ML training process, we decide to use only the other 13 cosmological models, C3, C4, ..., C15, and denote them as M1, M2, ..., M13 in this paper and consider M6, the one with the WMAP7 best-fitting configuration (Komatsu et al. 2011), as the fiducial cosmology consistently with the Magneticum project.

The cosmological parameters of M1 ~ M13 are specified in Table 1 and shown in Fig. 1, together with cosmological constraints obtained by different surveys and methods: CMB power spectra constraints (Planck Collaboration VI 2020), 3 × 2 pt analysis from DES Y1(Abbott et al. 2018), 3 × 2 pt analyses from KiDS-1000 with BOSS and 2dFLenS (Heymans et al. 2021), KiDS-1000 spec-z fiducial constraints (van den Busch et al. 2022), XMM-XXL C1 cluster abundance alone

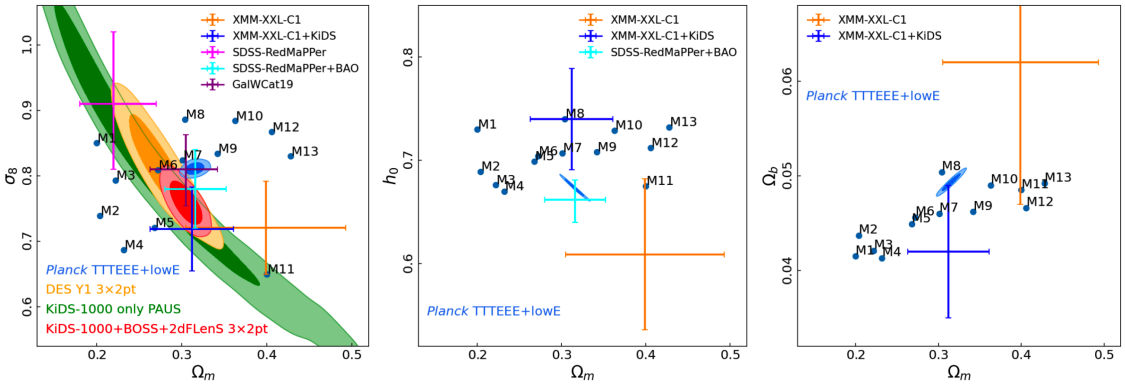

Fig. 1. Cosmological parameters' map for the 13 cosmological models. Blue points show the flat ΛCDM models in the multi-cosmology runs. For comparison, the error bars show the constraints from the XMM-XXL C1 cluster abundance alone (Pacaud et al. 2018) and plus the Kilo-Degree Survey (KiDS) tomographic weak lensing (Hildebrandt et al. 2017) joint analysis, the SDSS RedMaPPer cluster abundance alone and plus BAO joint analysis (Costanzi et al. 2019), the GalWCal19 cluster abundance (Abdullah et al. 2020). Contours show the marginalized posterior distributions of cosmic microwave background (CMB) constraints (Planck Collaboration VI 2020), 3×2 pt analysis from the Dark Energy Survey (DES) Y1 (Abbott et al. 2018), 3×2 pt analyses from KiDS-1000 with Baryon Oscillation Spectroscopic Survey (BOSS) and the 2-degree Frield Lensing Survey (2dFLenS; Heymans et al. 2021), and KiDS-1000 spec-z fiducial constraints (van den Busch et al. 2022) – see legend bottom left, in each panel.

(Pacaud et al. 2018) and adding KiDS tomographic weak lensing joint analysis (Hildebrandt et al. 2017), SDSS RedMaPPer cluster abundance alone and adding baryonic acoustic oscillation (BAO) joint analysis (Costanzi et al. 2019), GalWCal19 cluster abundance (Abdullah et al. 2020). From Fig. 1, we can see that the cosmological parameter ranges covered by the M1 ∼ M13 simulations, i.e., $0.200 < \Omega_m < 0.428$, $0.650 < \sigma_8 < 0.886$, $0.670 < h_0 < 0.740$ and $0.0413 < \Omega_b < 0.0504$, embrace the core of the confidence contours of most of the constraints of the above-mentioned experiments, especially in the $\Omega_m - \sigma_8$ space, while the constraints on h_0 and Ω_b are sometimes more scattered. This means that, in principle, the current Magneticum set of simulations is not fully representative of the overall variation of the cosmological parameters compatible with all observations. This limitation, together with the sparse coverage of the parameter space allowed by the current simulation set, does not make it optimal for applications to real data. However, with this paper, we want to make a first step toward the application to real data and test the suitability of the method for the kind of catalogs we expect to collect from current and future observations (see e.g., eFEDS, Chiu et al. 2023). On the other hand, if we demonstrate that with such a limited sample of simulations, ML is able to make predictions on the cosmological parameters underlying some cluster observations, then we can expect that the method will be even more effective when the simulation sample will be expanded to a wider range of parameters and a more fine coarse coverage of the parameter space. Hence, besides testing the suitability of this novel approach to infer cosmology from cluster observations, another outcome of the proof-of-concept test, discussed in this work, is to concretely motivate the investment in more extended simulation set-ups to offer flexible and accurate inferences.

2.2. Features and labels

Each of the selected clusters has corresponding features and a label. The labels are the cosmological models they come from, i.e., M1 ∼ M13. The features are the physical properties of the identified clusters in each simulation, namely:

1. R: the radius of the cluster, i.e., the comoving radius of a sphere centered at the minimum of the potential encompassing a given mean overdensity, in $h^{-1}(1+z)^{-1}$ kpc.
2. M_*: the stellar mass of the cluster, i.e., the sum of the mass of all star particles within the mean overdensity radius, R, defined above, in $h^{-1}M_\odot$.
3. M_g: the gas mass of the cluster, i.e., the sum of the mass of all gas particles within R, in $h^{-1}M_\odot$.
4. M_t: the total mass of the cluster, i.e., the sum of the mass of all star, gas, and dark matter particles within R, in $h^{-1}M_\odot$.
5. L_g: the gas luminosity of the cluster, i.e., the X-ray bolometric gas luminosity within R, in 10^{44} erg s^{-1}.
6. T_g: the gas temperature of the cluster, i.e., the mass-weighted gas temperature within R, in keV.
7. σ_v: the velocity dispersion of the cluster, i.e., the mass-weighted velocity dispersion of all particles belonging to a FOF halo, in km s^{-1}.
8. z: the redshift of the cluster.

All these features are continuous variables except for z, which only has six discrete values (0, 0.14, 0.29, 0.47, 0.67, 0.9). From the definitions above, we see that M_*, M_g, M_t, L_g and T_g are R-dependent quantities, i.e., they are integrated within a given overdensity radius, while σ_v is independent of R and has one value per halo (S+20). Magneticum simulations provide six typical definitions for radius. In addition to the standard virial radius, R_{vir}, at which the mean density crosses the one of a theoretical virialized homogeneous top-hat overdensity (Bryan & Norman 1998), there are radii corresponding to cluster densities which are 200 times (R_{200M}) and 500 times (R_{500M}) the mean matter density of the Universe at the cluster's redshift. Furthermore, there are the R_{200C} and R_{500C} radii that are similar to R_{200M} and R_{500M}, but based on the critical density of the Universe. In principle, we could use any of these radius definitions, as we can find a mapping of the values of cluster features between different definitions of characteristic radii by assuming a theoretical halo density profile (e.g., NFW profile, Navarro et al. 1996). However, to be consistent with the usual choices in previous literature (e.g., Liu et al. 2022), we adopt R_{500C} as the reference radius, and all quantities related to this radius in the rest of this analysis.

2.3. Preprocessed data

Data preprocessing refers to cleaning, transformation, integration, normalization, and other operations on the raw data before

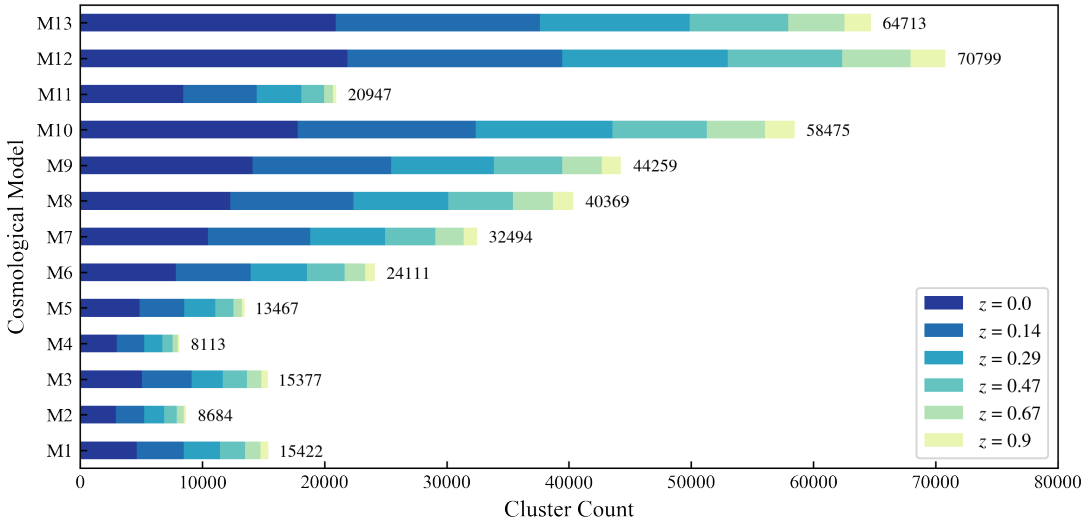

Fig. 2. Cluster count in each cosmological model and at each redshift. All of these clusters have already undergone the data preprocessing described in Sect. 2.3. The y-axis lists the different cosmological models, while the cluster counts are displayed in horizontal bars. The different redshifts are represented by different colors, as shown in the legend. As can be seen, the number of galaxy clusters in these 13 models varies significantly, with M12 (70,799) having almost nine times as many clusters as M4 (8,113). To balance the training sample among different cosmologies, we adopt an undersampling, as described in Sect. 3.1.

using ML algorithms to make the data more suitable for the training and testing of ML models. Through the inspection of the original data, we found that there are problems such as outliers and heavy-tailed distribution that, if not cleaned, can affect the training and prediction of the model to a certain extent. Hence, we decided to select all quantities defined within R_{500C} and deleted clusters with obviously nonphysical properties, such as negative M_* or σ_v, likely coming from artifacts of the FOF algorithm (0.2% over all simulations).

After this first cleaning step, being the features in simulations quite idealistic as, for instance, they do not have measurement errors, we decided to implement some "rough" observational realism. We artificially add Gaussian errors to mock a measurement process and make the quantities extracted from the simulation more similar to real cluster observations. As for the measurement errors, we have checked typical cluster observables from the literature and used eFEDS for reference. For example, in Bahar et al. (2022), M_g, L_g, and T_g, have typical relative errors of the order of 1%, 2%, and 4% or less, respectively. For M_t, Liu et al. (2022) provides errors of the order of 1%, which might be a little optimistic if compared to typical mass errors from weak lensing. To be conservative, we decided to adopt 5% relative errors as a reference experiment over all cataloged features discussed in Sect. 2 except for z which are assumed here to be spectroscopic redshift with negligible errors. However, we will also consider more conservative errors of the order of 10% for all features and up to 30% for the total mass. This latter takes into account the largest errors obtained in weak lensing analyses of mid-low mass clusters (see e.g., Sereno et al. 2018). After adding Gaussian noise to features other than z, we further performed logarithmic processing to solve the heavy-tailed distribution problem and make the predictive performance of subsequent ML models more stable.

The "mock" observations have been implemented by reassigning, to each cluster, the "observed" physical quantities ($R, M_*, M_g, M_t, L_g, T_g, \sigma_V$), assuming Gaussian errors. This is done by randomly drawing the observed quantities from a Normal distribution centered in their original (true) value and with standard deviation corresponding to the adopted relative errors (in turn, 5%, for the reference experiment, as discussed above). This produces catalogs of observable-like features we will use for training and testing the ML tool (see Sect. 3.2). To give an overview of the final catalogs provided by the Magneticum multi-cosmology sample, we first visualize the cluster count as the function of both the cosmological model and redshift, in Fig. 2. The different cosmological models are listed on the y-axis, and for each horizontal bar showing the cluster counts, different colors represent different redshifts as in the legend. As expected, we see that the total number of galaxy clusters in different universes varies greatly due to cosmological parameters. For example, M12 and M13 reach more than 60 000 clusters up to $z = 0.9$, while M2 and M4 have fewer than 9000 galaxy clusters in a volume of the same size. As the M1 ~ M13 models are listed with increasing $\Omega_m$values, this is mainly the impact of the mass density of the Universe making the cluster collapse more effective.

In Fig. 3, we also show all possible correlations (scaling relations) among the seven features for three cosmologies at redshift $z = 0$ (left), and as a function of the redshift for M6 (right), with M6 being the reference cosmology for Magneticum (see Sect. 2.1). This "cluster feature map" gives an impression of the scatter and the variation the ML method needs to be sensitive to, to distinguish different cosmologies and make correct predictions. Overall, from the figure we can see that some of the correlations are clearly distinguishable as a function of the cosmology at a fixed redshift (e.g., the correlations with M_* or the M_g–M_t in the left panel), while other correlations are rather mixed (e.g., the correlations involving the size, R). However, besides correlations, we can see that the expected distributions are different (see corner histograms), meaning that also the cluster densities in the parameter space can be used to distinguish cosmologies. We can also see that, for a given cosmology, there is a clear evolution of almost correlations with redshift (right panel). We expect the ML tools we intend to develop here can efficiently capture these features in the cluster catalogs.

It is worth noting that most of these features are standard products of cluster surveys, e.g., M_*, M_g, L_g, and T_g (Pratt et al. 2009; Vikhlinin et al. 2009a; Böhringer et al. 2013;

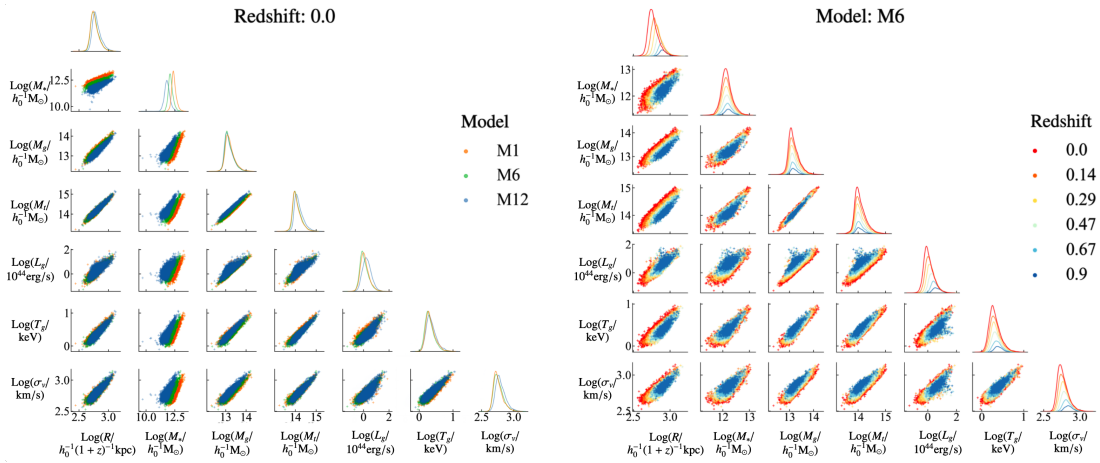

Fig. 3. Cluster features with 5% measurement errors. The panels show all possible correlations (i.e., scaling relations) among the seven features for three cosmologies (M1, M6, and M12) at redshift $z = 0$ (left), and as a function of the redshift ($z = 0, 0.14, 0.29, 0.47, 0.67, 0.9$) for M6 (right). Left panel: for a fixed redshift, we use a random sample of 7000 clusters from each cosmology to show how the slope of the scaling relations is affected by cosmology. This is particular clear for the scaling relation related to the stellar mass, M_*, and gas mass M_g, while all other scaling relations are more mixed. The corner histogram also shows the normalized distribution of the features in a given cosmological volume. Right panel: for a given cosmological model, apart from the differences in number counts, the galaxy clusters at different redshifts show similar power-law structures but with offsets driven by redshifts. This "cluster feature map" gives an overall impression of scatters and variations of cluster features among different cosmologies, which is the cornerstone of the method that uses galaxy cluster features to predict cosmology based on machine learning. For more details on definitions and accessibility of these cluster features, see Sects. 2.2 and 2.3, respectively.

Bulbul et al. 2019), while some other quantities are harder to get in real observations. For example, with respect to imaging and X-ray observations, only the most massive clusters can be used to derive precise total mass M_t (e.g., with weak lensing measurements). Similarly, σ_v needs time-consuming spectroscopical campaigns, and generally, these are also limited to a few tens of cluster members, although upcoming large all-sky redshift surveys (DESI: DESI Collaboration 2016, WEAVE: Dalton et al. 2012, 4MOST: de Jong et al. 2019) will soon produce rather large catalogs of clusters internal kinematics.

Hence, in this work, we have the chance to optimize the number of observables that are needed to constrain the cosmology. By performing a "feature importance" analysis, we can check if ML can fully exploit the cosmological information encoded in some features, and their scaling relations, with respect to others, for example, checking the impact of the quantities that are observationally more difficult to obtain, e.g., M_t and σ_v.

3. The Machine Learning Cluster Cosmology Algorithm

In Sect. 2, we have introduced the multi-cosmological simulation data and related cosmology labels and described the eight observational-like cluster features. In this section, we present the full Machine Learning Cluster Cosmology Algorithm (MLCCA), which we train to predict the best cosmology given a set of cluster observations (mock catalog). As anticipated, for this proof-of-concept we want to first demonstrate if an ML tool can recognize what cosmological simulation a given dataset has been extracted from. The basic idea is to produce random mock catalogs extracted from one of the M1 ∼ M13 simulations (including clusters from different redshifts) and let the MLCCA decide from which simulation this has been picked, on the basis of the correlations among the features (scaling relations as in Fig. 3). This can be treated as a typical classification problem, where a ML classifier can predict the probability that a dataset belongs to different cosmological models. This is the most obvious choice, given the limited number of cosmologies, although we will test also regression algorithms in the near future.

Classification-wise, due to the similarity of cosmological scaling relations in adjacent parameter spaces, the classification itself will have an error. This essentially produces uncertainties in the inference of cosmological parameters. Also, by sparsely sampling the cosmological parameter space (see Fig. 1), we can check whether the MLCCA can learn a pattern among the scaling relations in the cosmological parameter space and interpolate data coming from a "cosmology" (meaning a simulation) that is not included in the training. We quantify each of these steps by proper evaluation metrics defined in Sect. 3.3. The final goal is to build an algorithm that, starting from cluster catalogs, can return confidence contours of the four cosmological parameters ($\Omega_m, \sigma_8, h_0, \Omega_b$) used as labels in the ML training.

3.1. Machine learning classifiers

Broadly speaking, the task of the classifier will be to issue the probability for a given cluster i to belong to a given cosmological model j. Machine learning classifiers are mainly divided into two types: tree models and neural networks. In this work, we want to use tree models which are generally more robust and possess better interpretability than neural networks (Breiman 2001). In particular, we are interested in ensemble learning on tree models, which is a way to optimize the accuracy of single-tree models. The improvement of the performance, here, is obtained by constructing a set of tree models and then classifying new data points by taking a (weighted) vote on their predictions (Dietterich 2000), hence overcoming the nonoptimal performance (underfitting, overfitting, etc.) of each individual tree model.

To perform the classification on 13 cosmological models based on available features, we consider four typical ensemble tree models, i.e., Random Forest (RF, Breiman 2001), Extra Trees (ET, Geurts et al. 2006), Light Gradient Boosting machine

(LGB, Qi 2017) and eXtreme Gradient Boosting (XGB, Chen & Guestrin 2016). To select the most appropriate model for this project, we evaluated the above four ML models and selected the best option using appropriate evaluation criteria such as accuracy and logloss as described in Sect. 3.3.1. We anticipate here that LGB is the best solution, as it is discussed in detail in Sect. 4.1.

3.2. Training and test samples

For the training phase, we use the cluster features as the input to obtain the label of the predicted cosmological model as the output. In particular, we adopt a multi-class classification, which directly gives the probability that a cluster may belong to any of 13 available models, and take the model with the highest probability as the predicted model.

Regarding the construction of the training sample, the number of galaxy clusters in different cosmologies varies greatly due to the influence of cosmology itself on large-scale formation, as shown in Fig. 2. This uneven distribution can likely force the model prediction to skew toward categories with a higher number of samples (Prati et al. 2004). To correct this effect, we apply an under-sampling method, i.e., we reduce the size of the samples in the majority classes to balance the datasets of the smaller classes. Since all selected cosmologies have more than 8000 galaxy clusters, we randomly draw 7000 galaxy clusters for each cosmology as training samples. We stress here that this is a rather brute-force approach driven by low Ω_m cosmologies, producing a low number of clusters, that strongly penalizes the predictive power for more populated cosmologies in Fig. 2. We have decided to accept this drawback in order to keep the largest number of cosmologies for this first test based on the current Magneticum sample. For the testing phase, in order to make full use of the left behind non-training objects for each cosmology, we randomly selected 20 times 700 clusters, to obtain 20 different test samples with no overlap with the corresponding 7000 clusters which make up the training sample. Each test sample (i.e., mock catalog) can be regarded as a sample representative of typical observational catalogs currently available for cosmological tests (see e.g., Adami et al. 2018; Sereno et al. 2020).

3.3. Evaluation metrics

Here, we introduce the metrics to assess the three main tasks of this paper: 1) selecting the best classifier capable of performing the multi-class analysis of the mock catalogs; 2) classifying mock catalogs belonging to different cosmological models; 3) predicting cosmological parameters for a certain galaxy cluster mock catalog. In all cases, we first train the ML tool using an ensemble of clusters with the same size from each of m cosmological models distinguished by their labels (the four cosmological parameters). Then, we use a test set that contains n clusters coming from the same cosmology to finally measure the performance of the results. All the corresponding quantities of model j ($j \in \{1, 2, ..., m\}$) and cluster i ($i \in \{1, 2, ..., n\}$) are defined as follows:

1. $\{\theta_j\}$: cosmological parameters of model j;
2. $\{X_i\}$: features of cluster i;
3. y_i: true cosmological model of cluster i;
4. $\hat{y}_i$: predicted cosmological model of cluster i;
5. $\{\theta_i\}$: true cosmological parameters of cluster i;
6. $\{\mu_i\}$: mean values of predicted cosmological parameters of cluster i;
7. $\{\sigma_i\}$: standard deviations of predicted cosmological parameters of cluster i;
8. $P(\theta_j|X_i)$: probability that cluster i belongs to model j, which is the outcome of the classifier,

where cosmological parameters $\theta, \mu \in \{\Omega_m, \sigma_8, \Omega_b, h_0\}$, cluster features $X \in \{R, M_t, M_*, M_g, L_g, T_g, \sigma_v, z\}$, model labels $y, \hat{y} \in \{1, 2, ..., m\}$ and the sum of predicted probabilities for each cluster $\sum_{j=1}^{m} P(\theta_j|X_i) = 1$.

3.3.1. Classifier metrics

For the classifiers' performances, we include the following evaluators: 1) accuracy and 2) logloss. By accuracy we indicate the proportion of all correctly classified samples ($N(\hat{y}_i = y_i)$) in all samples (n). To estimate that we use the following equation:

$$\text{Accuracy} = \frac{N(\hat{y}_i = y_i)}{n} \tag{1}$$

ranging from 0 to 1. The closer to 1, the better the classifier performance on the whole. The logloss represents the average probability (in logarithm) of a cluster being correctly classified. The equation defining this is:

$$\text{Logloss} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\delta_{j,y_i}\log(P(\theta_j|X_i)), \tag{2}$$

where δ_{j,y_i} equals 1 if $j = y_i$ and 0 otherwise. The lower and upper limits of probability are set as 10^{-15} and 1, respectively, to avoid infinity in the logarithm. The Logloss also ranges from 0 to 1, and the closer to 0, the better the classifier performance.

3.3.2. Classification metrics

Once we have defined the best classifier, we can proceed with assessing the performance of the classification. This will be based on the recall rate, which represents the ratio of correctly predicted samples with respect to the total sample.

For each model, the classifier returns a true or false binary outcome and will produce four different results, in terms of correct (positive) or incorrect (negative) prediction: (1) TP: Truly predict positive to be Positive; (2) FP: Falsely predict negative to be Positive; (3) TN: Truly predict negative to be Negative; (4) FN: Falsely predict positive to be Negative. The recall rate of model j ($j \in \{1, 2, ..., m\}$) is defined as the fraction of the correctly classified j samples in all real j samples, as follows:

$$\text{Recall}_\text{j} = \frac{N(\hat{y}_i = y_i = j)}{N(y_i = j)} = \frac{\text{TP}_\text{j}}{\text{TP}_\text{j} + \text{FN}_\text{j}} \tag{3}$$

This ranges from 0 to 1, and the closer it is to 1, the better the classifier performance on model j is. As we are dealing with a multi-classification problem, the TP, FP, TN, and FN are defined in Eq. (3), with respect to the maximum probability received by each cluster i among the 13 j cosmologies. In principle, we could use a lower threshold to account for a reasonably significant probability for the ML tool to "recognize" a cluster to belong to a given cosmology, but this would alter the final distribution of the recall and arbitrarily reduce the "errors" on the classification[3]. On the other hand, assuming no lower threshold we can stress test the overall method by minimizing its accuracy and checking if it can really produce correct classifications and cosmological parameter estimates.

[3] We have tested a series of lower threshold such as 0.1, 0.2, 0.3 and checked that this would increase the TPs and reduce the FNs, overall improving the recall rate.

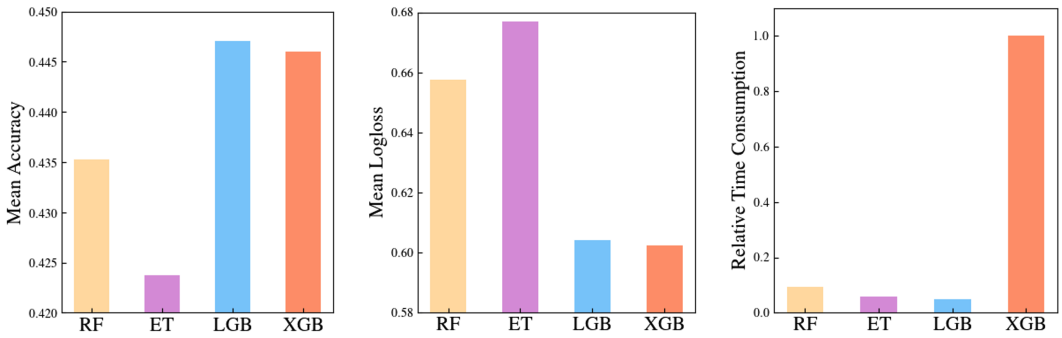

Fig. 4. Performance comparison of four classifiers (RF, ET, LGB, and XGB). From left to right is the result of mean accuracy, mean logloss, and relative time consumption during the cross-validation process.

3.3.3. Cosmological parameter metrics

After classification, for each cluster i, we use the probability that cluster i belongs to model j, $P(\theta_j|X_i)$, to infer its cosmological parameters. For each individual cluster, in principle, we can define the mean and standard deviation of a certain parameter as

$$\mu_i = \sum_{j=1}^{m} P(\theta_j|X_i) \cdot \theta_j \tag{4}$$

$$\sigma_i^2 = \sum_{j=1}^{m} P(\theta_j|X_i) \cdot (\theta_j - \mu_i)^2, \tag{5}$$

where $P(\theta_j|X_i)$ is considered as a probability distribution. Using the same $P(\theta_j|X_i)$, in order to account for asymmetric errors, we decide to compute the lower 16% percentile, the median, and the upper 84% percentile, roughly corresponding to 1σ lower bound, σ_l, median $\hat{\theta}_m$, and 1σ upper bound, σ_u, respectively. Then, we use 1) Bias and 2) Score to evaluate the parameter predictions. The Bias represents the deviation between the predicted median and the true value, i.e.,

$$\text{Bias} = \hat{\theta}_m - \theta. \tag{6}$$

The Score is short for Standard Score, which represents the magnitude of Bias relative to a confidence interval.

$$\text{Score} = \begin{cases} \dfrac{\hat{\theta}_m - \theta}{\sigma_l} & \text{when } \hat{\theta}_m > \theta \\ \dfrac{\hat{\theta}_m - \theta}{\sigma_u} & \text{when } \hat{\theta}_m < \theta. \end{cases} \tag{7}$$

We can finally obtain the marginalized 2D 1σ and 2σ confidence contours of all combinations of the four parameters, as the 68% and 95% enclosed probability of the probability distribution function (PDF) of the cluster catalog (see also Appendix A for more details). This latter can be defined as PDF= $\sum_{i=1}^{n} G(\mu_i, \sigma_i) = 1$, assuming a Gaussian distribution, $G(\mu, \sigma)$, for the cluster individual parameter estimates. We stress here that this returns a conservative estimate of the uncertainties of the parameter, fully capturing the uncertainties in the classification encoded in the σ_i.

4. Results

In this section, we show the results of 1) selecting the best classifier, 2) mock catalog classification, and 3) cosmological parameter estimates. We first choose the best classifier for the MLCCA, according to the performance evaluation discussed in Sect. 3.3. Then, we apply the MLCCA to the test sample described in Sect. 3.2 and assess its performance, including the accuracy and precision of the cosmological parameter estimates, in the perspective of future applications over real datasets.

4.1. Selecting a proper classifier

We start by using the four classifiers (RF, ET, LGB, and XGB) to perform a first-round test on the training sample with 5-fold cross-validation. That is, in five subsequent experiments, we rotate 4/5 of the sample as a training sample, and the other 1/5 as a test sample to calculate the results, and then take the mean of the five test experiments as the final result. In Fig. 4, we show the three indicators discussed in Sect. 3.3.1, i.e., the mean accuracy and mean logloss. We also show the computing time needed during the cross-validation process as a further indicator of the efficiency of the method. We find that the LGB has the highest mean accuracy and the 2nd lowest mean logloss with minimal time consumption. Therefore, we identify LGB as the best ML classifier among the four considered in our analysis, as it possesses clear advantages due to the fast training, high accuracy, and low memory footprint. These performances come from its ability to discretize continuous features through a histogram-based decision tree algorithm and to use distributed gradient boosting decision trees (GBDT), which are specifically efficient to improve training efficiency. To further optimize the LGB and reach a higher accuracy, we use Optuna (Akiba et al. 2019), which is an automated hyperparameter tuning framework, to mainly adjust `learning rate` and `n_estimators`, that are strictly related to accuracy. We finally find that the combination of `learning rate` = 0.07 and `n_estimators` = 150 can improve the accuracy and also reduce the logloss with respect to the default configuration with `learning rate` = 0.1 and `n_estimators` = 100. However, the mean accuracy of 5-fold cross-validation for the latter is 0.447 while for the optimized version is 0.449. Also, the mean logloss for the default configuration is 0.604 while for the optimized version is 0.602. Hence, from default to optimized LGB, the accuracy has increased by 0.002 and the logloss has decreased by 0.002. These are small changes, which prove that there is not much freedom in the set-up of the network and the final performances are fully dominated by the intrinsic complexity of the data and how these reflect the cosmological information encoded in them.

4.2. Classifying cosmological models

We now apply the MLCCA based on the optimized LGB to the test samples of 13 cosmological models respectively. In Fig. 5,

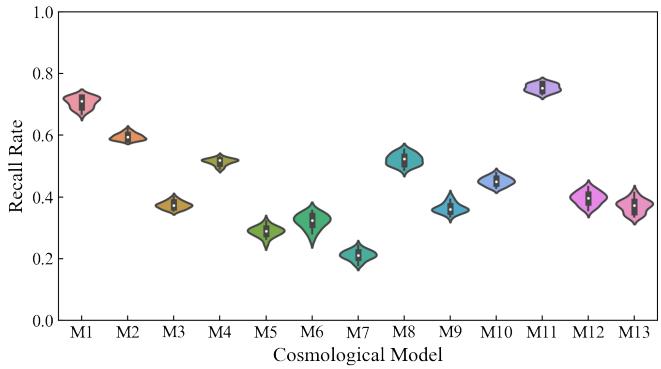

Fig. 5. Recall rate of mock catalogs over the 20 test samples in each cosmological model. The recall rate represents the proportion of galaxy clusters that are correctly classified into all clusters. The width of the "violin" is a function of the recall rate, representing the probability distribution of recall of the 20 test samples. The white dot in the center of the violin represents the median recall. As can be seen, the median recall rate displays distinct variations among different cosmological models.

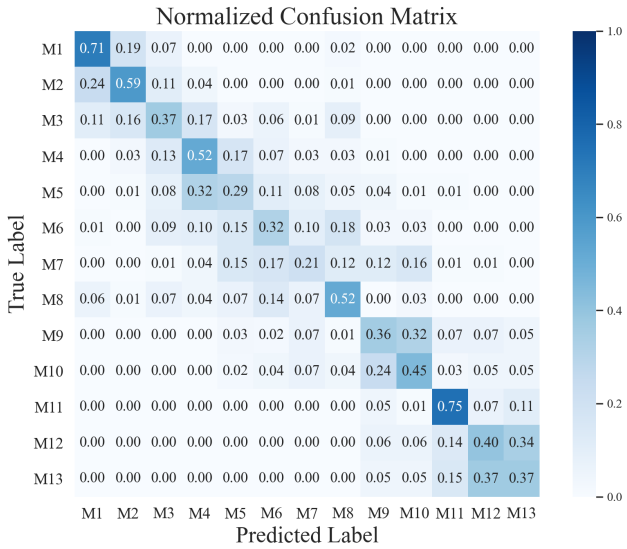

Fig. 6. Normalized confusion matrix of mock catalogs for test samples. Each row of this matrix represents a test sample taken from a certain cosmology (containing 700 galaxy clusters), where each cell represents the fraction of galaxy clusters classified as belonging to the x-label cosmology. The diagonal of the matrix represents the recall rate (i.e., the fraction of clusters correctly classified) coinciding with the median recall of each violin in Fig. 5. The non-diagonal elements of the matrix represent the fraction of clusters that have been misclassified to other universes. As it can be seen from Fig. 5 and this figure, ML has a low recall rate and large misclassified fractions for the central models (such as M5, M6, M7, and M9), indicating that these cosmologies have more overlap with neighboring cosmologies.

we show the statistics of the overall recall rate over the 20 test samples used in each cosmology. For each cosmological model, due to the variance among 20 test sets, the recall distribution has a certain fluctuation, which we quantify with a violin plot, where the width of each violin represents the probability at a certain recall level. The "median" recall rate varies from different cosmological models, with lower recall rates found for cosmologies that have more overlap with neighbor cosmological models, given a larger chance that the classifier assigns a cluster to some close cosmology.

For each mock test sample from a given cosmology from the violin diagram above, in Fig. 6 we show the median confusion matrix, showing the median fraction of a given cluster sample that has been classified on each cosmology, color-coded by the density of the allocated cluster in a given sample. A perfect classifier would return a series of 1 along the diagonal, while in Fig. 6 we see this is not the case, as the confusion matrix mirrors the situation seen in Fig. 5. In particular, we can see that for simulations with larger overlaps with close cosmologies, there is a larger spread or recall cluster from each sample. However, in all cases (except M5[4]), the classifier assigns the plurality of the cluster of the sample to the correct cosmology (along the diagonal), while the misclassified clusters still carry on their cosmological information. As we will see in the next sections, this cosmological information remains encoded in the classification probabilities among all these cosmological models and effectively impacts the recovery of the true cosmological parameters, as well as their uncertainties.

4.3. Inferring cosmological parameters

We can now check the performance of the MLCCA in the prediction of the cosmological parameters from the test sample, using the metrics described in Sect. 3.3. In Fig. 7, we start by showing the score of the predicted cosmological parameters (reported on the x-axis) for all cosmological models (y-axis). This plot gives in one glance the accuracy and precision for each cosmological parameter as a function of the "true" cosmology the mock catalog is originally extracted from. For instance, for the catalog extracted from M13 (on the top row), only σ_8 is constrained at less than 1σ level, while the other parameters are off the scale, i.e., are "biased" by $\sim 1\sigma$. Similarly for M1 (bottom row) none of the parameters is constrained with accuracy better than 0.5σ.

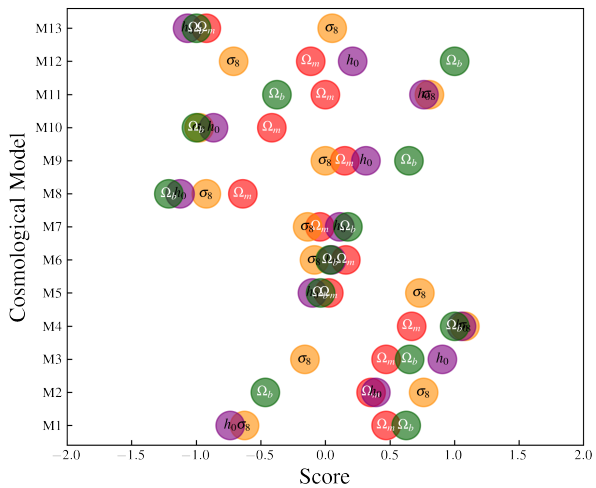

Fig. 7. Score values for the different cosmological models. The figure shows the distribution of the estimated cosmological parameters represented with different colors. Negative and positive score values indicate underprediction and overprediction, respectively. As can be seen, almost all parameters are predicted by the MLCCA within 1σ from their true value. Notably, for cosmologies at the center of the parameter space, such as M5, M6, M7, and M9, the MLCCA method can accurately recover the four cosmological parameters well within the 1σ level.

[4] We note that the close off-diagonal bin has a recall rate which is larger but consistent with the diagonal one within Poissonian noise. As we will show in Appendix B, this does not impact an unbiased cosmological parameter prediction.

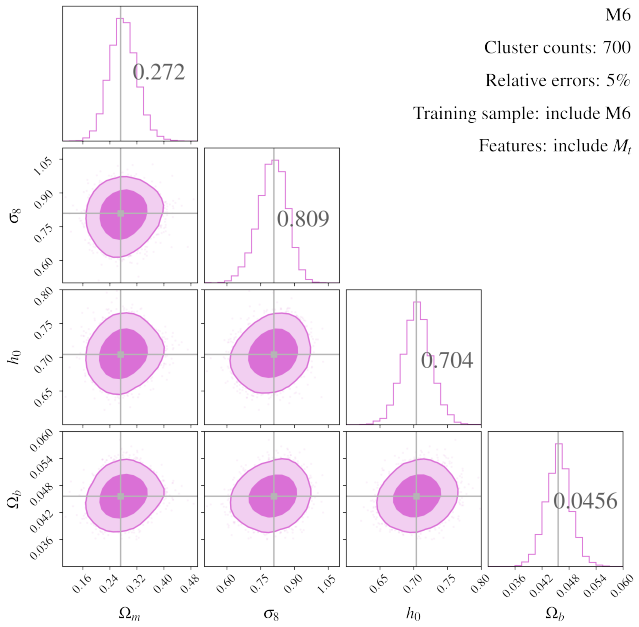

Fig. 8. Cosmological parameters of M6 inferred by the MLCCA. The contours enclose 1σ and 2σ confidence intervals for the cosmological parameters of each 2D projection. The histograms on the diagonal represent the posterior probability distribution of the four cosmological parameters. The gray lines in the figure represent the true values of the various parameters of M6 cosmology, with the true value of each parameter shown in the posterior probability diagrams. It can be seen that all cosmological parameters are within the 1σ confidence interval.

On the other hand, for models such as M5, M6, M7, and M9, the MLCCA correctly recovers Ω_m, σ_8, h_0, and Ω_b with the true values all well within 1σ confidence intervals of the prediction ranges. Overall, the models lying in the bulk of the parameter space covered by the Magneticum multi-cosmology simulations obtain a $|score| < 0.5$ for most of the cosmological parameters, especially Ω_m and σ_8. Besides, there is a mild trend that the farther a parameter is from the parameter space bulk in Fig. 1, the larger the probability of being under or overestimated.

We can have a better perception of the remarkable accuracy and precision of the recovered parameters from the corner plot in Fig. 8, where we draw the confidence contours for M6. As mentioned before, the mock catalog from M6 cosmology, used to derive these constraints, contains R, M_*, M_g, M_t, L_g, T_g, σ_v, and z values for 700 galaxy clusters, having all, except the redshift, relative error of 5%. The predicted values are $(0.279^{+0.041}_{-0.039}, 0.806^{+0.060}_{-0.066}, 0.705^{+0.021}_{-0.021}, 0.0457^{+0.0027}_{-0.0028})$, respectively. They are all consistent with the true values of $(\Omega_m, \sigma_8, h_0, \Omega_b)$ of M6, which are $(0.272, 0.809, 0.704, 0.0456)$, within the estimated errors. The corresponding 1σ relative precisions are 14% for Ω_m, 8% for σ_8, 3% for h_0, 6% for Ω_b. These constraints are somehow tighter for Ω_m but similar to the ones on σ_8 of the ones obtained by using joint analyses of the cluster abundance and the weak-lensing mass calibration (22% for Ω_m and 8% for σ_8 in, e.g., Chiu et al. 2023). This can be due to the error size adopted here, which might be optimistic for some parameters, although they are still more conservative than the ones from Chiu et al. (2023). In Sect. 5.3, we will check the impact of even more conservative errors and see that the parameter precisions are little affected, except for Ω_m.

We finally remark that, for a certain cosmological model, both the accuracy of the classification and the estimated parameters are related to its position in the parameter space (i.e., the parameter distribution: Fig. 1). Some of the more extreme cosmologies, such as M1 and M11, are at the edge of the sampled parameter space, so they are easier to recognize by classifiers and therefore have higher classification accuracy (see confusion matrix: Fig. 6). At the same time, though, due to their position on the edge of the parameter space, the misclassified clusters are oddly distributed, as they are mixed with cosmology located more likely on the same side of the parameter space (at least in some projections), resulting in an overall overestimation or underestimation of some parameters with a larger overlap. For instance, M1 and M11 lie in the opposite edges of the $\Omega_m - \sigma_8$ and $\Omega_m - h_0$ projections in Fig. 1, which makes them easy to classify (recall rate larger than 0.7 in Fig. 6); however, from Table 1, M1 seats on the minimum of the Ω_m range and close to the maximum of σ_8 and these parameters are overestimated and underestimated[5], respectively (see Fig. 7), while M11 has a minimum in both σ_8 and h_0, which are overestimated and is the second ranked in Ω_b (see Table 1), which is underestimated (Fig. 7). For cosmologies in the bulk of the parameter space, such as M5, M6, M7, and M9, despite a lower classification accuracy, the misclassified clusters are more evenly distributed on both sides of the parameter space, hence producing a more balanced parameter prediction, with a smaller bias. This can be seen in Fig. 7, where the accuracy of the prediction of the four parameters of M5, M6, M7, and M9 is obviously better than that of other models (see also contour plots in Appendix B).

We conclude that the MLCCA algorithm works better for cosmological predictions in the center of the sampled parameter space. More precisely, for a specific cosmological model, the MLCCA can efficiently recover the true cosmological parameters, provided that the training set, made by a series of multi-cosmology hydro-simulations, evenly covers the cosmological parameter space around the true cosmology. This represents the main results of this paper as it strongly suggests increasing the number of cosmologies covered by large-volume, mid-resolution hydro-simulations, to fully apply this method to real data in the future.

5. Robustness and systematics

In the previous sections, we demonstrated the ability of the MLCCA to recover the cosmological parameters by giving a mock catalog of 700 clusters randomly distributed in redshift, for which seven specific observational quantities are given. In this section, we want to check the robustness of this result and discuss the impact of some assumptions made in our analysis and by the properties of the simulations adopted. To be more specific we will consider: 1) the ability of the MLCCA to predict the cosmology of the test sample in the case this is not covered in the training sample, in fact by testing the capability to interpolate between different cosmologies in a grid of parameters; 2) the accuracy of the MLCCA predictions excluding some relevant features, in particular, the total mass; 3) the impact of the size of the measurement errors; 4) the impact of the simulation resolution.

5.1. Excluding a certain cosmology

The cosmological parameters of the real Universe may not be the same as any of the cosmological models in a given simulation

[5] M1 is also close to the minimum of the Ω_b and has a large h_0, so these parameters are also biased.

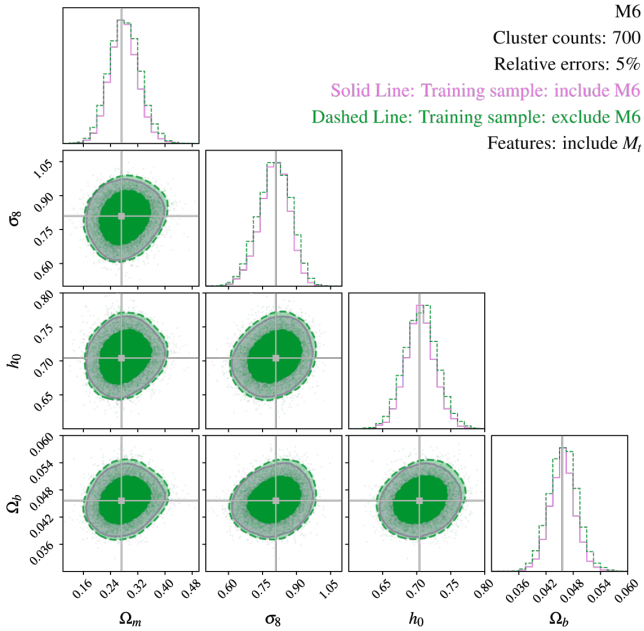

Fig. 9. Parameter constraints for M6 mock catalog including or excluding the M6 cosmology. This graph is the same type as Fig. 8. In both cases, all cosmological parameters are in the 1σ region, indicating that our method has the potential to be applied to the cosmology where each cosmological parameter is roughly located in the center of the parameter space of the training sample, but the specific configuration is unknown.

set. In this case, we need to check if the ML trained with various existing cosmological models can still accurately predict a model that has not been directly learned before. In Fig. 9, we show the distribution of the predicted cosmological parameters from a mock catalog from M6 using an MLCCA trained on all the cosmologies in Table 1, except M6 itself. The predicted values are $(0.281^{+0.046}_{-0.044}, 0.805^{+0.070}_{-0.075}, 0.707^{+0.023}_{-0.024}, 0.0458^{+0.0030}_{-0.0032})$, respectively. They are all consistent with the true values of $(\Omega_m, \sigma_8, h_0, \Omega_b)$ of M6, which are $(0.272, 0.809, 0.704, 0.0456)$, within the estimated errors. This is a remarkable result, showing the ability of the MLCCA to interpolate even over a sparse grid of simulations around the true cosmology the test sample belongs.

5.2. Excluding a certain feature

Ensemble algorithms based on tree models are commonly used to measure the feature importance. This evaluates the influence of features on the final model accuracy and loss. However, this does not give any information on how the features are related to the final prediction results. To measure the impact of the individual features in the final predictions, we adopt a more direct experiment-based approach, by comparing the performance of the model retrained after excluding a certain feature with the performance of the model including the full set of features.

In Fig. 10, we report the variation in the percentage of the mean Accuracy and Logloss over the 5-fold cross-validation process, by excluding each of the features in turn. It is evident, that the "mass features" (i.e., M_*, M_g, and M_t) are the ones most affecting the results. For example, excluding stellar mass M_* will cause a 35% reduction in mean accuracy and a 33% increase in mean logloss. Excluding the gas mass, M_g, the accuracy is reduced by 20% and the logloss increased by 18%, while without the total mass M_t, the accuracy is reduced by 14% and the logloss

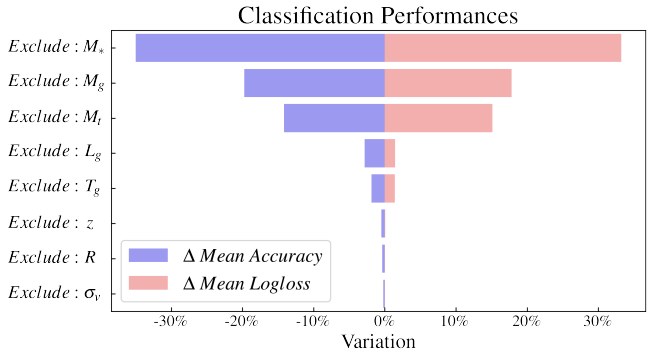

Fig. 10. Comparisons between the performance of the model retrained after excluding a certain feature and the performance of the model before exclusion. The red bars and purple bars represent the percentage change in mean accuracy and mean logloss during the 5-fold cross-validation process respectively. As can be seen, stellar mass, M_*, and gas mass, M_g, have the most substantial impact on the overall performance of the classifier, indicating their crucial importance in MLCCA inference.

increased 15%. On the other hand, excluding the gas luminosity L_g or the gas temperature T_g would not affect the accuracy or logloss by more than 3%. The redshift z, the radius R, and the velocity dispersion σ_v, surprisingly rank the lowest with the combined influence on the overall results amounting only to $\sim$1%. This is likely because most of the information encoded in these features is also contained in the other features above (e.g., σ_v is a proxy of the total mass). However, we need to remark on two facts here. First, this feature importance analysis is related to the simultaneous constraints of all the cosmological parameters together, while possibly the individual parameters can be more sensitive to a certain feature (e.g., h_0 being more sensitive to M_*[6] and z). This is a test that is beyond the purposes of the current paper and we will address it in forthcoming analyses. Second, this feature importance is related to the classification, which is not related to the ability to constrain the cosmology, as stressed above. Hence, we need to check if the absence of an important feature in classification can yet allow us to recover true cosmology.

In Fig. 11, as an example, we show the results of excluding total mass M_t from the list of the features used to train and predict the cosmological parameters for M6 (our reference cosmology). The reason to check the impact of the absence of the total mass among the catalog features is that the mass is among the more uncertain quantities to estimate from observations (see Sect. 1). In this case, the confidence contours are still quite similar to the case of including M_t, except for the Ω_m contours and posterior probability, which look more broadened. For M6, again, the predicted values for $(\Omega_m, \sigma_8, h_0, \Omega_b)$ are $(0.274^{+0.048}_{-0.045}, 0.802^{+0.061}_{-0.065}, 0.704^{+0.021}_{-0.021}, 0.0454^{+0.0028}_{-0.0028})$, against the true values of M6, that are $(0.272, 0.809, 0.704, 0.0456)$. This indicates that excluding M_t would somehow affect the accuracy of classification, but produce a limited impact on the parameter constraints, except for the Ω_m precision. This means that the cosmological information about all parameters is still encoded in some other features that are directly accessible in observation (such as stellar mass M_* and gas mass M_g). Therefore, this experiment shows, specifically, that AI can help extract information

[6] Despite stellar ages are not included in the simulation features, it is possible that the assembly of stellar masses in clusters is tightly correlated with the age of the universe, with stars being cosmological clocks (see e.g., Jimenez & Loeb 2002).

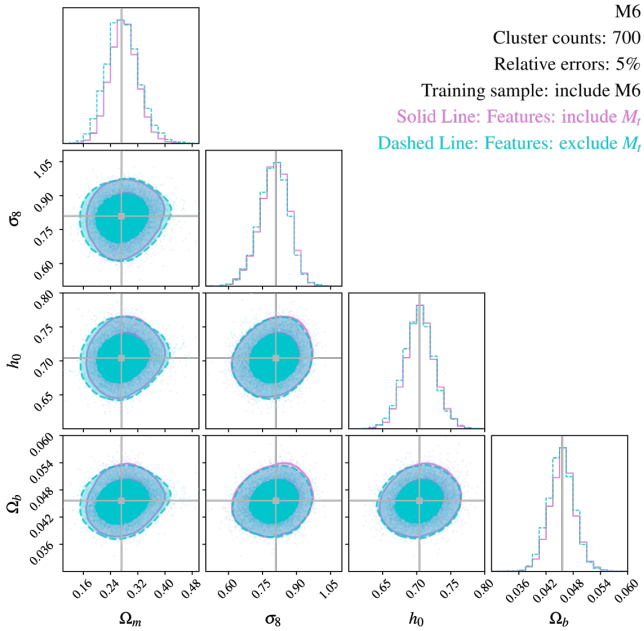

Fig. 11. Parameter constraints for M6 mock catalog including or excluding the total mass feature in the training sample. This graph is the same type as Fig. 8. In both cases, all cosmological parameters are in the 1σ region, indicating that our method could achieve high limiting accuracy for cosmological parameters without using total mass.

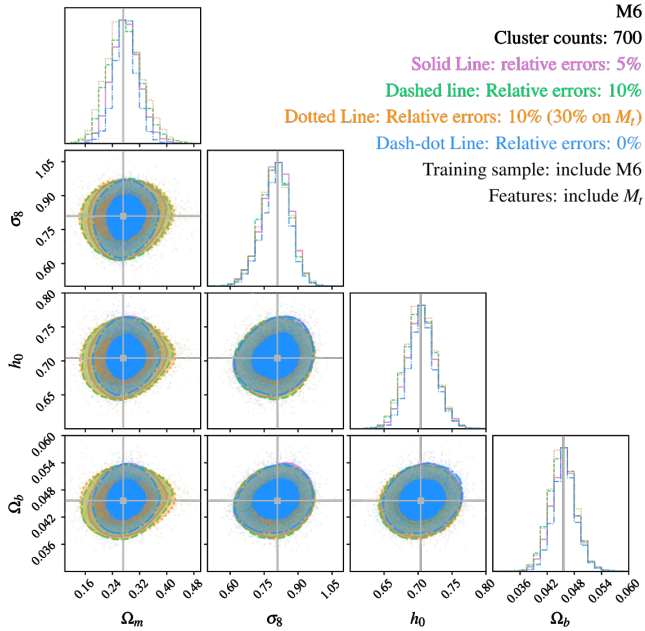

Fig. 12. Parameter constraining results for M6 mock catalog with different error setting. This graph is the same type as Fig. 8. The model was trained by the training sample adding 0% (dash-dot line), 5% (solid line), and 10% (dashed line) errors on M_*, M_g, M_t, L_g, T_g, R, and σ_v, and adding 10% errors on M_*, M_g, L_g, T_g, R, and σ_v while adding 30% errors on M_t (dotted line). In these four cases, all cosmological parameters are in the 1σ region, indicating that our method has relatively good robustness to the error degree of features.

from multiwavelength features to infer cosmological parameters even without the total mass.

5.3. The impact of the measurement errors

To take into account the measurement errors of cluster features in the real observation, we added 5% Gaussian errors to the simulation data. As discussed in Sect. 2.3, this was a conservative choice for most of the features, or even optimistic for others (see e.g., the total mass from weak lensing). Hence, we are interested to consider a wider range of statistical errors and check whether, by improving the precision of observations (smaller errors), one obtains tighter constraints on classification and cosmological parameter inferences and vice versa for larger observational errors. The uncertainties on the observed quantities equally impact traditional methods, e.g., the mass function of galaxy clusters, where higher or lower accuracy of cluster features produces more or less accurate cosmological results. In Fig. 12, we show the confidence intervals for the prediction of the four cosmological parameters where we consider the extreme case of 0% errors for all features, which provides information on the uncertainties inherent to the ML model. We also consider the pessimistic cases where we assume 10% errors for all features or 30% errors for the M_t and 10% on other observables (see Sect. 2.3). These are shown against the reference case with 5% errors in overall quantities. The predictions for the peaks are almost identical in all these cases, implying a rather resilient accuracy, while the confidence contours are slightly shrunken in the 0% case and expanded in the 10% case, as expected, for all parameters. However, the 0% errors allow an improvement in terms of accuracy of Ω_m, by ~21%, which is reasonably good, but not significant improvements for the other parameters. On the other hand, for the case of 10% errors, we observe a significant degradation of the Ω_m precision (~23% larger than the 5% error case), but, again, no sensible changes for the other parameters, which are recovered with similar precision. Finally, the extreme case of 30% on the M_t does not show a catastrophic impact on the size of the contours of Ω_m, that increases by ~38% with respect to the 5% error case and by ~12% with respect to the 10% error case. This is possibly due to the fact that the scaling relations, to which Ω_m is sensitive, are more tightly distributed with respect to the ones the other parameters are sensitive to. Hence larger measurement errors increase the overlap among scaling relations sensitive to Ω_m more than the ones of the other parameters. Finally, we can also argue that the measurement errors of M_t have little effect on the cosmological parameter predictions, because this is a "less important feature" than M_* and M_g and the model performs well when M_* and M_g have 10% errors and M_t is much noisier than other features. Interestingly, we find that either including noisy M_t estimates (as just discussed) or excluding M_t from the catalogs (as discussed in Sect. 5.2), leads to similar results.

5.4. The impact of the simulation resolution

In the previous section, we discussed measurement errors as a basic implementation of observational realism. This latter element has larger ramifications than simple measurement errors and it tracks back to the definition of the observational quantities in simulations and how the observational conditions can affect the inferred physical measurements in synthetic datasets (see e.g., Bottrell et al. 2019; Tang et al. 2021). However, there are other profound implications related to the technical aspects of simulations and the way these are calibrated to observations, that might affect the proper training of ML tools and impact their application to real data. For instance, one problem is the "resolution convergence." It is known that any given property of a

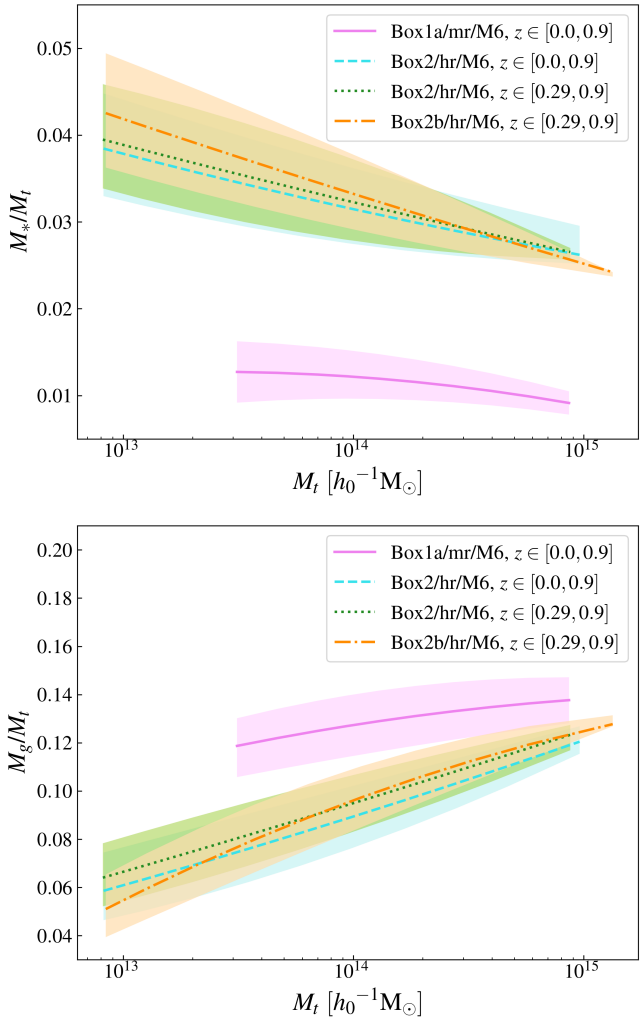

Fig. 13. Stellar mass ratio M_*/M_t (the upper panel) and gas mass ratio M_g/M_t (the lower panel) as a function of the total mass M_t. Different colors represent different simulation boxes and different redshift intervals. Box1a has a medium-resolution, while Box2 and Box2b have a high-resolution.

simulated halo may not be fully converged at any given mass or spatial resolution (Weinberger et al. 2017; Pillepich et al. 2018b). Due to the different impact of the sub-grid physics (e.g., Colín et al. 2010), both stellar masses and star formation rates can increase with better resolution for dark matter haloes of a fixed mass. This has been proven, e.g., in TNG simulations[7] (Pillepich et al. 2018a).

To check this effect in Magneticum simulations, we have derived the distribution of high-resolution (hr) cluster features with respect to the mid-resolution (hr) simulations. M1−M13 all have hr simulations in the hr box, Box1a. For M6 (the fiducial cosmology considered in Magneticum), additional simulations are available in hr boxes, for instance, Box2 and Box2b. The sizes of Box1a/mr, Box2/hr and Box2b/hr are ~896 h_0^{-1} Mpc, ~352 h_0^{-1} Mpc and ~640 h_0^{-1} Mpc, respectively. More details of these three boxes can be found at the Magneticum website[8].

In Fig. 13, we calculate the stellar mass ratio M_*/M_t (top) and gas mass ratio M_g/M_t (bottom) as the function of total mass M_t both for the M6 medium-resolution simulations (Box1a/mr) and two high-resolution simulation boxes (Box2/hr, Box2b/hr). We stress here, in particular, that the Box2/hr/M6 simulation not only shares the same cosmology and feedback but also covers the same redshift interval of Box1a/mr/M6, while the Box2b/hr/M6 covers a higher redshift range ($z \geq 0.29$). As expected, the stellar mass ratios and gas mass ratios are quite sensitive to the resolution levels. The higher the resolution, the smaller the stellar mass and gas mass at a fixed total mass. Statistically, for clusters with masses between 2×10^{13} $h^{-1}M_\odot$ and 10^{15} $h^{-1}M_\odot$ in M6 cosmology, stellar mass averages 3% of the total mass at high-resolution, while at a medium-resolution, this percentage decreases to 1.2%. On the other hand, the gas mass ratio seems to rise with decreasing resolution with about 10% at high-resolution and 13% at medium-resolution. This is consistent with what has been found in TNG simulations for haloes with total mass $\log M_t/M_\odot > 14$ (Pillepich et al. 2018a). We also observe that different volumes (Box2/hr and Box2b/hr), show a sensitive tilt. To check if this is due to the lack of low-redshift data for Box2b (which is limited to $z \geq 0.29$) or to cosmic variance, in Fig. 13 we also add the stellar and gas mass fractions for Box2/hr for redshifts $z \geq 0.29$ only, consistently with Box2b/hr. As we can see this latter is slightly offset with respect to the case including clusters down to $z = 0$, hence we conclude that the tilt possibly comes from cosmic variance. We notice though that the larger variance comes from $M_t < 10^{14} h^{-1} M_\odot$ and is of the order of 1%.

In general, other features in hr, such as gas luminosity and temperature, also show deviations from those in hr. This raises the question of which resolution should be taken as the best representation of reality. This is certainly a question we will need to address when applying the MLCCA to real data, as we will need to ensure that the algorithm is trained over simulations for which the calibration of the relevant scaling relations and resolution do conspire to match observations. We anticipate here that this is not a simple task as observations do not provide an obvious indication about the "ground truth," having clusters a stellar mass fraction varying from 0.5% to 3% (see e.g., Chiu et al. 2018), i.e., a scatter well beyond either the hr or hr relations in Fig. 13. The obvious warning emerging from the question above is that we need to keep the subgrid-physics under control in simulations to produce predictions, given a baryon physics recipe, resolution-independent (see e.g., Murante et al. 2015). However, in the perspective of our proof-of-concept experiment, this yet important "realism" aspect is irrelevant as long as the training and the test sample are extracted from the same knowledge base provided by the same simulations with the same stellar mass or gas mass fraction. While it becomes relevant if one needs to train on a simulation with a resolution different from the one from which the test sample is extracted. In this case, one can use a "rescaling procedure" (see e.g., Pillepich et al. 2018a) by applying a resolution correction factor to align the physical quantities from different resolution boxes. Of course, this is a workaround needed in order to compensate resolution effect and make the simulation predictions consistent at all resolution levels. From the point of view of this work, there is no particular reason why one wants to mix simulations of different resolutions, however, it can still be useful to check if the naif rescaling procedure makes the MLCCA predictions insensitive to the resolution correction.

Indeed, according to the "independent identically distribution" hypothesis in ML inferences, any model can have reliable predictions only when the feature distributions of the test sample are comparable with those of the training sample. Hence, if we use a test sample from hr simulations, we expect the MLCCA trained on hr to fail, because the net effect of the resolution is to

[7] https://www.tng-project.org/
[8] http://magneticum.org/simulations.html

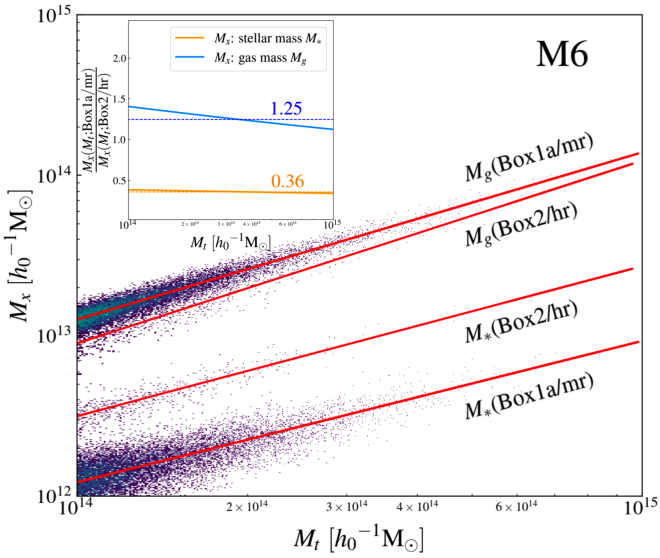

Fig. 14. $M_g \sim M_t$ and $M_g \sim M_t$ relationships in the medium-resolution simulation (Box1a/mr) and high-resolution simulation (Box2/hr). The subplot in the upper left corner represents the conversion coefficients of stellar mass (blue) and gas mass (orange) between medium-resolution and high-resolution for a fixed total mass, corresponding to Eq. (8).

scale up or down the $M_* - M_t$ and the $M_g - M_t$ relations, similarly to what the different cosmology do at a fixed resolution (see Fig. 3). This is a general problem that we would also face using real data where, in the case of the nonuniform definition of the observed quantities, the real features and the training feature can have deviations even if they come from the same cosmology, as the hr and hr mock catalogs do as shown in Fig. 13.

Following Pillepich et al. (2018a), we adopt a heuristic correction to convert hr cluster features into their hr versions that approximately reproduce the hr training sample. First, from both Box1a/mr/M6 and Box2/hr/M6, we select clusters with the total mass within $10^{14} \sim 10^{15}\ h^{-1}M_\odot$ to mitigate the effects of resolution on a too wide mass range and assume a constant correction. Second, despite different Box2/hr/M6 features explicitly varying from those of Box1a/mr/M6, we only adjust two of the most important features (M_* and M_g as from Fig. 10), conservatively. In Fig. 14, we can see how the correlation of these two quantities changes as a function of M_t in the different boxes and resolutions. We can adopt a mass-modulated conversion strategy to derive a conversion coefficient that can reflect the resolution-induced feature drift. For brevity, we assume that the conversion coefficient is the average of multiples of M_x/mr and M_x/hr obtained at each fixed M_t. Accordingly, for clusters whose M_t within $10^{14} \sim 10^{15}\ h^{-1}M_\odot$, their hypothetical hr versions of M_x (M_* or M_g) can be approximately obtained by multiplying the conversion coefficient α and their original hr versions as follows:

$$M_x(M_t;\mathrm{Box2/mr}) \approx \left\langle \frac{M_x(\mathrm{M_t};\mathrm{Box1a/mr})}{M_x(\mathrm{M_t};\mathrm{Box2/hr})} \right\rangle \cdot M_x(M_t;\mathrm{Box2/hr}) \approx \alpha \cdot M_x(M_t;\mathrm{Box2/hr}). \quad (8)$$

From Fig. 14, we find that the best fit α is 0.36 and 1.25 for M_* and M_g, respectively. We further apply these two coefficients to derive hypothetical Box2/mr clusters and have checked that their three features (M_*, M_g, and M_t) finally use these features to make the cosmological parameter predictions.

Figure 15 shows parameter constraints from M_*, M_g, and M_t for Box1a/mr and Box2/hr (converted to Box2/mr version).

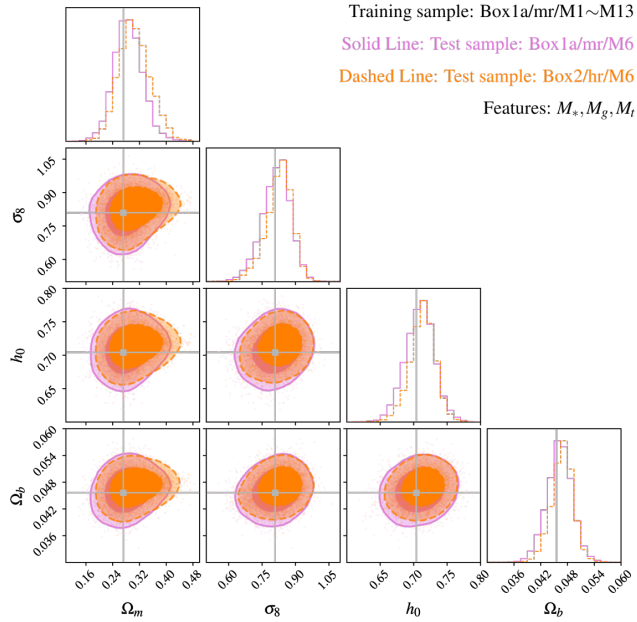

Fig. 15. Parameter constraints for medium-resolution M6 mock catalog (Box1a/mr/M6, solid line) and high-resolution M6 mock catalog (Box2/hr/M6, dashed line). The model was trained by medium-resolution training sample. This graph is the same type as Fig. 8. In both cases, we guarantee that the total mass of the cluster ranges from 10^{14} to $10^{15}\ h^{-1}M_\odot$. It can be seen that the high-resolution prediction values are higher than the real values, but all cosmological parameters are still in the 1σ region, indicating that our method has certain application potential for different resolution cosmology.

Compared to the true cosmological configuration of M6 (Ω_m : $0.272, \sigma_8$: $0.809, h_0$: $0.704, \Omega_b$: 0.0456), the predictions for Box1a/mr and Box2/hr are:

$$(0.286^{+0.043}_{-0.040}, 0.818^{+0.056}_{-0.068}, 0.710^{+0.020}_{-0.023}, 0.0463^{+0.0027}_{-0.0029}),$$
$$(0.303^{+0.050}_{-0.043}, 0.832^{+0.050}_{-0.063}, 0.714^{+0.017}_{-0.019}, 0.0470^{+0.0022}_{-0.0025}),$$

respectively, i.e., yet consistent within the errors.

We find the overall predictions made over Box2/hr are similar to that of Box1a/mr, especially for h_0 and Ω_b, which demonstrates that our conversion strategy maintains most of the inner-correlations among the three mass quantities (M_*, M_g, and M_t). However, MLCCA overestimates all parameters of both boxes, especially the Ω_m and σ_8. This residual discrepancy might come from the fact that changing M_* and M_g, without changing the total mass, substantially alters the baryon fraction of the sample and, intrinsically, the underlying cosmology of the cluster catalog. This test shows that we cannot straightforwardly generalize the results, obtained from mid-resolution to high-resolution, as this would imply corrections on the features that might introduce biases in the predicted cosmology. This suggests that to avoid systematics, one should train the algorithm using features from numerically converged simulations.

6. Summary and conclusions

In this paper, we have introduced and tested a first proof-of-concept ML pipeline which is able to predict the cosmological parameters starting from mock catalogs of galaxy clusters' physical parameters, namely the stellar mass, M_*, gas mass, M_g, total mass, M_t, gas luminosity, L_g, and temperature, T_g, sizes, R_{500c}, velocity dispersion, σ_v, and redshift, z. These are typical observables (or features) we expect to collect from current and future

imaging surveys in optical and NIR (e.g., *Rubin*/LSST, CSST, and *Euclid*), spectroscopical surveys (e.g., DESI and 4MOST), and X-ray surveys (e.g., eROSITA). We have used the mock catalogs of galaxy clusters extracted from the multi-cosmology set of Magneticum hydrodynamical simulations, which spans a limited volume in the $(\Omega_m, \sigma_8, h_0, \Omega_b)$ parameter space, centered around the WMAP7 cosmology. There are 15 different simulations available, which also include some variations of the feedback recipe from AGN and supernovae. We used only 13 of them, excluding twocosmologies with too few clusters to use as training samples, and also skipped the inclusion of multi-feedback for this first test, as there were onlyfour simulations with two feedback recipes available. Again, these are too few to be used for a meaningful test. The mock catalogs, including measurement error, are used to train an optimized Light Gradient Boosting machine (LGB) network to classify the cluster catalogs and predict the cosmological parameters. Based on this optimized LGB network, we have built a Machine Learning Cluster Cosmology Pipeline (MLCCA). The MLCCA has proven to be very effective in predicting the right set of cosmological parameters although the classification of the individual clusters to belong to the right cosmology suffers from the similarity of the scaling relations of close cosmologies. Here below, we summarize the main results of the application of the MLCCA to mock catalogs of 700 clusters from different cosmologies:

1. The MLCCA can accurately predict the true cosmological parameters corresponding to the cosmological simulation the catalogs are drawn from. Despite the limited coverage in the parameter space, for cosmological models in the center of the parameter space, the classification recall rate is between ~0.2 and 0.4, but the predictions of the cosmological parameters are tighter. Typical 1σ level are 14% for Ω_m, 8% for σ_8, 3% for h_0, 6% for Ω_b. For cosmological models at the edge of parametric space, the classification accuracy increases because there is not any confusion with cluster properties from close models, but the cosmological parameters are slightly biased. This is clearly a "border effect" due to the training sample, rather than the true underperformance of the MLCCA. This leads us to conclude that more mid-resolution hydro-simulation Magneticum-like are needed to make the MLCCA effectively applicable to real data.
2. In order to fully check the performance of the MLCCA and, in particular, the ability to extrapolate to cosmologies that are not included in the training sample (this is a situation that might happen also if one uses a regular grid of cosmologies), we have tested the ability to recover the cosmology over a mock catalog taken from a cosmology (specifically we tried M6 and M7) that was not included in the training set and found that the MLCCA can recover the cosmological parameter with comparable accuracy and precision as the case where the training contains the mock catalog cosmology.
3. We have tested the impact of the measurement errors, particularly how the recall rate of the classifier and the uncertainties on the cosmological parameters would be affected. We have found that for errors of the order of 2%, the 1σ contours are shrunk by ~18%, while for larger errors, i.e., 10%, only Ω_m show large degradation of the precision with typical 1σ contours widened by up to ~20–40%. We note that the current accuracy can be strongly affected by two main factors: 1) the limited size of the training sample, and 2) the limited number of the it mock catalog sizes, which we need to check with larger volumes of multi-cosmology simulations.
4. We have tested the resilience of the MLCCA for missing features, i.e., in case cluster catalogs do not contain one or more of the observations used for the main experiment as at point 1) above. Also in this case, the MLCCA can correctly recover the cosmological parameter even if the mass features are missing, despite the fact that these are the most important features for the classification. We have understood that by the ability of the ML tool to still extract relevant cosmological information from the scaling relations involving all other features. Among all features, stellar mass and gas mass have the greatest weight on accuracy for the classification.
5. Finally, we have checked the effect of simulation resolution, as this latter produces a sensitive impact on the stellar and gas mass of clusters, due to the different effects of the sub-grid physics (Pillepich et al. 2018a). In particular, we have tested whether simple "rescaling" of the major cluster features can leave the predictions of the MLCCA unaltered and found that if one limits to only the major baryonic mass features (stars and gas) without also recorrecting the total mass, one ends with systematic effects. This calls for effective strategies to improve the sub-grid physics treatment in hydrodynamical simulations to make their predictions more stable toward the change of resolution.

This first application of cosmological inference from ML based on galaxy clusters shows that these tools have a rather strong predictive power, by efficiently cross-correlating features among different cosmological predictions. This is very promising for future applications making use of finer sampling of the cosmological and galaxy formation parameter space in future multi-cosmology hydrodynamical simulation runs. And in the long term, this could help to fully exploit multiwavelength observations from current and future surveys, to gain a more profound understanding of the true universe model.

This work follows a line of experiments trying to extract cosmological information from observational data using ML tools applied to multi-cosmology simulations. Villaescusa-Navarro et al. (2022) use the internal properties of a single galaxy simulated by the CAMELS project[9] to predict cosmological parameters, especially Ω_m and σ_8. Their ML model could infer the value of Ω_m with a precision of $\delta\Omega_m/\Omega_m \simeq 10$–15% (with a possible explanation that Ω_m could affect the dark matter content of galaxies and then further result in a unique change in the observables' manifold). However, they could not infer σ_8 due to the small non scale of galaxies. Further works by CAMELS include quantifying the robustness of the ML model by testing on galaxies from different codes (Echeverri-Rojas et al. 2023), improving the inference on cosmological parameters by enlarging the simulation sets (Ni et al. 2023), etc.

In our work, we show that galaxy clusters are very powerful in inferring cosmological parameters, mainly because of the stronger connection with large-scale structure formation, which is more sensitive to cosmology. Among the cluster features that we use, the underlying halo mass function has been widely proved to constrain Ω_m and σ_8, the gas mass (and baryonic mass in general) has been proved to sensitively depend on Ω_b, while the stellar mass, velocity dispersion, and gas temperature have been proved to sensitively depend on h_0. In the next analyses, we expect to apply the MLCCA to upcoming sets of mid-resolution, large-volume hydrodynamical simulations, considering a wider range of cosmologies and, for each of them, different feedback recipes, to finally test the predictions of the cosmological parameters and baryonic physics at the same time. This will

[9] https://www.camel-simulations.org/

eventually allow us to move toward the first application to real data.

Acknowledgements. N.R.N. acknowledges financial support from the Research Fund for International Scholars of the National Science Foundation of China (NSFC), grant no. 12150710511 and from China Manned Space project no. CMS-CSST-2021-A01. S.B. acknowledges support from Fondazione ICSC – Spoke 3 Astrophysics and Cosmos Observations – National Recovery and Resilience Plan (PNRR) Project ID CN_00000013 "Italian Research Center on High-Performance Computing, Big Data and Quantum Computing" funded by MUR Missione 4 Componente 2 Investimento 1.4: "Potenziamento strutture di ricerca e creazione di campioni nazionali di R&S (M4C2-19)" – Next Generation EU (NGEU) and INFN InDark Grant. X.L. acknowledges the science research grants from the China Manned Space Project with no. CMS-CSST-2021-A03, no. CMS-CSST-2021-B01. W.L. acknowledges financial support from NSFC, grant no. 12073089. K.D. acknowledges support by the COMPLEX project from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program grant agreement ERC-2019-AdG 882679 as well as support by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC-2094 – 390783311. The calculations for the hydrodynamical simulations were carried out at the Leibniz Supercomputer Center (LRZ) under project pr83li (Magneticum).

References

Abbott, T. M. C., Abdalla, F. B., Alarcon, A., et al. 2018, Phys. Rev. D, 98, 043526
Abbott, T. M. C., Aguena, M., Alarcon, A., et al. 2020, Phys. Rev. D, 102, 023509
Abdullah, M. H., Klypin, A., & Wilson, G. 2020, ApJ, 901, 90
Adami, C., Giles, P., Koulouridis, E., et al. 2018, A&A, 620, A5
Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. 2019, in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (New York: Association for Computing Machinery), 785
Allen, S. W., Evrard, A. E., & Mantz, A. B. 2011, ARA&A, 49, 409
Armitage, T. J., Kay, S. T., & Barnes, D. J. 2019, MNRAS, 484, 1526
Bahar, Y. E., Bulbul, E., Clerc, N., et al. 2022, A&A, 661, A7
Beck, A. M., Murante, G., Arth, A., et al. 2016, MNRAS, 455, 2110
Biviano, A., Rosati, P., Balestra, I., et al. 2013, A&A, 558, A1
Bleem, L. E., Stalder, B., de Haan, T., et al. 2015, ApJS, 216, 27
Bocquet, S., Dietrich, J. P., Schrabback, T., et al. 2019, ApJ, 878, 55
Bocquet, S., Heitmann, K., Habib, S., et al. 2020, ApJ, 901, 5
Böhringer, H., Chon, G., Collins, C. A., et al. 2013, A&A, 555, A30
Borgani, S., & Guzzo, L. 2001, Nature, 409, 39
Borgani, S., Girardi, M., Carlberg, R. G., Yee, H. K. C., & Ellingson, E. 1999, ApJ, 527, 561
Borgani, S., Murante, G., Springel, V., et al. 2004, MNRAS, 348, 1078
Bottrell, C., Hani, M. H., Teimoorinia, H., et al. 2019, MNRAS, 490, 5390
Boylan-Kolchin, M., Springel, V., White, S. D. M., Jenkins, A., & Lemson, G. 2009, MNRAS, 398, 1150
Breiman, L. 2001, Mach. Learn., 45, 5
Bryan, G. L., & Norman, M. L. 1998, ApJ, 495, 80
Bulbul, E., Chiu, I. N., Mohr, J. J., et al. 2019, ApJ, 871, 50
Chawak, C., Villaescusa-Navarro, F., Echeverri Rojas, N., et al. 2023, arXiv e-prints [arXiv:2309.12048]
Chen, T., & Guestrin, C. 2016, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 785
Chiu, I., Mohr, J. J., McDonald, M., et al. 2018, MNRAS, 478, 3072
Chiu, I. N., Klein, M., Mohr, J., & Bocquet, S. 2023, MNRAS, 522, 1601
Cohn, J. D., & Battaglia, N. 2020, MNRAS, 491, 1575
Colín, P., Avila-Reese, V., Vázquez-Semadeni, E., Valenzuela, O., & Ceverino, D. 2010, ApJ, 713, 535
Costanzi, M., Rozo, E., Simet, M., et al. 2019, MNRAS, 488, 4779
Cui, W., Borgani, S., Dolag, K., Murante, G., & Tornatore, L. 2012, MNRAS, 423, 2279
Dalton, G., Trager, S. C., Abrams, D. C., et al. 2012, SPIE Conf. Ser., 8446, 84460P
de Andres, D., Cui, W., Ruppin, F., et al. 2022, Nat. Astron., 6, 1325
de Jong, R. S., Agertz, O., Berbel, A. A., et al. 2019, The Messenger, 175, 3
DESI Collaboration (Aghamousa, A., et al.) 2016, arXiv e-prints [arXiv:1611.00036]
Dietrich, J. P., Bocquet, S., Schrabback, T., et al. 2019, MNRAS, 483, 2871
Dietterich, T. G. 2000, in International Workshop on Multiple Classifier Systems
Dolag, K., Borgani, S., Murante, G., & Springel, V. 2009, MNRAS, 399, 497
Dolag, K., Komatsu, E., & Sunyaev, R. 2016, MNRAS, 463, 1797
Echeverri-Rojas, N., Villaescusa-Navarro, F., Chawak, C., et al. 2023, ApJ, 954, 125
Euclid Collaboration (Giocoli, C., et al.) 2024, A&A, 681, A67
Fabjan, D., Borgani, S., Tornatore, L., et al. 2010, MNRAS, 401, 1670
Falco, M., Mamon, G. A., Wojtak, R., Hansen, S. H., & Gottlöber, S. 2013, MNRAS, 436, 2639
Geurts, P., Ernst, D., & Wehenkel, L. 2006, Mach. Learn., 63, 3
Giocoli, C., Marulli, F., Moscardini, L., et al. 2021, A&A, 653, A19
Henson, M. A., Barnes, D. J., Kay, S. T., McCarthy, I. G., & Schaye, J. 2017, MNRAS, 465, 3361
Heymans, C., Tröster, T., Asgari, M., et al. 2021, A&A, 646, A140
Hildebrandt, H., Viola, M., Heymans, C., et al. 2017, MNRAS, 465, 1454
Hilton, M., Sifón, C., Naess, S., et al. 2021, ApJS, 253, 3
Hirschmann, M., Dolag, K., Saro, A., et al. 2014, MNRAS, 442, 2304
Ho, M., Rau, M. M., Ntampaka, M., et al. 2019, ApJ, 887, 25
Hoekstra, H., Herbonnet, R., Muzzin, A., et al. 2015, MNRAS, 449, 685
Ingoglia, L., Covone, G., Sereno, M., et al. 2022, MNRAS, 511, 1484
Jimenez, R., & Loeb, A. 2002, ApJ, 573, 37
Kobayashi, Y., Nishimichi, T., Takada, M., & Miyatake, H. 2022, Phys. Rev. D, 105, 083517
Kodi Ramanah, D., Wojtak, R., Ansari, Z., Gall, C., & Hjorth, J. 2020, MNRAS, 499, 1985
Kodi Ramanah, D., Wojtak, R., & Arendse, N. 2021, MNRAS, 501, 4080
Komatsu, E., Smith, K. M., Dunkley, J., et al. 2011, ApJS, 192, 18
Kravtsov, A. V., & Borgani, S. 2012, ARA&A, 50, 353
Lesci, G. F., Marulli, F., Moscardini, L., et al. 2022a, A&A, 659, A88
Lesci, G. F., Nanni, L., Marulli, F., et al. 2022b, A&A, 665, A100
Liu, A., Bulbul, E., Ghirardini, V., et al. 2022, A&A, 661, A2
Łokas, E. L., Wojtak, R., Gottlöber, S., Mamon, G. A., & Prada, F. 2006, MNRAS, 367, 1463
Mantz, A., Allen, S. W., Rapetti, D., & Ebeling, H. 2010, MNRAS, 406, 1759
Mantz, A. B., Allen, S. W., Morris, R. G., et al. 2016, MNRAS, 463, 3582
Maturi, M., Bellagamba, F., Radovich, M., et al. 2019, MNRAS, 485, 498
Melchior, P., Gruen, D., McClintock, T., et al. 2017, MNRAS, 469, 4899
Munari, E., Biviano, A., & Mamon, G. A. 2014, A&A, 566, A68
Murante, G., Monaco, P., Borgani, S., et al. 2015, MNRAS, 447, 178
Navarro, J. F., Frenk, C. S., & White, S. D. M. 1996, ApJ, 462, 563
Ni, Y., Genel, S., Anglés-Alcázar, D., et al. 2023, ApJ, 959, 136
Ntampaka, M., Trac, H., Sutherland, D. J., et al. 2015, ApJ, 803, 50
Pacaud, F., Pierre, M., Melin, J. B., et al. 2018, A&A, 620, A10
Pillepich, A., Nelson, D., Hernquist, L., et al. 2018a, MNRAS, 475, 648
Pillepich, A., Springel, V., Nelson, D., et al. 2018b, MNRAS, 473, 4077
Planck Collaboration XXIV. 2016, A&A, 594, A24
Planck Collaboration VI. 2020, A&A, 641, A6
Prati, R. C., Batista, G., & Monard, M. C. 2004, in Proceedings of the 4th Indian International Conference on Artificial Intelligence, IICAI 2009, Tumkur, Karnataka, India, December 16–18, 2009
Pratt, G. W., Croston, J. H., Arnaud, M., & Böhringer, H. 2009, A&A, 498, 361
Pratt, G. W., Arnaud, M., Biviano, A., et al. 2019, Space Sci. Rev., 215, 25
Qi, M. 2017, in Neural Information Processing Systems
Ragagnin, A., Tchipev, N., Bader, M., Dolag, K., & Hammer, N. J. 2016, in Advances in Parallel Computing, 411
Ragagnin, A., Fumagalli, A., Castro, T., et al. 2023, A&A, 675, A77
Remus, R.-S., Dolag, K., Naab, T., et al. 2017, MNRAS, 464, 3742
Rykoff, E. S., Rozo, E., Hollowood, D., et al. 2016, ApJS, 224, 1
Sereno, M. 2015, MNRAS, 450, 3665
Sereno, M., & Umetsu, K. 2011, MNRAS, 416, 3187
Sereno, M., Umetsu, K., Ettori, S., et al. 2018, ApJ, 860, L4
Sereno, M., Umetsu, K., Ettori, S., et al. 2020, MNRAS, 492, 4528
Singh, P., Saro, A., Costanzi, M., & Dolag, K. 2020, MNRAS, 494, 3728
Springel, V. 2005, MNRAS, 364, 1105
Springel, V., White, S. D. M., Tormen, G., & Kauffmann, G. 2001, MNRAS, 328, 726
Springel, V., Di Matteo, T., & Hernquist, L. 2005a, MNRAS, 361, 776
Springel, V., White, S. D. M., Jenkins, A., et al. 2005b, Nature, 435, 629
Storey-Fisher, K., Tinker, J. L., Zhai, Z., et al. 2024, ApJ, 961, 208
Tang, L., Lin, W., Wang, Y., & Napolitano, N. R. 2021, MNRAS, 508, 3321
Tornatore, L., Borgani, S., Dolag, K., & Matteucci, F. 2007, MNRAS, 382, 1050
Umetsu, K., Zitrin, A., Gruen, D., et al. 2016, ApJ, 821, 116
Umetsu, K., Sereno, M., Lieu, M., et al. 2020, ApJ, 890, 148
van den Busch, J. L., Wright, A. H., Hildebrandt, H., et al. 2022, A&A, 664, A170
Vikhlinin, A., Burenin, R. A., Ebeling, H., et al. 2009a, ApJ, 692, 1033
Vikhlinin, A., Kravtsov, A. V., Burenin, R. A., et al. 2009b, ApJ, 692, 1060
Villaescusa-Navarro, F., Anglés-Alcázar, D., Genel, S., et al. 2021, ApJ, 915, 71
Villaescusa-Navarro, F., Ding, J., Genel, S., et al. 2022, ApJ, 929, 132
Vogelsberger, M., Genel, S., Springel, V., et al. 2014, MNRAS, 444, 1518
Wechsler, R. H., & Tinker, J. L. 2018, ARA&A, 56, 435
Weinberger, R., Springel, V., Hernquist, L., et al. 2017, MNRAS, 465, 3291
Yan, Z., Mead, A. J., Van Waerbeke, L., Hinshaw, G., & McCarthy, I. G. 2020, MNRAS, 499, 3445

Appendix A: Cosmological parameter estimates from classification probability

In this Appendix, we summarize the statistical arguments behind using the classification as a starting point to infer cosmology. In particular, we use n cluster observables $O = \{O_1, O_2, ..O_i...O_n\}$ in a series of cosmological models M_i with corresponding cosmological parameters $\theta_j = (\Omega_m, \sigma_8, h_0, \Omega_b)$. We can start from the assumption that every single cluster carries the information of the cosmology behind the universe it lives in. The expectation of the cosmological parameters for one observation O_i can be assumed to be

$$\mu_i = \sum_j P(\theta_j|O_i)\,\theta_j, \tag{A.1}$$

where $P(\theta_j|O_i)$ is the conditional distribution related to the individual observable O_i. The corresponding error on this expectation is defined by the variance:

$$\sigma_i^2 = \sum_j (\theta_j - \mu_i)^2 P(\theta_j|O_i). \tag{A.2}$$

The observation O can be thought of as a series of measuring processes, then the expectation is

$$\mu = \sum_i^n \mu_i/n, \tag{A.3}$$

and the error is

$$\sigma = \sqrt{\sum_i^n \sigma_i^2/n}. \tag{A.4}$$

Eqs. (A.3), (A.4) represent the main statistics adopted in Sect. 4.3 to estimate the cosmological parameters estimates from independent observations of clusters. To fully define them, we need to define the $P(\theta_j|O_i)$, i.e., the probability of a single cluster observations i to come from a cosmological model j (see also Sect. 3.3.3). In principle, it can be estimated by Bayes probability, as

$$P(\theta_j|O_i) = p(\theta_i)\frac{P(O_i|M_j)}{P(O_i|M)}, \tag{A.5}$$

where $M = M_1 \bigcup M_2 ... \bigcup M_i ... M_m$ is all your m models, or simulations. The prior here can be $p(\theta_i) = 1/m$, which means a flat distribution in the absence of observations. However, it is difficult to find a smooth probability distribution function for $P(O_i|M_j)$ or $P(O_i|M)$ under small m and high-dimension output simulation data.

ML technique provides a good way to find out the best fit $P(O_i|M_j)$. This is possible by training a network with simulation cluster pair $(\theta_j|M_{jk})$, where M_{jk} is one simulation cluster k from a cosmological simulation j, and the training label are set to be $P_{ML}(\theta_j|M_{jk}) = 1$. If a series of simulations cover the real observations, ML can provide a good approximation of it via the best likelihood (P_{ML})

$$P(\theta_j|O_i) \approx P_{ML}(\theta_j|O_i). \tag{A.6}$$

Under the statistical viewpoint of ML, this approximation does not have to be as accurate as possible (due to the cluster degeneracy on different simulations with nearly the same parameters), as long as the accuracy is greater than the prior $p(\theta_i)$. The larger the threshold, the more significant the cosmological information carried out by the individual cluster. Of course, the precision of the method increases as a function of the size of the cluster catalog as Eq. (A.4) show. This means that we cannot perform cosmology with one cluster.

Appendix B: Constraints for other cosmologies

As a continuation of the results presented in Sect. 4.3, we extend our analysis to three additional cases, namely M5, M7, and M9, to further demonstrate the parameter prediction power of our ML method. Our results reveal that the parameter constraints of M7 (Fig. B.2) are similar to that of M6 (Fig. 8). However, M5 (Fig. B.1) and M9 (Fig. B.3) are located further away from the center of the parameter space compared to M6 and M7, resulting in a relatively poorer parameter prediction effect for these cases.

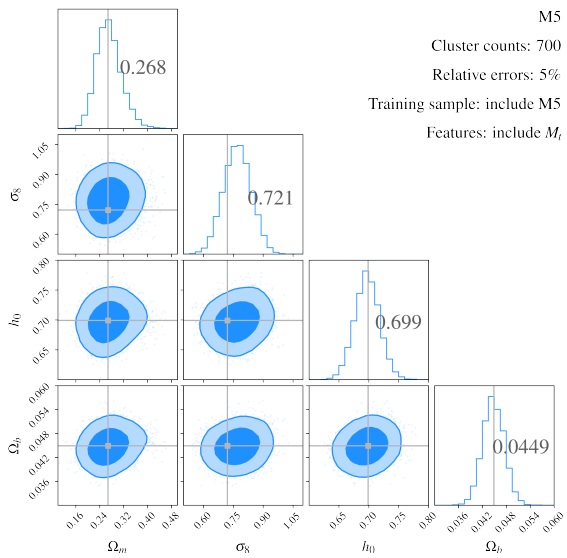

Fig. B.1. Cosmological parameters of M5 inferred by the MLCCA. This graph is the same type as Fig. 8. The true values of all cosmological parameters are within the 1σ confidence interval.

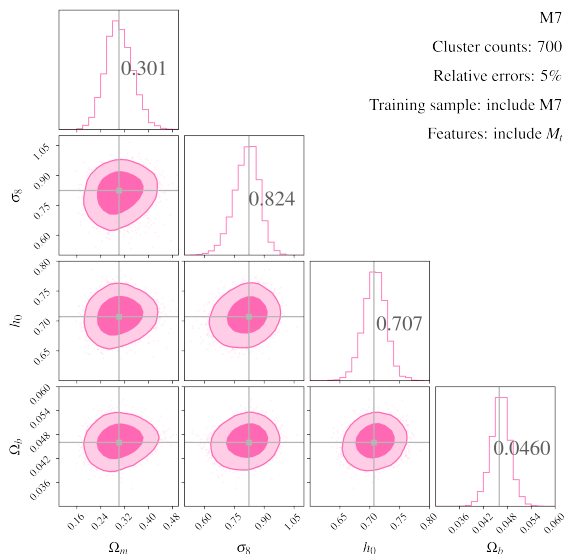

Fig. B.2. Cosmological parameters of M7 inferred by the MLCCA. This graph is the same type as Fig. 8. The true values of all cosmological parameters are within the 1σ confidence interval.

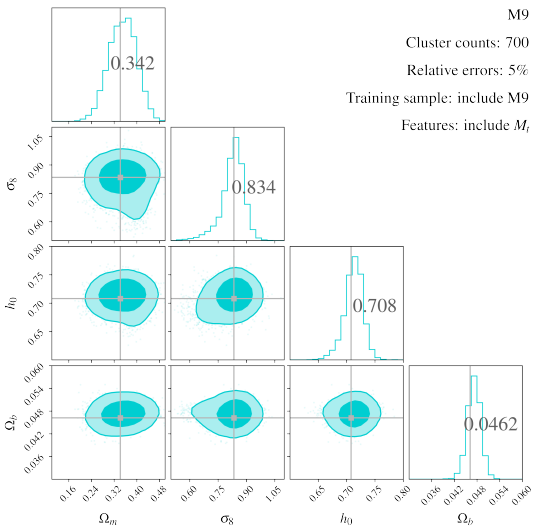

Fig. B.3. Cosmological parameters of M9 inferred by the MLCCA. This graph is the same type as Fig. 8. The true values of all cosmological parameters are within the 1σ confidence interval.

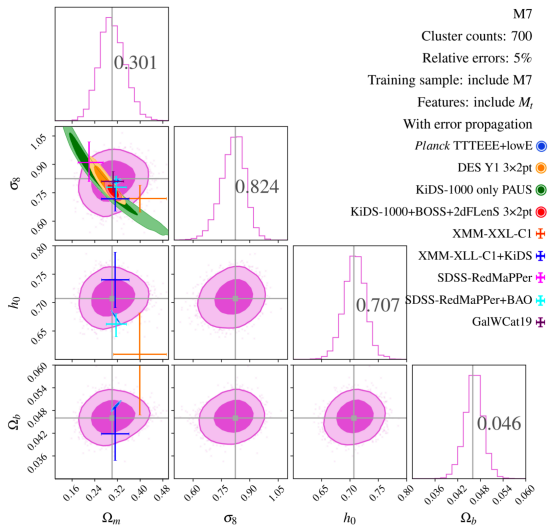

Fig. C.1. Cosmological parameters of M7 inferred by the MLCCA with error propagation. This diagram is overlaid with cosmology constraints as in Fig. 1, with modifications mainly to fit the coordinate range.

Appendix C: Combination with other methods: Effect of training size and error propagation

In this section we visually compare the precision of the MLCCA approach with other methods, to check the ability of the new approach to compete with standard approaches and possibly help solve some degeneracies among the cosmological parameters. This check is not meant to be complete, as we use only the weak lensing and CMB results shown in Fig. 1, but is meant to put the results found in Sect. 4.3 in the context of the cosmological parameter tensions. In Fig. C.1 we overlap the confidence contours from weak lensing and CMB as from Fig. 1, which allows us to compare how the degeneracies among the parameters work differently in the different methods. In particular, the MLCCA contours are more symmetric around the true values and do not show the classical degeneracy between σ_8 and Ω_m parameters found for the weak lensing. The size of the 1σ and 2σ contours are larger than the ones produced by the CMB constraints but compatible with the ones from weak lensing. In principle, we can expect to reduce the size of the MLCCA contours by: 1) increasing the training sample size; 2) assuming a less conservative choice to propagate the errors on the parameters of the individual clusters. For the former, we have tested the impact of the training sample selecting from Fig. 2 with cluster counts larger than 20k, and repeated the training of the MLCCA using 20k clusters and by still testing on the usual 700 over 20 random extractions, with no overlap with the training sample. We have, thus, excluded cosmologies M1 to M5 and chose to predict the cosmological parameter for the catalog from the most central of the residual cosmology, which was M9. We have then compared the contours with the one obtained from the standard training made on the 7000 cluster sample and shown in Fig. B.3 and found that the accuracy of all cosmological parameters is, in fact, slightly improved. We go from $(0.351^{+0.047}_{-0.050}, 0.836^{+0.046}_{-0.061}, 0.714^{+0.014}_{-0.016}, 0.0473^{+0.0016}_{-0.0018})$ for Ω_m, σ_8, h_0, Ω_b for the 7k training case to the $(0.355^{+0.041}_{-0.043}, 0.844^{+0.038}_{-0.046}, 0.715^{+0.013}_{-0.014}, 0.0474^{+0.0014}_{-0.0014})$ for the 20k training sample, i.e., with an average $15 \pm 5\%$ improvement.

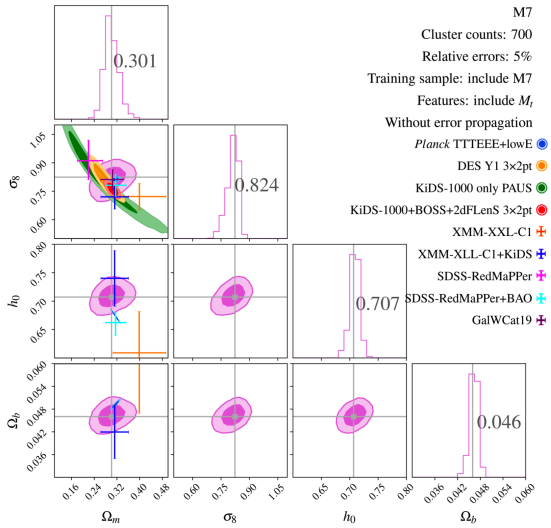

Fig. C.2. Cosmological parameters of M7 inferred by the MLCCA without error propagation. This diagram is overlaid with cosmology constraints as in Fig. 1, with modifications mainly to fit the coordinate range.

This further motivates the use of larger volumes for future applications. To test the impact of the error propagation, we have designed a comparative experiment for M7 cosmology but bypassed the step of error propagation (see Sect. 3.3.3). This is shown in Fig. C.2, where we can clearly see that each contour shrinks after canceling the conservative option we have made for the error propagation.

We can finally conclude that the MLCCA has the advantage of fully accounting for the degeneracies among the cosmological parameters, i.e., providing relatively symmetric confidence contours (especially in the presence of a uniform distribution of prior cosmologies), and regardless of the choice of error propagation, it provides comparable precisions on the parameters that can be eventually improved using considerably larger training samples.