# Learning from Topology: Cosmological Parameter Estimation from the Large-scale Structure

**Anonymous Authors**[1]

## Abstract

The topology of the large-scale structure of the universe contains valuable information on the underlying cosmological parameters. While persistent homology can be applied to extract this topological information, the optimal method for parameter estimation from this tool remains an open question. To address this, we propose a neural network model to map persistence images to cosmological parameters. Through a parameter recovery test, we demonstrate that our model provides accurate and precise estimates, considerably outperforming a Bayesian inference approach.

## 1. Introduction

The recent decade has brought the cosmological community powerful machine learning tools to analyze the vast amount of data from large-scale sky surveys. In this paper, we propose a new method to attack the long-standing problem of cosmological parameter estimation from the large-scale structure of the universe. It has been shown that decent results can be achieved using deep learning (e.g., Ravanbakhsh et al., 2017; Ntampaka et al., 2020; Wen et al., 2023; Hwang et al., 2023). However, despite their effectiveness, neural networks trained directly on low-level data (such as dark matter fields and galaxy maps) lack interpretability and provide little understanding of the underlying physics due to the high degrees of freedom in the input data.

Simply put, cosmologists seek summary statistics that not only stem from physical intuitions but also retain a substantial amount of information. In this regard, we consider applying persistent homology (PH) to point clouds of dark matter halos. PH is a topological data analysis tool that quantifies the robustness of topological features across length scales, allowing for a natural description of the multi-scale patterns in the large-scale structure that the halos trace. Recent studies have demonstrated that this approach can be used to detect primordial non-Gaussianity (Biagetti et al., 2021), identify cosmic structures (Xu et al., 2019), and differentiate dark matter models (Cisewski-Kehe et al., 2022).

The raw outputs of a PH computation are persistence diagrams, which can be conveniently vectorized as persistence images. We may flatten and use them directly for Bayesian inference by assuming a Gaussian likelihood via a covariance structure between pixels (e.g., Cole & Shiu, 2018). However, such a high-dimensional covariance matrix is often poorly behaved and requires a considerable amount of additional data for its estimation. More critically, the covariance does not capture higher-order correlations, such as patterns spanned by multiple neighbouring pixels.

We hereby propose to train a convolutional neural network (CNN) model on persistence images to estimate the underlying cosmological parameters. For comparison, we perform the same task using Bayesian inference with a histogram-based summary statistic recently studied in the literature (Biagetti et al., 2022). We find that our CNN model recovers parameters more accurately and precisely, thus improving the conventional approach. This work also serves as a pioneering example of integrating computational topology and machine learning in the context of cosmology.

## 2. Cosmology and the Large-scale Structure

The ΛCDM model is the best-supported theory of our universe (Dodelson & Schmidt, 2020; Baumann, 2022). In particular, it explains the evolution of inhomogeneities in the matter distribution in an expanding spacetime. This paper focuses on two parameters within this model: the matter density $\Omega_{\rm m}$, and the clustering amplitude $\sigma_8$. While the impacts of these parameters on the evolution are well understood in the linear regime, beyond which we generally resort to N-body simulations. Hence, the inverse problem, i.e., parameter estimation from the late-time matter distribution, holds significance both in theory and in practice.

The Zel'dovich approximation, a non-linear model for the evolution of non-interacting particles, predicts that ellipsoidal distributions of matter collapse along their axes into

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the workshop on Synergy of Scientific and Machine Learning Modeling. Do not distribute.

clusters, filaments, and walls (Zel'dovich, 1970; Hidding et al., 2013). These components assemble hierarchically to form multi-scale halo clusters, filament loops, and cosmic voids, which are the topological features we track with PH.

## 3. Persistence Statistics

### 3.1. Persistent Homology

We apply PH to 3D point clouds of halos. A point cloud is first triangulated as a simplicial complex, i.e., a set of points, line segments, triangles, and tetrahedrons. We can then define a *parametrized* family of nested subcomplices called a filtration. That is, as the filtration parameter $\nu$ increases, simplices are added to the subcomplex according to a specified set of rules. We use the $\alpha$DTM$\ell$-filtration (Biagetti et al., 2021; 2022; Chazal et al., 2017) in this work. The idea is roughly that $\nu$ is a length scale, and a simplex is added if its "size" is smaller than the given $\nu$.

For every value of $\nu$, we identify the topological features in the corresponding subcomplex. They are 0-, 1-, and 2-cycles, which represent islands, closed loops, and enclosed cavities, respectively. Physically, they correspond to halo clusters, filament loops, and cosmic voids in the large-scale structure. Hence, the evolution of the simplical subcomplex throughout the filtration process can be described by a set of $(\nu_{\rm birth}, \nu_{\rm death})$ values at which $n$-cycles are born and killed.

### 3.2. From Persistence Diagrams to Summary Statistics

The list of $(\nu_{\rm birth}, \nu_{\rm death})$ pairs can be recast in the $(\nu_{\rm birth}, \nu_{\rm persist} = \nu_{\rm death} - \nu_{\rm birth})$ coordinates. For each of the three $n$ values, we can plot $(\nu_{\rm birth}, \nu_{\rm persist})$ of all the $n$-cycles. These three plots called the persistence diagrams are the raw outputs of our PH computation. We can further convert them into useful data vectors:

*Persistence Images* - We assign a Gaussian kernel to each point in a persistence diagram. Then we discretize the birth-persistence plane and sum up all the kernel contributions within each pixel to obtain a 2D array of numbers, called a persistence image (Adams et al., 2017).

*Histograms* - We follow Biagetti et al., 2022 and construct histograms from the distributions of $\nu_{\rm birth}$ and $\nu_{\rm persist}$ for all $n$. There are $3 \times 2 = 6$ histograms in total, and they are concatenated into a 1D array of numbers.

We use persistence images for our CNN model (**PI-CNN**) and histograms for Bayesian inference (**Hist-BI**).

## 4. Simulation and Dataset

This section details all the computational steps (Figure 1) we take to create our dataset of summary statistics.

### 4.1. Dark Matter Simulation

We employ FlowPM (Modi et al., 2021) to produce snapshots of late-time distributions of dark matter particles. Although it lacks exact dynamics, this fast solver is sufficiently accurate for this work as a proof of concept. (see Feng et al., 2016 for discussions on accuracy issues).

Given cosmological parameter values, we can compute the primordial matter power spectrum, from which Gaussian random fields at a required redshift can be generated via the transfer function (Eisenstein & Hu, 1998) and linear growth factors. We begin evolving the Gaussian field at $z = 9$ for numerical stability, and a total of 10 time-steps are used. Snapshots are taken at redshift $z = 0.5$ to match standard survey samples (e.g., the BOSS CMASS galaxies, Reid et al., 2016) for the convenience of future analyses on galaxy maps. Each simulation box is $(256\,h^{-1}{\rm Mpc})^3$ in volume and contains $(160)^3$ particles, resulting in a particle resolution on par with state-of-the-art simulations such as the Quijote suite (Villaescusa-Navarro et al., 2020).

For both the training of our CNN model and likelihood evaluation, we vary $\Omega_{\rm m}$ and $\sigma_8$ in our simulations, where $\Omega_{\rm m} \in [0.21, 0.41]$ and $\sigma_8 \in [0.72, 0.92]$. Each range is divided into 60 equal intervals for a total of 3600 distinct $(\Omega_{\rm m}, \sigma_8)$ configurations, and we generate 10 independent realizations for each configuration. Hence, there is a grand total of 36000 simulations in the main dataset. We use the Planck 2015 results for all other cosmological parameters (final column of Table 4 in Planck Collaboration, 2016). Moreover, a flat universe is assumed such that we take the dark energy density as $\Omega_\Lambda = 1 - \Omega_{\rm m}$ whenever applicable.

We further produce 15000 realizations at the fiducial cosmology $(\Omega_{\rm m}, \sigma_8) = (0.3089, 0.8159)$. We use 5000 of these to estimate the covariance matrix in the likelihood and the remaining 10000 for the parameter recovery test.

### 4.2. Halo Catalog

We use the ROCKSTAR halo finder (Behroozi et al., 2013) to identify dark matter halos in each simulation snapshot. We set the force resolution to $0.005\,h^{-1}{\rm Mpc}$ and keep other settings as default. In each fiducial box, ∼2000 halos are identified, which is comparable to Quijote as a credibility check. Hence, each halo catalog is a list of halo positions in real space to which we apply PH.

### 4.3. Summary Statistics

Our PH code relies on the GUDHI libraray (The GUDHI Project, 2015). From persistence diagrams to images, we employ SCIKIT-LEARN to fit the Gaussian kernel density model, with the bandwidth parameter set to 2. Each point in the diagrams is also customarily weighted by $\sqrt{\nu_{\rm persist}}$

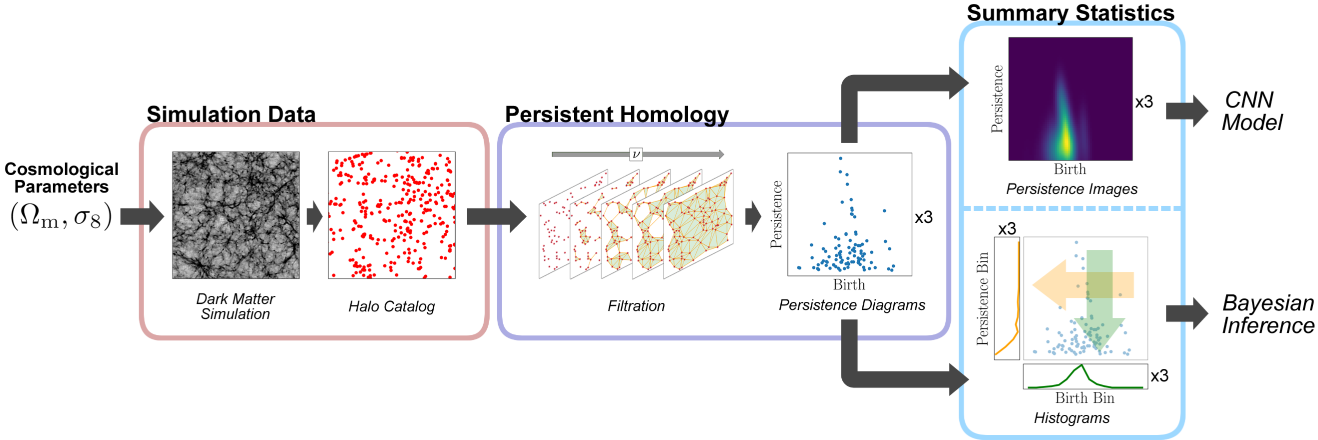

*Figure 1.* Given a $(\Omega_{\rm m}, \sigma_8)$ pair, we generate a realization of the dark matter field at $z = 0.5$ in a $(256\,h^{-1}{\rm Mpc})^3$ box. We locate the halos in the field and apply persistent homology to the halo catalog. The output from the filtration is a list of $(\nu_{\rm birth}, \nu_{\rm persist})$ values, plotted as persistence diagrams. We can 1) pixelate the diagrams into persistence images for our CNN model, or 2) sum up the topological features along each axis into histograms for Bayesian inference. “×3” refers to having three copies for the 0-, 1-, and 2-cycles.

to emphasize more persistent features. The dimensions of each persistence image is $64^2$. For the histograms, we use the same binnings across all parameter configurations. After concatenation, we further downsample the full data vector to $6 \times 16 = 96$ numbers for optimal inference performance.

## 5. Estimation Methods

### 5.1. Convolutional Neural Network Model

We design a neural network model that combines in parallel a CNN with a stack of dense layers to map the persistence images to $(\Omega_{\rm m}, \sigma_8)$. The inputs to the model are the 0-, 1-, and 2-cycle persistence images stacked into 3 channels, and the outputs are 2 numbers for our 2 parameters (Figure 2).

On the CNN side of our parallel networks, we use four sequential blocks, each made of a $3 \times 3$ convolution with no padding, and ReLU activation functions. Each convolution is followed by a $2 \times 2$ max pooling layer with stride 2. After the fourth block, the data is reduced to $2 \times 2$ pixels, and two dense layers with ReLUs follow to output 2 numbers.

On the dense side, we turn the persistence images into a 1D list of data by summing along the birth and persistence axes of the image. As this summed data is only $3 \times 2 \times 64 = 384$ numbers, we use a stack of dense layers on the entire data without pooling. We tested different combinations of 1D convolutions and pooling, with and without ResNet blocks, and found no increase in model performance over simply using a stack of 5 dense layers with ReLUs.

We find a modest increase in model precision ($1\sigma$’s are reduced by $\sim$10%) with the parallel networks over just the CNN side alone. The 2 numbers output from each side of the model are finally averaged into our $(\Omega_{\rm m}, \sigma_8)$ estimate.

We use 33000 persistence images for our training set and 3000 for the validation set. For every $(\Omega_{\rm m}, \sigma_8)$ configuration, we average 10 corresponding images pixelwise. While this averaging does reduce the number of training set images by a factor of 10, we find increased precision of our model even with the extra risk of overfitting. Our loss function is the mean squared error between the model output parameters and true parameter values. By analyzing training and validation performance, we have carefully chosen the number of parameters in our network (detailed in Figure 2) to maximize performance while preventing overfitting. The model has 1.34 million parameters in total. To additionally help prevent overfitting, we use a small batch size of 16. We train with a learning rate of $10^{-4}$ with the Adam optimizer, decreased by a factor of 0.75 when the loss plateaus.

A potential hurdle we suspect in using a traditional CNN on persistence images is the overall smoothness of each image. In a traditional image classification dataset, we encounter, e.g., many distinct edges in each image; the convolutional filters in the first few layers of the network would learn to detect these edges. It may require a more novel approach to capture more of the features of persistence images.

### 5.2. Bayesian Inference

We adopt a flat prior $p(\boldsymbol{\theta})$ and a Gaussian likelihood $\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{D})$ such that we have for the log-posterior:

$$\ln p(\boldsymbol{\theta}|\boldsymbol{D}) = -\frac{1}{2}(\boldsymbol{D} - \boldsymbol{\mu}(\boldsymbol{\theta}))^T C^{-1}(\boldsymbol{D} - \boldsymbol{\mu}(\boldsymbol{\theta})) + {\rm const.}$$

by Bayes’ theorem. Here $\boldsymbol{\theta}$ is one of the 3600 $(\Omega_{\rm m}, \sigma_8)$ configurations at which the log-posterior is to be numerically evaluated. $\boldsymbol{\mu}(\boldsymbol{\theta})$ is the full histogram data vector measured at $\boldsymbol{\theta}$ averaged over 10 realizations. $\boldsymbol{D}$ is an observation, also

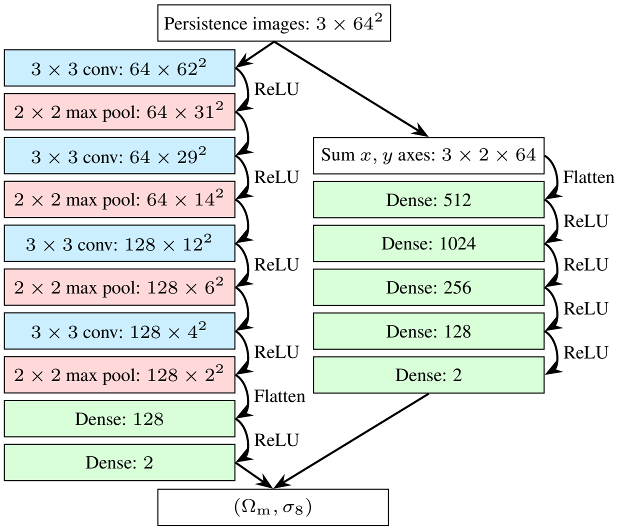
Persistence images: $3 \times 64^2$
$3 \times 3$ conv: $64 \times 62^2$
ReLU
$2 \times 2$ max pool: $64 \times 31^2$
$3 \times 3$ conv: $64 \times 29^2$
ReLU
$2 \times 2$ max pool: $64 \times 14^2$
$3 \times 3$ conv: $128 \times 12^2$
ReLU
$2 \times 2$ max pool: $128 \times 6^2$
$3 \times 3$ conv: $128 \times 4^2$
ReLU
$2 \times 2$ max pool: $128 \times 2^2$
Flatten
Dense: 128
ReLU
Dense: 2
Sum $x, y$ axes: $3 \times 2 \times 64$
Flatten
Dense: 512
ReLU
Dense: 1024
ReLU
Dense: 256
ReLU
Dense: 128
ReLU
Dense: 2
$(\Omega_m, \sigma_8)$

*Figure 2.* Architecture of our CNN model, with layer names and layer output dimensions.

averaged over 10 realizations, measured at some unknown configuration that we want to recover. $C$ is the covariance matrix, and we include the Hartlap factor (Eq. (17) in Hartlap et al., 2006) for the unbiased estimation of $C^{-1}$. The constant term $\mathrm{const.} = \ln \frac{p(\boldsymbol{\theta})}{p(\boldsymbol{D})}$ can be ignored as it plays no role in finding the maximum a posteriori (MAP) estimate.

We use a 2D Gaussian filter to smooth out the noise in the numerically evaluated log-posterior, with $\sigma$ of the kernel set to 2 pixels. The MAP estimate is taken to be the $(\Omega_m, \sigma_8)$ configuration that maximizes the (log-)posterior.

## 6. Parameter Recovery Test

We conduct a parameter recovery test to compare the PI-CNN and Hist-BI pipelines. Persistence statistics are averaged over every 10 boxes in the test dataset of 10000 fiducial boxes, i.e., we have 1000 independent observations at $(\Omega_m, \sigma_8) = (0.3089, 8159)$. This averaging is in accordance with how the estimation methods are trained, as described in the previous section.

We present our result in Figure 3. In the central panel, we plot the PI-CNN estimates in red and Hist-BI in blue. The MAP estimates are restricted to grid points because the log-posterior is evaluated on the discretized $\Omega_m$-$\sigma_8$ plane. We fit a bivariate Gaussian to each of the distributions and plot the contours for the 68% and 95% confidence interval. The marginalized Gaussians are plotted in the side panels, from which we obtain the mean estimates to be $(0.3088 \pm 0.0105, 0.8160 \pm 0.0154)$ for PI-CNN and $(0.2817 \pm 0.0267, 0.8345 \pm 0.0177)$ for Hist-BI. The true fiducial values are marked by the dashed lines.

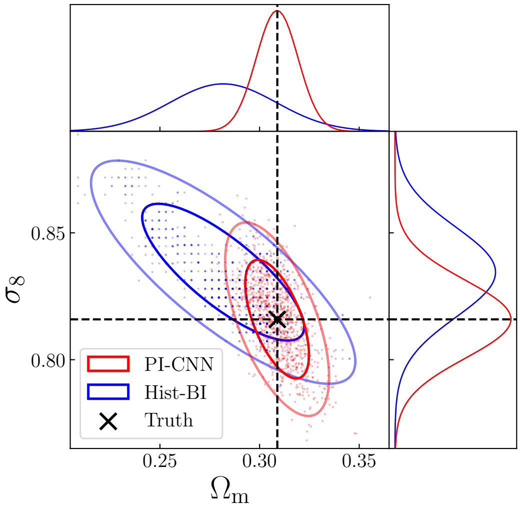

*Figure 3.* Red and blue dots in the central panel mark the 1000 $(\Omega_m, \sigma_8)$ values estimated on the fiducial measurements by our CNN model and Bayesian inference respectively. We fit a Gaussian to each distribution, and the contours mark the 68% and 95% confidence intervals. Side panels show the marginalized Gaussians. True values are located by the dashed lines.

The PI-CNN Gaussian is accurately centered on the true value while that for Hist-BI is heavily biased. The variance in the PI-CNN estimates is also lower, particularly for $\Omega_m$. Both Gaussians show that there is a negative correlation between the two parameters, which is a common result for many summary statistics (e.g., mass function of low-redshift galaxy clusters, Vikhlinin et al., 2009). Intriguingly, the covariance from PI-CNN $(-9.73 \times 10^{-5})$ is less negative than that from Hist-BI $(-3.80 \times 10^{-4})$. This implies that the 2D persistence images contain extra information that can be extracted by our CNN model for breaking partially the parameter degeneracy.

## 7. Conclusion and Outlook

We have put cosmology, computational topology, and machine learning together by training a CNN model on persistence images for cosmological parameter estimation. From the parameter recovery test, we have found that our CNN model gives accurate and precise estimates, outperforming a Bayesian inference method that uses a histogram-based persistence statistic.

There are two possible future directions. First, we should use galaxy maps and take into account observational effects, such as sky cuts and instrumental systematics, for our model to be readily applicable to survey data. Second, we may investigate the effectiveness of our CNN model by, e.g., studying the saliency maps (Simonyan et al., 2013), which may help understand the information content of persistence images relevant to cosmology.

## Broader Impact

In this work we use machine learning to learn cosmological information from the topology of simulated dark matter halo catalogs. This research benefits the theoretical cosmology community by demonstrating that a machine learning method of analyzing topological information can outperform the traditional methods. Considering the result of this work, some previous work on persistent homology in cosmology that did not use a machine learning approach might be revisited, and we encourage future work to consider using machine learning. The data used in this work are taken from cosmological simulations, and we use well-known machine learning and statistical techniques, so we see no ethical or societal consequences introduced by this work.

## References

Adams, H., Emerson, T., Kirby, M., and et al. Persistence images: A stable vector representation of persistent homology. *The Journal of Machine Learning Research*, 18, 2017.

Baumann, D. (ed.). *Cosmology*. Cambridge University Press, 2022.

Behroozi, P. S., Wechsler, R. H., and Wu, H.-Y. The rockstar phase-space temporal halo finder and the velocity offsets of cluster cores. *The Astrophysical Journal*, 762(109), 2013.

Biagetti, M., Cole, A., and Shiu, G. The persistence of large scale structures i: Primordial non-gaussianity. *Journal of Cosmology and Astroparticle Physics*, 04(061), 2021.

Biagetti, M., Calles, J., Castiblanco, L., and et al. Fisher forecasts for primordial non-gaussianity from persistent homology. *Journal of Cosmology and Astroparticle Physics*, 10(002), 2022.

Chazal, F., Fasy, B., Lecci, F., and et al. Robust topological inference: distance to a measure and kernel distance. *The Journal of Machine Learning Research*, 18, 2017.

Cisewski-Kehe, J., Fasy, B. T., Hellwing, W., and et al. Differentiating small-scale subhalo distributions in cdm and wdm models using persistent homology. *Physical Review D*, 106, 2022.

Cole, A. and Shiu, G. Persistent homology and non-gaussianity. *Journal of Cosmology and Astroparticle Physics*, 03(025), 2018.

Dodelson, S. and Schmidt, F. (eds.). *Modern Cosmology*. Academic Press, 2020.

Eisenstein, D. J. and Hu, W. Baryonic features in the matter transfer function. *The Astrophysical Journal*, 496:605–614, 1998.

Feng, Y., Chu, M.-Y., Seljak, U., and McDonald, P. Fastpm: a new scheme for fast simulations of dark matter and haloes. *Monthly Notices of the Royal Astronomical Society*, 463:2273–2286, 2016.

Hartlap, J., Simon, P., and Schneider, P. Why your model parameter confidences might be too optimistic. unbiased estimation of the inverse covariance matrix. *Astronomy and Astrophysics*, 464(1), 2006.

Hidding, J., Shandarin, S. F., and van de Weygaert, R. The zel'dovich approximation: key to understanding cosmic web complexity. *Monthly Notices of the Royal Astronomical Society*, 437, 2013.

Hwang, S. Y., Sabiu, C. G., Park, I., and Hong, S. E. The universe is worth $64^3$ pixels: Convolution neural network and vision transformers for cosmology. 2023.

Modi, C., Lanusse, F., and Seljak, U. Flowpm: Distributed tensorflow implementation of the fastpm cosmological n-body solver. *Astronomy and Computing*, 37(100505), 2021.

Ntampaka, M., Eisenstein, D. J., Yuan, S., and Garrison, L. H. A hybrid deep learning approach to cosmological constraints from galaxy redshift surveys. *The American Astronomical Society*, 889(151), 2020.

Planck Collaboration. Planck 2015 results. xiii. cosmological parameters. *Astronomy & Astrophysics*, 594(A13), 2016.

Ravanbakhsh, S., Oliva, J., Fromenteau, S., and et al. Estimating cosmological parameters from the dark matter distribution. 2017.

Reid, B., Ho, S., Padmanabhan, N., and et al. Sdss-iii baryon oscillation spectroscopic survey data release 12: galaxy target selection and large-scale structure catalogues. *Monthly Notices of the Royal Astronomical Society*, 455:1553–1573, 2016.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. 2013.

The GUDHI Project. *GUDHI User and Reference Manual*. GUDHI Editorial Board, 2015.

Vikhlinin, A., Kravtsov, A. V., Burenin, R. A., and et al. Chandra cluster cosmology project iii: Cosmological parameter constraints. *The Astrophysical Journal*, 692, 2009.

Villaescusa-Navarro, F., Hahn, C., Massara, E., and et al. The quijote simulations. *The Astrophysical Journal Supplement*, 250(2), 2020.

Wen, Y., Yu, W., and Li, D. Cosnas: Enhancing estimation on cosmological parameters via neural architecture search. *New Astronomy*, 99(101955), 2023.

Xu, X., Cisewski-Kehe, J., Green, S., and Nagai, D. Finding cosmic voids and filament loops using topological data analysis. *Astronomy and Computing*, 27, 2019.

Zel'dovich, Y. B. Gravitational instability: An approximate theory for large density perturbations. *Astronomy and Astrophysics*, 5, 1970.