
Lognormal-Field Based Inference on Weak Lensing Convergence Maps with Graph Convolutional Neural Networks

Rintaro Kanaki



München 2023

Lognormal-Feld-Basierte Inferenz auf Schwache-Lensing-Konvergenzkarten mit Graph Convolutional Neural Networks

Rintaro Kanaki

Masterarbeit
an der Universitätssternwarte München
Fakultät für Physik

der Ludwig-Maximilians-Universität München

vorgelegt von
Rintaro Kanaki
aus Tokio, Japan

München, den 15. September 2023

Erstgutachter: Prof. Dr. Daniel Grün

Zweitgutachter: Dr. Oliver Friedrich

Contents

Zusammenfassung	xiii
1 Introduction	1
2 Statistics of Cosmic Density Fields	3
2.1 The Interpretation of Probability	3
2.1.1 Frequentist Interpretation	4
2.1.2 Bayesian Interpretation	4
2.1.3 Bayesian Parameter Inference	4
2.2 Random Field	6
2.2.1 Probability Density Function	6
2.2.2 Gaussian Fields	8
2.2.3 Lognormal Fields	10
2.3 Statistics of Cosmic Density Field	13
2.3.1 Statistically Homogenous and Isotropic Field	14
2.3.2 Random Field in Fourier Space	15
2.3.3 Density Field on the Sphere	17
2.3.4 Limber Integration	18
3 Theory of Cosmology	21
3.1 General Relativity	21
3.1.1 Geometry of Curved Space	22
3.1.2 The Einstein Field Equations	23
3.1.3 Friedmann–Lemaître–Robertson–Walker Metric	23
3.1.4 Friedmann Equations	24
3.2 Homogeneous Universe	26
3.2.1 The Expansion History of the Universe	26
3.2.2 The Energy Components of the Λ CDM Universe	28
3.2.3 Observational Quantities in Cosmology	30
3.3 The Origin of Density Fluctuation	32
3.4 Linear Structure Formation	33
3.4.1 Linear Perturbation Theory	34
3.4.2 Matter Power Spectrum	34

3.4.3	The Power Spectrum and σ_8	35
3.5	Non-Linear Structure Formation	36
3.5.1	N-body Simulation	37
3.5.2	The Statistics beyond 2-point Functions	38
3.6	Summary of Λ CDM Cosmology	39
3.6.1	The Λ CDM Parameters	39
3.6.2	Challenges in Λ CDM Theory	40
4	Weak Gravitational Lensing	43
4.1	Gravitational Lensing	43
4.1.1	The Lens Equation	45
4.1.2	Shear and Convergence	45
4.2	Weak Lensing	47
4.2.1	Weak Lensing Basics	48
4.2.2	Cosmic Shear and Convergence Field	49
4.2.3	Weak Lensing Convergence Power Spectrum	50
5	Simulation Pipeline	53
5.1	Simulation Setup	54
5.2	Structure of Simulation Pipeline	55
5.2.1	Matter Power Spectrum Generation	56
5.2.2	Projection Integral	57
5.2.3	Lognormal Shift Parameter	58
5.2.4	Generation of Convergence Maps	59
5.2.5	Execution Time of the Simulation Pipeline	60
5.3	Validation of Simulation Pipeline	61
5.3.1	T17 N-body Simulation	61
5.3.2	Validation with T17 Simulations	62
6	Theory of Neural Networks	67
6.1	Terminology	68
6.1.1	Artificial Intelligence	68
6.1.2	Machine Learning	69
6.1.3	Deep Neural Network	70
6.2	Structure of Deep Neural Networks	71
6.2.1	Activation Functions	72
6.2.2	Loss Function	74
6.2.3	Optimizer	76
6.3	Convolutional Neural Networks	76
6.3.1	Structure and Network Layers of CNN	77
6.3.2	Regularization of CNNs	79
6.4	Graph Convolutional Neural Networks	80
6.4.1	Graph Convolutional Neural Networks	81

6.4.2	ChebNet	82
6.4.3	DeepSphere	83
7	Inference Pipeline	87
7.1	Data Preprocessing	87
7.1.1	Downsampling	88
7.1.2	Effect of Downsampling	89
7.2	Structure of Inference Pipeline	90
7.2.1	Model Architecture	91
7.2.2	Negative-Log Gaussian Likelihood Loss	93
7.3	Training of GCNN Models	95
7.3.1	Training Strategy	95
7.3.2	Interpreting Loss and Metrics	97
7.4	Validation Test of Inference Pipeline	100
8	Results and Discussions	103
8.1	Clouds of Points	104
8.1.1	Cluster of Points	104
8.1.2	Bias and Variance	104
8.1.3	Sample Covariance	107
8.2	Posterior Distributions from the Estimation using GCNN	110
8.2.1	Parameter Covariances	111
8.2.2	Contours from Single Cosmology	113
8.2.3	Posterior Distributions	113
8.2.4	Figure of Merits	117
8.3	Comparison with Fisher Forecast	118
8.3.1	Derivation of Fisher Matrix	120
8.3.2	Data Covariance Matrixes	120
8.3.3	Comparison with GCNN Model	123
9	Conclusion	127
10	Futural Research	129
10.1	Towards Trustworthy Estimations in Field-Based Cosmology with Machine Learning	129
10.2	Addressing Real-World Complexities in Field-Based Cosmological Inference	130
10.3	Cosmology beyond Λ CDM	132
A	Graph Spectral Theory	135
A.1	Graph Theory	135
A.2	Graph Spectral Theory	136

B Robustness of Inference Pipeline	139
B.1 Robustness of GCNN Model	139
B.2 Robustness of Training of GCNN Model	141
C The contours of S_8 vs. Ω_m	151
C.1 Conversion from σ_8 to S_8	151
C.2 Clouds of Points in $S_8 - \Omega_m$ Plane	152
C.3 Posterior Distributions in $S_8 - \Omega_m$ Plane	154
Danksagung	167
Selbstständigkeitserklärung	169

List of Figures

2.1	Cumulants and Moments	8
2.2	Gaussian Distribution	9
2.3	The Figure of the Projected Density Field	11
2.4	Lognormal PDFs	11
3.1	Geometry of Universe	25
3.2	Hubble Parameter in Flat Universes	30
3.3	Comparison of Cosmological Distances	32
3.4	CMB Anisotropy Maps	33
3.5	Linear Power Spectrum	35
3.6	The Convergence Field from Takahashi N-body Simulation	37
3.7	Non-Linear Power Spectrum	38
3.8	Matter Power Spectra in Various Cosmologies	41
4.1	The Cheshire Cat Image of Galaxy Cluster	44
4.2	The Lens System	46
4.3	Shear and Convergence	47
4.4	Convergence Power Spectra in Various Cosmologies	51
5.1	LHS Sampling of Training and Validation Datasets	55
5.2	The Structure of the Simulation Pipeline	56
5.3	The Source Redshift Distributions	58
5.4	Flask Lognormal and Gaussian Histograms	60
5.5	Effect of the Correction Formulae	64
5.6	Comparison Based on Angular Power Spectrum of our Simulation Pipeline to T17 Simulation	65
5.7	Comparison of Measured Higher Moments between the Flask Lognormal Simulation and T17 Simulation	66
6.1	An example of the output of generative model	67
6.2	The Venn diagram of the ideas of "AI", "Machine Learning" and "Deep Learning"	69
6.3	The Pictorial Explanation of Overfitting and Underfitting	71
6.4	Structure of Deep Neural Network	73

6.5	The Comparison of Activation Functions	74
6.6	The Outputs of Each Layer of CNNs	77
6.7	Max Pooling	79
6.8	Comparison of Image and Graph	81
6.9	Localization of Filter in Spectral Domain	83
6.10	Wavelet Kernel vs. Chebyshev Polynomial	84
6.11	Pooling Layer of DeepSphere	85
7.1	Inference Pipeline	88
7.2	KL Divergence of Density Field from N-body Simulation with Respect to Gaussian	90
7.3	KL Divergence of Various Random Fields with Respect to Gaussian	91
7.4	Model Architecture of GCNN Model	92
7.5	Loss versus Epoch	98
7.6	χ^2 versus Epoch	98
7.7	$\log(\det(\Sigma))$ versus Epoch	99
7.8	MSE versus Epoch	99
7.9	Validation Test in 2D	101
7.10	Validation Test in Ω_m	102
7.11	Validation Test in σ_8	102
8.1	Point Clouds from Gaussian Model	105
8.2	Point Clouds from Lognormal Model	106
8.3	Predicted Mean vs. Fiducial Cosmology	107
8.4	Bias and Variance	108
8.5	Euclidean Distance between the Predicted Mean and the Fiducial Value . .	109
8.6	Derived Parameter Covariances using Sample Covariance of Point Cloud .	110
8.7	Posterior Distributions with Gaussian Model and sample covariance-Derived Covariance	111
8.8	Posterior Distributions with Lognormal Model and sample covariance-Derived Covariance	112
8.9	Mean of Parameter Covariance Matrixes	113
8.10	Posterior Distributions from Single Realization	114
8.11	Comparison of two Averaging Methods	115
8.12	Comparison of Various Test Datasets using Gaussian Model	116
8.13	Comparison of Various Test Datasets using Lognormal Model	117
8.14	Figure of Merits of Sample Covariance	118
8.15	Figure of Merits of Averaged Parameter Covariance	119
8.16	Data Correlation Matrix	122
8.17	Fisher Contour vs. Contour from GCNN	124
10.1	Examples of Learned Filters	131
A.1	Example of Undirected Weighted Graph	137

B.1	Predicted Mean vs. Fiducial Cosmology by Varying the Test Data	140
B.2	Fractional Difference of Positions and Variance	141
B.3	Comparison of Posterior Distribution between Test Datasets 1 and Test Datasets 2	142
B.4	Loss versus Epoch: Comparison with Model 2	143
B.5	Validation Test in 2D: Comparison with Model 2	144
B.6	Point Clouds from Gaussian Model	145
B.7	Point Clouds from Lognormal Model	146
B.8	Predicted Mean vs. Fiducial Cosmology by Varying the Training Data . .	147
B.9	Euclidean Distance between the Predicted Mean and the Fiducial Value: Model 2	147
B.10	Posterior Distributions with Gaussian Model and Averaged Covariance: Model 2	148
B.11	Posterior Distributions with Lognormal Model and Averaged Covariance: Model 2	149
C.1	Point of Clouds of S_8 - Ω_m Plane	152
C.2	Predicted Mean vs. Fiducial Cosmology using S_8	153
C.3	Euclidean Distance between the Predicted Mean and the Fiducial Value using S_8	154
C.4	Posterior Distribution of S_8 and Ω_m Using Sample Covariances	155
C.5	Posterior Distribution of S_8 and Ω_m using Averaged Covariances	155
C.6	Figure of Merits of Sample Covariance Using S_8	156
C.7	Figure of Merits of Averaged Parameter Covariance Using S_8	156

Zusammenfassung

Die Genesis des Universums beginnt mit winzigen, hoch-gaußschen Fluktuationen, die im Laufe der nichtlinearen Strukturbildung zu kosmischen Strukturen wie Galaxien oder Galaxienhaufen heranwachsen. In der beobachtenden Kosmologie ist die 2-Punkt-Statistik die Standardmethode zur Analyse, da sie die Varianz der Verteilung der Large Scale Structures misst. Allerdings erfasst diese Methode nicht die gesamte Verteilung des kosmischen Dichtefeldes, was zu zusätzlichen Bias und Ungenauigkeiten in den posteriori-Wahrscheinlichkeiten der kosmologischen Parameter führen kann. Um dieses Ziel zu erreichen, haben wir zunächst projekzierte kosmische Dichtefelder mithilfe der logarithmischen Normalverteilung simuliert und dabei vorgegebene kosmologische Parametern verwendet, um Trainingsdaten zu generieren. Das Modell der logarithmischen Normalverteilung erweist sich als rechnerisch effizient und leistungsstark zur Beschreibung kosmischer Dichtefelder. Dennoch bleibt die genaue Genauigkeit dieser Annäherung bei der Schätzung der posteriori-Wahrscheinlichkeiten durch feldbasierte Inferenz eine offene Frage. Aus diesem Grund haben wir zusätzlich auch Gaussische Felder erzeugt, um Vergleiche und Robustheitsprüfungen unseres künstlichen Intelligenz-Modells mit der 2-Punkt-Statistik durchzuführen. Unsere künstliche Intelligenz-Architektur verwendet ein Graph Convolutional Neuronales Netzwerk, das im Gegensatz zu herkömmlichen Convolutional Neural Networks die Rotationsinvarianz berücksichtigt. Wir haben die negative Log-Gauß-Likelihood als Verlustfunktion verwendet, die nicht nur die Schätzungen der kosmologischen Parametern ermöglicht, sondern auch die Schätzung der Kovarianzmatrix der Parameter. In unserer Untersuchung haben wir zwei GCNN-Modelle trainiert: eines mit einer Gaußschen Verteilung und das andere mit einer logarithmischen Normalverteilung. Wir haben die trainierten Netzwerke mit drei Arten von Testdatensätzen geprüft: solche, die kosmische projektierte Dichtefelder unter Verwendung einer logarithmischen Normalverteilung, einer Gaußschen Verteilung und N-Körper-Simulationen. Wir vergleichen die Ergebnisse der Schätzung der posterioren Verteilungen aus unserer künstlichen Intelligenz mit den Schätzungen aus der Fisher-Analyse des Power-Spektrums der Materie, um die Robustheit und Genauigkeit unserer Methode zu validieren.

Chapter 1

Introduction

Cosmology provides empirical evidence for fundamental physics as Belarussian physicist Zeldovich said "Our Universe is the accelerator for poor people". These "poor man's accelerators" have been a delightful success in the last half-century. For example, the CMB survey through the Planck satellite shows excellent agreement with Λ CDM theory [97], and the baryonic oscillation spectroscopic survey (BOSS) provides detailed 3D dark matter distribution [25]. Nonetheless, our understanding of fundamental components of standard cosmology, such as dark energy and dark matter, remains incomplete. Moreover, the conventional cosmological model, Λ CDM, grapples with inconsistencies in estimated parameters like H_0 or S_8 when different tracers are employed. These inconsistencies may arise from the limitations of traditional statistical approaches.

Now, how do we enhance our comprehension of the universe through observational data? In modern cosmology, Bayesian statistics plays a pivotal role. Bayesian parameter inference involves updating our prior constraints using likelihood, which measures the correspondence between observed and theoretical data. This approach hinges on the type of data we employ. One way to foster a Bayesian understanding of the universe is by utilizing the matter density field to precisely estimate cosmological parameters. However, modeling the matter density field in the late-time universe presents challenges due to non-linear gravitational structure formation, which introduces non-Gaussian features into the cosmic matter density field. For example, non-Gaussianity manifests in the weak lensing convergence field—a line-of-sight projection of the 3D matter distribution. However, conventional cosmological analyses predominantly rely on two-point summary statistics that overlook this non-Gaussian information. The lognormal approximation of the cosmic density field offers a more accurate estimation of cosmological parameters compared to the Gaussian approximation, as the shifted lognormal field can capture non-Gaussianity owing to the presence of higher order moments [60]. Although not as precise as N-body simulations, the lognormal model is computationally efficient, making it a preferred choice for our simulation pipeline.

However, the question arises: How can we infer cosmological parameters without relying on statistical summaries and instead directly derive them from simulated random fields? In our inference pipeline, we leverage machine learning techniques. By using simu-

lated projected density fields generated with the lognormal model and pairing them with corresponding cosmological parameters as labels, we train a graph convolutional neural network using layers from `DeepSphere` [27], [96]. Those graph convolutional layers are designed for spherical data representation, such as `HEALPix` maps, and adhere to rotational equivariance—unlike traditional convolutional neural networks. Our GCNN minimizes the negative log Gaussian likelihood as a loss function and yields five outputs. The first two outputs provide direct estimates of cosmological parameters, specifically Ω_m and σ_8 , while the remaining three outputs form the building blocks of the parameter covariance matrix. After hyperparameter tuning and training, we employ this architecture to obtain posterior distributions.

To assess the impact of different density field approximations, we employ three distinct test datasets: the lognormal field, the Gaussian field from `Flask`, and the convergence map from the N-body simulation by Takahashi et al. (2017) [114]. We compare the posterior distribution derived from each combination of training and test datasets with the posterior distribution generated by Fisher forecasting using power spectra. This comparison helps us to evaluate the validity of approximating the density field as Gaussian or lognormal and assess the improvement over conventional two-point statistics.

In this Master’s thesis, Chapters 2, 3, 4, and 6 provide foundational theoretical insights. Chapter 2 introduces the fundamentals of statistics pertaining to cosmic density fields and the shifted lognormal field used for modeling the lensing convergence field. Chapter 3 delves into the foundation of cosmology and theoretical description of density profile of our universe within the Λ CDM cosmology, the prevailing cosmological theory that incorporates cold dark matter and a cosmological constant as late-time dark energy. Chapter 4 elucidates the basics of weak gravitational lensing, as the ultimate objective of this project is to apply real projected density fields obtained from weak lensing surveys.

The final theoretical section, Chapter 6, offers a concise introduction to deep neural networks and the theory of graph convolutional neural networks, which form the foundation of our inference pipeline.

Chapters 5 and 7 address the methodology and validation of our simulation and inference pipeline. Chapter 5 provides a detailed explanation of how we acquire the training datasets using `Flask` [124] simulations across the parameter space of cosmological parameters. In contrast, Chapter 7 outlines our inference pipeline for GCNN modeling and conventional two-point analysis. Both methods are employed to obtain posterior distributions of cosmological parameters.

Lastly, Chapter 8 showcases the results and compares the posterior distributions generated by the GCNN model using different test and training datasets. This comparison aims to assess the impact of different approximations. We initially analyze the outputs of the neural networks using Frequentist statistical methods and subsequently derive posterior distributions from estimated parameters and covariances. These posterior distributions are compared in various scenarios with two-point statistics to gauge the robustness of our inference methodology. The culmination of these findings is presented in Chapter 9. Chapter 10 offers a glimpse into the future directions of our analysis.

Chapter 2

Statistics of Cosmic Density Fields

Modern cosmology places substantial reliance on statistical methods for the purpose of quantifying and qualifying our understanding of our universe through extensive datasets. The central focus of this thesis revolves around cosmological parameter inference, a domain encompassing two distinct statistical models. The first model entails the theoretical representation of random fields emanating from a specified cosmological framework. The second model pertains to the inference of cosmological parameters through the comparison of observational data with the stochastic models originating from the aforementioned theoretical construct. Thus, it becomes imperative to provide an introduction to concepts of statistics as well as the stochastic models governing cosmic density fields, setting the stage for subsequent discussions on cosmological theory.

The foundation of this chapter draws significantly from several scholarly works, including [60], [124], [119], [6], [39], [7] and [40].

2.1 The Interpretation of Probability

To establish the groundwork, the initial step involves defining the notion of probability and delving into its interpretations, a fundamental necessity given the model's reliance on this concept. It is important to note that the concept of probability is multifaceted and not constrained to a singular understanding. In the realm of physics, two primary interpretations of probability emerge prominently: the empirical frequentist interpretation and the subjective Bayesian interpretation¹. Each interpretation offers distinct perspectives on the nature of probability, bearing relevance to the construction of the model at hand.

¹The foundational axioms of probability are attributed to Kolmogorov as well as Cox [21]. These axioms, though originating from distinct sources, converge into a unified framework of probability. Notably, while the axiomatic structure remains consistent, variations in the interpretation of probabilistic outcomes are what set these approaches apart.

2.1.1 Frequentist Interpretation

The Frequentist interpretation of probability revolves around perceiving probability as the limit of relative frequencies of events occurring across an infinitely extensive series of statistically independent trials. Within this framework, the probability $P(n|N)$ of a particular event transpiring in relation to the number of trials can be expressed as follows:

$$P(n|N) = \lim_{n \rightarrow \infty} \frac{n}{N}, \quad (2.1)$$

where n represents the count of specific events transpiring, while N symbolizes the total count of trials that are statistically independent. This particular interpretation finds application in domains such as experimental collider physics [79]. In practice, experiments accumulate data through repetitive trials, enabling the derivation of measurements. Subsequently, Frequentist confidence intervals can be formulated, and hypotheses can be tested based on these measurements.

2.1.2 Bayesian Interpretation

The Bayesian interpretation of probability entails perceiving probability as a subjective degree of belief in a given hypothesis. Central to this interpretation is the posterior probability $P(H|D)$, which characterizes the constraint on a hypothesis H given a set of data D . This is governed by Bayes' theorem:

$$P(H|D) = \frac{P(H)P(D|H)}{P(D)}. \quad (2.2)$$

Here, $P(H)$, referred to as the prior, encapsulates existing constraints from prior knowledge, $P(D|H)$, known as the likelihood, outlines the forward probability of observing data D under the assumption of hypothesis H , and $P(D)$ represents the Bayesian evidence, acting as a normalizing factor.

A fundamental difference between the Bayesian and Frequentist interpretations lies in the capability of the former to incorporate prior information into analyses, aiding in the constraining of model parameters. In the domain of cosmology, the Bayesian interpretation holds greater preference [119]. This stems from the realization that our observed universe constitutes merely one realization from a broader ensemble. Consequently, cosmological experiments are markedly distinct from those in collider physics, as they lack the ease or the possibility of repeatability.

2.1.3 Bayesian Parameter Inference

How can we update our understanding of knowledge of cosmology in a Bayesian manner²? First, let's delve into the concept of Bayesian parameter inference. The fundamental objec-

²The author of this thesis had a serious problem with accepting Bayesian statistics for scientific application. Since it is hard to accept "the cosmology is just like poker but using different utility functions". However, my perspective has evolved, as I now perceive Bayesian statistics as a mathematical framework that encapsulates the process of understanding.

tive of Bayesian parameter inference is to establish credible intervals³ for model parameters denoted as $\boldsymbol{\theta}$ based on observed data \mathbf{D} obtained through measurement procedures.

Upon fixing the parameters⁴, it becomes possible to consolidate our understanding by integrating information such as past measurements or fundamental physical principles into what is known as the prior distribution, denoted as $P(\boldsymbol{\theta})$. Subsequently, the computation of the likelihood $P(\mathbf{D}|\boldsymbol{\theta})$ involves comparing the observed measurement values with the theoretical predictions generated by our model.

This entire process of Bayesian parameter inference operates within the framework of Bayes' theorem, which facilitates the refinement of our understanding. The prior distribution $P(\boldsymbol{\theta})$ is multiplied by the likelihood $P(\mathbf{D}|\boldsymbol{\theta})$ and normalized, resulting in the posterior distribution $P(\boldsymbol{\theta}|\mathbf{D})$ as depicted by the equation:

$$P(\boldsymbol{\theta}|\mathbf{D}) = \frac{P(\mathbf{D}|\boldsymbol{\theta})P(\boldsymbol{\theta})}{P(\mathbf{D})}. \quad (2.3)$$

Up to this point, the discussion hasn't touched upon the Bayesian evidence $P(\mathbf{D})$, which functions as a normalization factor. It becomes particularly intricate when dealing with high-dimensional parameters due to the requirement of a high-dimensional integral. This task often proves computationally intensive. To address this, the Monte Carlo Markov Chain (MCMC) technique comes into play. It is employed to calculate the posterior $P(\boldsymbol{\theta}|\mathbf{D})$, making the computation practically feasible.

The primary objective of this thesis revolves around a comparative analysis of two distinct subsets of Bayesian parameter inference methods within the context of the standard cosmological theory, utilizing the cosmic density field. The initial method involves the computation of the likelihood function through the utilization of summary statistics. Given that computing the likelihood based on the entire field is numerically challenging, the density field is condensed into specific summary statistics⁵. However, this summarization process can potentially discard a certain degree of information, thereby introducing additional bias and variance, especially when the summary proves inadequate in capturing all relevant details.

The second method centers around map-based inference. Within this approach, the likelihood is computed without resorting to any summary statistics. To handle the computationally intensive nature of this calculation, a graph convolutional neural network is employed—a focal point in my Master's thesis. The ensuing step involves a comparison between the posterior distributions yielded by these two methods. This comparative analysis aims to discern potential improvements achieved through the map-based inference technique with compared to the traditional summary statistics-based approach.

³Credible interval is used in Bayesian statistics, while frequentists use confidence interval.

⁴Uniform priors are commonly called 'uninformative priors,' but this terminology can be misleading because even specifying a uniform prior imparts some information by restricting the parameter within a defined range.

⁵Primarily, 2-point summary statistics, as elaborated upon in section 2.3.

2.2 Random Field

This section delves into the concept of random fields and the probability distributions that will be employed to model the cosmic density field in this thesis—specifically, the Gaussian distribution and the shifted lognormal distribution.

Randomness is a ubiquitous phenomenon in our world, manifesting in various contexts such as the formation and evolution of cosmic structures [104], forecasting tomorrow’s weather conditions, and whether the ticket controllers of Deutsche Bahn inspect your ticket on the train⁶. A random field comprises an assortment of random variables, serving as a framework to construct statistical models that furnish probable answers to questions entailing inherent randomness. Within this project, our focus centers on weak lensing convergence fields, which function as representative instances of random fields⁷. These fields correspond to two-dimensional projections portraying density fluctuations within the dark matter composition.

2.2.1 Probability Density Function

The Probability Density Function (PDF) is a key tool for fully characterizing a random field [40]. Let’s first consider a random field described by a finite-dimensional random vector $F = (F_1, \dots, F_N)$ where each F_i represents a scalar real random variable. In this context, the PDF of the random field becomes an N -dimensional function [40],

$$p : \mathbb{R}^N \rightarrow \mathbb{R}, \quad (2.4)$$

and it satisfies normalization with respect to any variables f :

$$p \geq 0, \quad \int d^N f p(f) = 1. \quad (2.5)$$

Consequently, the probability of the random field F falling within any subvolume $V \in \mathbb{R}$ can be expressed as follows:

$$P(F \in V) = \int_{f \in V} d^N f p(f) \leq 1. \quad (2.6)$$

Now, let’s consider the scenario when we move to a 1D case. We focus on a continuous variable $X \in \mathbb{R}$ distributed according to the PDF $p \in \mathbb{R}$. The analogous normalization statement, equivalent to (2.5), holds for individual x values within \mathbb{R} :

$$1 = \int dx p(x). \quad (2.7)$$

⁶Traveling on trains in Germany without a valid ticket or intentionally avoiding payment is considered an offense under §265a of the German Criminal Code (StGB).

⁷The derivation of the convergence field will be discussed in section 4.

For a finite interval $[x_1, x_2] \in X$, the probability of X falling within this interval can be expressed analogously to (2.6):

$$\int_{x_1}^{x_2} dx P(x) \equiv P(X \in [x_1, x_2]) \leq 1 . \quad (2.8)$$

The expectation value, which represents the average outcome of a continuous random variable, can be written as follows:

$$\langle X \rangle = \int dx x p(x) . \quad (2.9)$$

Moving on to generalized expectations, moments play a crucial role. The n -th order moment of X is defined as[40],

$$\langle X^n \rangle = \int dx x^n p(x) . \quad (2.10)$$

Assuming that all moments of random variables are finite, we can express the n -th moment in terms of the n -th derivative of the moment generating function (MGF) $\psi(\lambda)$ as follows[40],

$$\langle X^n \rangle = \left. \frac{d^n \psi(\lambda)}{d\lambda^n} \right|_{\lambda=0} . \quad (2.11)$$

Here, $\psi(\lambda)$ is defined as,

$$\psi(\lambda) \equiv \langle e^{\lambda X} \rangle \equiv \sum_{n=0}^{\infty} \frac{\langle X^n \rangle}{n!} \lambda^n = \int dx e^{\lambda x} p(x) . \quad (2.12)$$

The last equality in (2.12) makes use of the Laplace transform. Another crucial tool for characterizing higher-order moments is the cumulant generating function $\varphi(\lambda)$. It's defined as the natural logarithm of the MGF[40],

$$\varphi(\lambda) = \ln(\psi(\lambda)) . \quad (2.13)$$

Furthermore, the cumulant generating function φ can be expressed in terms of the Maclaurin series of cumulants $\langle X^n \rangle_c$ ⁸ and λ as follows[40]:

$$\varphi(\lambda) = \sum_{n=1}^{\infty} \frac{\langle X^n \rangle_c}{n!} \lambda^n . \quad (2.14)$$

An intriguing aspect of cumulants is that the higher moments of cumulants form polynomial functions of central moments⁹. In essence, cumulants involve subtracting the contributions of all combinations of lower moments from the higher moment.

⁸In addition to cumulants, another tool for characterizing non-Gaussian distributions is the Gram-Charlier expansion. In this expansion, the PDF is expressed in terms of a series of Hermite polynomials. This technique also allows for the calculation of skewness and kurtosis and can also be extended to the multivariate case[44].

⁹For the 2nd and 3rd orders, these polynomial functions coincide with central moments. However, for instance, the 4th order of cumulants is not identical to the central moment, as (2.14) yields $\langle X^4 \rangle_c = \langle (X - \langle X \rangle)^4 \rangle - 3(\langle (X - \langle X \rangle)^2 \rangle)^2$.

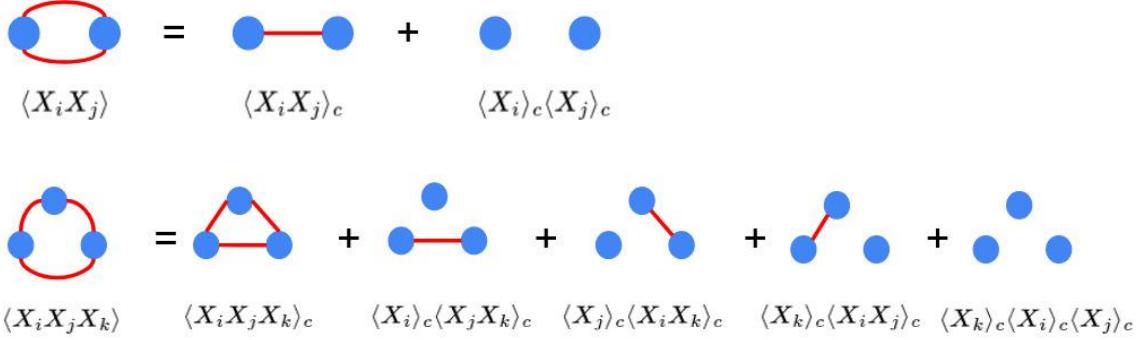


Figure 2.1: Geometrical illustration of the relation between cumulants and moments.

2.2.2 Gaussian Fields

First, we would like to introduce the Gaussian distribution¹⁰, a widely employed probability distribution, which is imperative due to its frequent emergence as a convergence point for statistical distributions as per the central limit theorem¹¹. In the realm of cosmology, the Gaussian field stands as an accurate model¹², representing temperature fluctuation fields within the Cosmic Microwave Background (CMB) sky map. These fluctuations are intricately linked to the cosmic density field in the early universe [97].

For an N -dimensional Gaussian field $\mathbf{F} \in \mathbb{R}^N$, the joint N -point probability density function (PDF) denoted as $p(\mathbf{F})$ is defined, incorporating the mean vector $\boldsymbol{\mu}$ and the covariance matrix \mathbf{C} [124]:

$$p(\mathbf{F}) = \frac{1}{\sqrt{(2\pi)^N \det(\mathbf{C})}} \exp\left(-\frac{1}{2}(\mathbf{F} - \boldsymbol{\mu})^T \mathbf{C}^{-1} (\mathbf{F} - \boldsymbol{\mu})\right). \quad (2.15)$$

The essence of the Gaussian random field lies in its characterization through only two fundamental parameters. Firstly, the mean vector $\boldsymbol{\mu}$ stands as the initial moment of the random variables:

$$\boldsymbol{\mu} \equiv (\langle F_1 \rangle, \dots, \langle F_N \rangle)^T, \quad (2.16)$$

where $\langle \cdot \rangle$ signifies the expectation values. The second defining element is the covariance matrix, a square matrix of size $N \times N$. The matrix entries signify the second central moment of random variables, pertaining to the elements F_i and F_j [39]:

$$\mathbf{C}_{i,j} \equiv \langle (F_i - \langle F_j \rangle)(F_j - \langle F_j \rangle) \rangle. \quad (2.17)$$

¹⁰Gaussian distribution is also known as Normal distribution.

¹¹The central limit theorem, although widely applicable, is not universally valid for all statistical distributions. A notable exception is the Cauchy distribution, which does not converge to the Gaussian distribution even with a substantial increase in sample size.

¹²Multiple-field inflation theories predict a significant amount of non-Gaussianity in the CMB temperature anisotropy [80]. However, it's worth noting that even in inflaton field inflation models, non-Gaussianity can arise due to the contribution of primordial gravitational waves through the tensor perturbations.

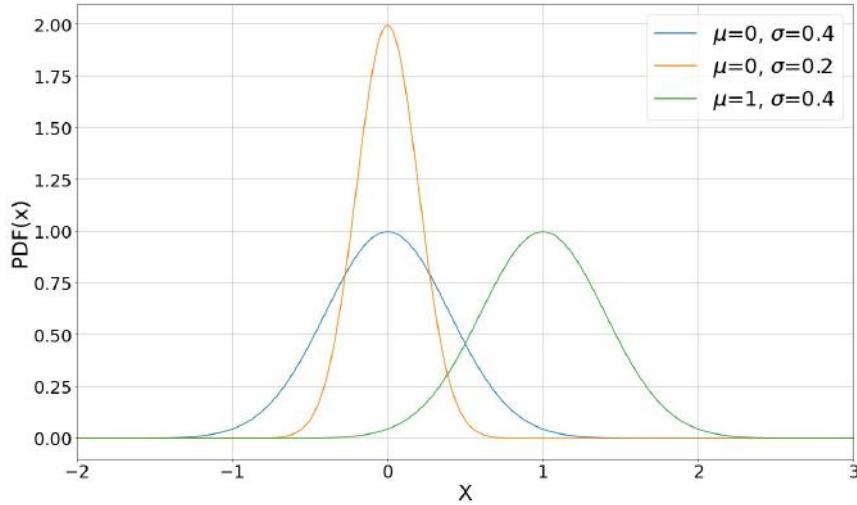


Figure 2.2: This figure illustrates the univariate Gaussian distribution. It can be characterized using only two parameters: the mean μ and the variance σ . The parameter μ determines the distribution's central location, while σ quantifies the extent of statistical dispersion in the distribution.

Assuming that the field \mathbf{F} can be realized through the random variable $F_i = F(x_i)$ ¹³ at a given point x_i , in cosmological contexts, the second central moment equates to the two-point correlation function $\xi(\mathbf{x}_i, \mathbf{x}_j)$:

$$\xi(\mathbf{x}_i, \mathbf{x}_j) \equiv \langle (F_i - \langle F_j \rangle)(F_j - \langle F_j \rangle) \rangle . \quad (2.18)$$

Similarly, higher moments can be defined, where the N -th order correlation function reflects the N -th order cumulant [39]:

$$\xi_N(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N) = \langle F_1 F_2 \dots F_N \rangle_c . \quad (2.19)$$

The variance σ_i^2 corresponds to the covariance element C_{ii} :

$$\sigma_i^2 = \mathbf{C}_{i,i} = \langle (F_i - \langle F_i \rangle)^2 \rangle . \quad (2.20)$$

A notable property of multivariate Gaussian distributions is that the marginal distribution of a 1-dimensional random variable can be characterized by a 1-point Gaussian PDF [124]. The 1-point PDF $p(F_i)$ for the Gaussian random variable F_i at the location x_i can be succinctly expressed as:

$$p(F_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{1}{2\sigma_i^2}(F_i - \mu_i)^2\right) . \quad (2.21)$$

¹³In this Master's thesis, we posit that the density fluctuations across the celestial sphere's surface constitute the random field of interest. This random field is discretized into a set of N pixels, where each location \mathbf{x}_i corresponds to a pixel with value F_i , realized through $F_i = F(\mathbf{x}_i)$.

Intriguingly, Isserlis' theorem¹⁴ [65] applies to this 1-point PDF. Higher central moments of Gaussian distributions can be expressed solely in terms of variances [124]:

$$\langle (F_i - \langle F_i \rangle)^n \rangle = \begin{cases} 0 & \text{if } n \text{ is odd,} \\ \frac{n!}{(\frac{n}{2})!} \left(\frac{\sigma_i^2}{2} \right)^{\frac{n}{2}} & \text{if } n \text{ is even.} \end{cases} \quad (2.22)$$

For a multivariate Gaussian distribution, higher moments can be fully delineated by the covariance matrix¹⁵, solidifying the sufficiency of covariance and mean for describing the Gaussian field. Conversely, identifying odd-order cumulants enables the detection of non-Gaussianity.

2.2.3 Lognormal Fields

In the subsequent chapter, it will be discussed that the initial cosmic density field closely adheres to a Gaussian distribution. However, this Gaussian nature does not persist during non-linear gravitational processes like gravitational collapse or the hierarchical merging of structures [39]. Consequently, in the late-time universe, an appreciable contribution from higher moments of the distribution emerges, especially on smaller scales. This renders the Gaussian distribution inadequate for describing the late-time cosmic density field. In this Master's thesis, the shifted log-normal distribution is employed to capture the characteristics of late-time density fields. Notably, the shifted log-normal distribution serves as a model for describing projected late-time density fields within test pipelines and estimators for data covariance in large-scale structure cosmology[124]. This subsection delves into the underlying mathematical principles of the shifted log-normal PDF.

To begin, we derive the log-normal random variable X_i . We start with the assumption that variables Z_i follow a Gaussian distribution with a mean μ_i and a covariance σ_i . Subsequently, log-normal random variables are expressed as follows:

$$X_i = e^{Z_i} - \lambda_i , \quad (2.23)$$

where λ_i represents the shift parameter [60]¹⁶. A univariate log-normal field can be effectively characterized using three parameters: the mean μ_i , acting as the scaling parameter; the variance σ_i , serving as the shape parameter; and λ_i , functioning as the location parameter. In contrast to Gaussian variables, log-normal variables possess an additional parameter, rendering them more adaptable for capturing non-Gaussian features in the late-time universe.

¹⁴In the context of classical probability theory or statistics, Isserlis' theorem is used, whereas Wick's theorem is employed in the context of quantum field theory.

¹⁵In this context, even-order central moments can be expressed using the covariance matrix elements, while odd-order central moments, similar to the case in 1-dimension, evaluate to 0.

¹⁶In [116], $-\lambda_i$ is referred to as 'minimum values', whereas, in this thesis, λ_i is denoted as the lognormal shift parameter.

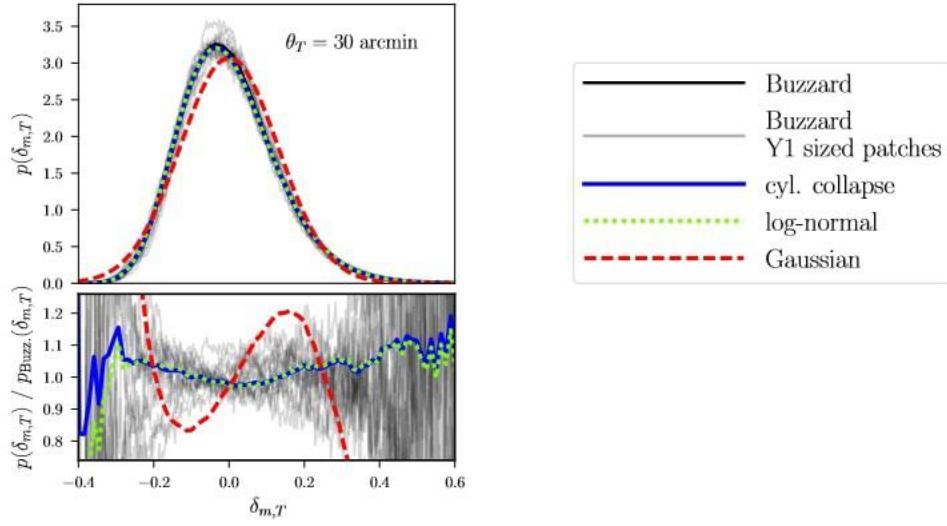


Figure 2.3: The PDF of the projected matter density contrast field $\delta_{m,T}$ adapted from [39]. The theoretical PDF of a lognormal field is depicted by the green dotted line, while the red dashed line represents a Gaussian field with identical mean and variance as the lognormal field. As a consequence of the presence of odd moments, lognormal models offer improved description capabilities for the late-time projected field.

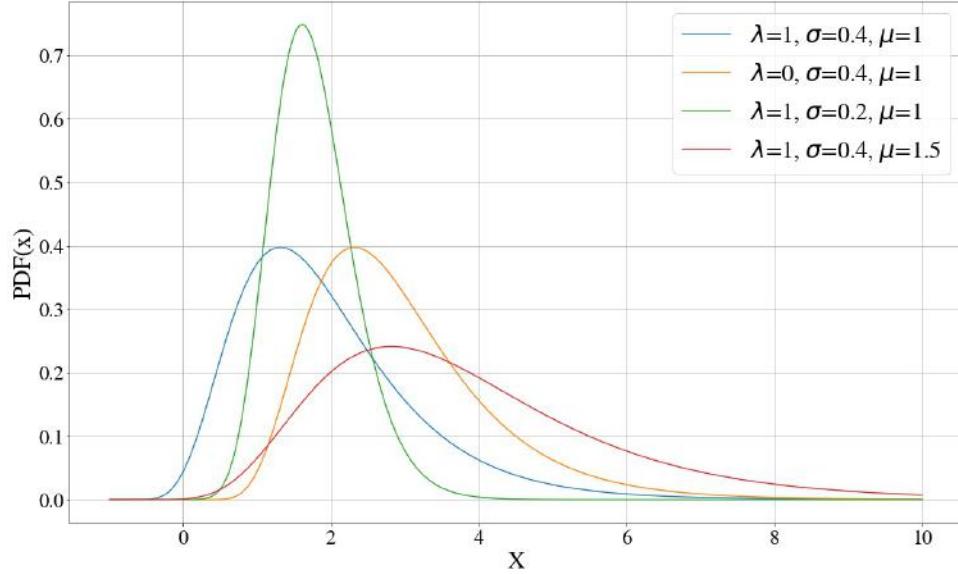


Figure 2.4: Lognormal PDFs are illustrated in this figure. Altering λ results in a shifted location. Increasing μ corresponds to stretching the distribution towards the right. Reducing σ changes the shape of the distribution, bringing it closer to a Gaussian distribution.

Leveraging Isserlis's theorem (2.22), the log-normal shift parameter can be expressed with the mean μ_i and the variance σ_i of the corresponding Gaussian distribution, along

with the mean of the log-normal random variable $\langle X_i \rangle^{17}$ [124]:

$$\lambda_i = e^{\mu_i + \frac{\sigma_i}{2}} - \langle X_i \rangle . \quad (2.24)$$

Utilizing this relation, the 1-point PDF $p(X_i)$ of the shifted log-normal distribution can be presented as [60]:

$$p(X_i) = \begin{cases} \frac{\exp\left(-\frac{1}{2\sigma_i}[\ln(X_i + \lambda_i) - \mu_i]^2\right)}{\sqrt{2\pi}\sigma_i(X_i + \lambda_i)} & \text{for } X_i > -\lambda_i \\ 0 & \text{otherwise,} \end{cases} \quad (2.25)$$

Expanding further to encompass the multi-variate log-normal field, we consider log-normal random variables $\mathbf{X} = (X_1, X_2, X_3, \dots, X_N)^T$ and their associated Gaussian random variables $\mathbf{Z} = (Z_1, Z_2, Z_3, \dots, Z_N)^T$ with a mean vector $\boldsymbol{\mu}$ and covariance ξ_G^{ij} . Consequently, the multi-variate log-normal N -point PDF $p(\mathbf{Z})$ can be formulated as [43]:

$$p(\mathbf{X}) = \begin{cases} \frac{\exp\left(-\frac{1}{2}[\ln(\mathbf{X} + \boldsymbol{\lambda}) - \boldsymbol{\mu}]^T \xi_G^{ij}^{-1} [\ln(\mathbf{X} + \boldsymbol{\lambda}) - \boldsymbol{\mu}]\right)}{\sqrt{(2\pi)^N \det(\xi_G^{ij}) \prod_{i=1}^N (X_i + \lambda_i)}} & \text{for } \mathbf{X} > -\boldsymbol{\lambda} \\ 0 & \text{otherwise,} \end{cases} \quad (2.26)$$

Here, $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_N)^T$ represents the shift parameter vector. We can then deduce the covariance matrix ξ_{\ln}^{ij} of two log-normal variables X_i and X_j in terms of the covariance matrix of the associated Gaussian distribution ξ_G^{ij} using (2.24) [124]:

$$\begin{aligned} \xi_{\ln}^{ij} &\equiv \langle X_i X_j \rangle - \langle X_i \rangle \langle X_j \rangle \\ &= \alpha_i \alpha_j (e^{\xi_G^{ij}} - 1) . \end{aligned} \quad (2.27)$$

Consequently, we arrive at the following relationship:

$$\xi_G^{ij} = \ln \left(\frac{\xi_{\ln}^{ij}}{\alpha_i \alpha_j} + 1 \right) . \quad (2.28)$$

Here, $\alpha_i \equiv \langle X_i \rangle + \lambda_i > 0$. Analogously, employing (2.24), we can represent the three-point correlation ζ_{\ln}^{ijk} for the log-normal field in terms of the covariance matrix ξ_{\ln}^{ij} [124]:

$$\begin{aligned} \zeta_{\ln}^{ijk} &\equiv \langle (X_i - \langle X_i \rangle)(X_j - \langle X_j \rangle)(X_k - \langle X_k \rangle) \rangle \\ &= \frac{\xi_{\ln}^{ij} \xi_{\ln}^{jk} \xi_{\ln}^{ki}}{\alpha_i \alpha_j \alpha_k} + \frac{\xi_{\ln}^{ij} \xi_{\ln}^{ki}}{\alpha_i} + \frac{\xi_{\ln}^{ij} \xi_{\ln}^{jk}}{\alpha_j} + \frac{\xi_{\ln}^{ki} \xi_{\ln}^{jk}}{\alpha_k} . \end{aligned} \quad (2.29)$$

While a Gaussian random field lacks odd-numbered higher moments due to Isserlis's theorem (2.22), the log-normal distribution exhibits these odd-numbered higher moments as indicated by the above equation. Subsequently, consider a local position X_i within the

¹⁷In this thesis, we are exploring the general case. However, it's important to note that in the context of the cosmic density field, the mean is zero by definition.

log-normal field. The variance of the log-normal field, denoted as v_i , can be expressed using (2.27) in terms of the variance of the associated Gaussian distribution σ_i :

$$v_i \equiv \xi_{\ln}^{ij} = \alpha_i^2 (e^{\sigma_i^2} - 1) . \quad (2.30)$$

Utilizing (2.29), the skewness γ_i of a log-normal random variable can be computed from the variance v_i as follows:

$$\gamma_i \equiv \frac{\zeta_{\ln}^{iii}}{v_i^{\frac{3}{2}}} = \frac{\sqrt{v_i}}{\alpha_i} \left(\frac{v_i}{\alpha_i^2} + 3 \right) . \quad (2.31)$$

By calculating α_i , we can deduce the log-normal shift parameter λ_i at the specific location X_i , expressed in terms of the mean, variance, and skewness of the log-normal random variable [124]:

$$\lambda_i = \frac{\sqrt{v_i}}{\gamma_i} \left(1 + y(\gamma_i) + \frac{1}{\gamma_i} \right) - \langle X_i \rangle . \quad (2.32)$$

Here, $y(\gamma)$ is defined as:

$$y(\gamma) \equiv \sqrt[3]{\frac{2 + \gamma^2 + \gamma\sqrt{4 + \gamma^2}}{2}} . \quad (2.33)$$

Utilizing (2.24) and (2.30), we can determine the mean μ_i and variance σ_i^2 of the associated Gaussian distribution at the specific location X_i in terms of the lognormal distribution's moments:

$$\mu_i = \ln \left(\frac{\alpha_i^2}{\sqrt{\alpha_i^2 + v_i}} \right) \quad (2.34)$$

$$\sigma_i = \sqrt{\ln \left(1 + \frac{v_i}{\alpha_i} \right)} . \quad (2.35)$$

In order to construct the log-normal field, the mean μ_i and variance v_i of the given Gaussian field, alongside the shift parameter λ_i , are required components.

2.3 Statistics of Cosmic Density Field

In the previous section, we acquainted ourselves with the concept of a random field. The next chapter will delve into the theoretical description of the evolution of the density contrast field. Bridging the gap between theory and observation necessitates a robust statistical framework to characterize the density field. Presently, our focus shifts to introducing foundational statistical methodologies essential for the analysis of the cosmic random field.

2.3.1 Statistically Homogenous and Isotropic Field

In cosmology, the cosmological principle is often embraced, ensuring the statistical homogeneity and isotropy of the density field. Our focus now shifts to introducing these foundational assumptions underpinning the modeling of the cosmic density field.

To commence, let us define the density contrast field δ at a given position \mathbf{x} as follows:

$$\delta(\mathbf{x}) = \frac{\rho(\mathbf{x}) - \langle \rho \rangle}{\langle \rho \rangle}. \quad (2.36)$$

Here, $\rho(\mathbf{x})$ represents the density. It is evident that the mean of this field, denoted as $\langle \delta \rangle$, is inherently zero by definition.

Consider, for simplicity, a Gaussian random field. A zero-mean Gaussian field¹⁸ can be uniquely characterized by its covariance matrix ξ_G , a consequence derived from Isserlis's theorem (2.22). The two-point correlation of the density contrast field between points \mathbf{x}_i and \mathbf{x}_j simply amounts to the covariance:

$$\xi_G(\mathbf{x}_i, \mathbf{x}_j) = \langle \delta(\mathbf{x}) \delta(\mathbf{x} + \mathbf{r}) \rangle. \quad (2.37)$$

When this two-point correlation function $\xi_G(\mathbf{x}_i, \mathbf{x}_j)$ exclusively hinges on the separation vector $\mathbf{r} = \mathbf{x}_i - \mathbf{x}_j$, it follows that:

$$\xi_G(\mathbf{x}_i, \mathbf{x}_j) = \xi_G(\mathbf{r}). \quad (2.38)$$

This Gaussian field can hence be characterized as statistically homogeneous¹⁹.

Now, let's examine the subsequent assumption. If the two-point correlation function $\xi_G(\mathbf{x}_i, \mathbf{x}_j)$ is independent of direction, then employing assumption (2.37), the two-point correlation function can be expressed in terms of the separation $r = |\mathbf{x}_i - \mathbf{x}_j|$:

$$\xi_G(\mathbf{x}_i, \mathbf{x}_j) = \xi_G(|\mathbf{x}_i - \mathbf{x}_j|) = \xi_G(r). \quad (2.39)$$

Such a Gaussian field would be both statistically homogeneous and isotropic.

Cosmological observations [94] assure us of statistical homogeneity on scales larger than 100 Mpc²⁰, as well as statistical isotropy. This implies that the statistical properties of our universe remain invariant under spatial rotation and translation²¹. In a generally statistically homogeneous and isotropic field, the N -point correlation function can be expressed in terms of a rotation matrix \mathbf{R} and a translational vector \mathbf{a} as follows:

$$\xi(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_N) = \xi(\mathbf{R} \cdot \mathbf{x}_1 + \mathbf{a}, \mathbf{R} \cdot \mathbf{x}_2 + \mathbf{a}, \mathbf{R} \cdot \mathbf{x}_3 + \mathbf{a}, \dots, \mathbf{R} \cdot \mathbf{x}_N + \mathbf{a}). \quad (2.40)$$

¹⁸As you can see in Equation (2.36), the density contrast field is a zero-mean field by definition.

¹⁹This holds true as the cosmic density field inherently possesses a zero-mean characteristic. Consequently, the mean vector μ remains independent of any specific location \mathbf{x} .

²⁰Some researchers report inhomogeneous cosmic structure beyond this scale[48].

²¹Another important symmetry to consider is spatial parity symmetry, which becomes crucial when calculating the bispectrum.

In the context of non-Gaussian fields, validating homogeneity or isotropy necessitates examining higher-order correlation functions. However, for the purposes of this project, we will proceed with the assumption of a statistically homogeneous and isotropic field, as a comprehensive investigation into the validity of this assumption falls beyond the scope of this project.

2.3.2 Random Field in Fourier Space

In the realm of observational cosmology, theoretical models are often conceptualized and worked upon in Fourier space. This preference is based on the independent behavior of density contrast of each mode in the linear regime. The subsequent chapter will delve into our exploration of power spectrum modeling within the framework of the Λ CDM cosmology. However, before delving into that, we shall examine the intricate relationship between the power spectrum that we can theoretically model and the two-point correlation function that we can directly extract from observations.

We commence by assuming a flat universe, allowing us to represent the cosmic density contrast field $\delta(\mathbf{x})$ defined within a volume V in Fourier space [89]:

$$\delta(\mathbf{x}) \propto \frac{1}{\sqrt{V}} \sum_k \delta_k e^{i\mathbf{k} \cdot \mathbf{x}}. \quad (2.41)$$

Here, $\delta_k \equiv \tilde{\delta}(\mathbf{k})$ represents the Fourier coefficient²². Importantly, as the cosmological field is real, the following relationship holds true:

$$\tilde{\delta}(-\mathbf{k}) = \tilde{\delta}^*(\mathbf{k}). \quad (2.42)$$

Now, we transition towards the continuous limit as $V \rightarrow \infty$, transforming equation (2.41) into the form [29]:

$$\delta(\mathbf{x}) = \frac{1}{(2\pi)^3} \int d^3\mathbf{k} \tilde{\delta}(\mathbf{k}) e^{i\mathbf{k} \cdot \mathbf{x}}. \quad (2.43)$$

Considering the two-point correlation function of the density contrast field $\tilde{\delta}(\mathbf{k})$ in Fourier space at points \mathbf{k}_1 and \mathbf{k}_2 (corresponding to real space points \mathbf{x}_1 and \mathbf{x}_2), we can express it²³ using equation (2.43):

$$\langle \tilde{\delta}(\mathbf{k}_1), \tilde{\delta}^*(\mathbf{k}_2) \rangle = \int d^3x_1 d^3x_2 \langle \delta(\mathbf{x}_1) \delta(\mathbf{x}_2) \rangle e^{-i\mathbf{k}_1 \mathbf{x}_1 + i\mathbf{k}_2 \mathbf{x}_2}. \quad (2.44)$$

Assuming a statistically homogeneous universe, the two-point correlation function $\langle \delta(\mathbf{x}_1) \delta(\mathbf{x}_2) \rangle$ simplifies to $\xi_G(\mathbf{r})$, where $\mathbf{r} = \mathbf{x}_2 - \mathbf{x}_1$ represents the spatial separation. This allows us to represent the covariance in Fourier space as:

$$\langle \tilde{\delta}(\mathbf{k}_1), \tilde{\delta}^*(\mathbf{k}_2) \rangle = \left(\int d^3x_1 e^{i(\mathbf{k}_2 - \mathbf{k}_1) \mathbf{x}_1} \right) \left(\int d^3r \xi(\mathbf{r}) e^{i\mathbf{k}_2 \mathbf{r}} \right). \quad (2.45)$$

²²It is generally complex quantity.

²³The second term in the exponential function has a negative sign due to the utilization of the relation (2.42).

The first integral can be expressed using the Dirac delta function²⁴ in 3D space:

$$\delta_D^{(3)}(\mathbf{k}) = \frac{1}{(2\pi)^3} \int d^3x e^{i\mathbf{k}\mathbf{x}} . \quad (2.46)$$

Defining the second term as the power spectrum $P(\mathbf{k})$ ²⁵:

$$P(\mathbf{k}) = \int d^3r \xi(\mathbf{r}) e^{i\mathbf{k}\mathbf{r}} . \quad (2.47)$$

We arrive at the following relation²⁶:

$$\langle \delta(\mathbf{k}) \delta(\mathbf{k}') \rangle = (2\pi)^3 \delta_D(\mathbf{k} + \mathbf{k}') P(k) . \quad (2.48)$$

Further, assuming a statistically isotropic and homogeneous universe leads to the relation between the two-point function and the power spectrum:

$$\xi(r) = \frac{1}{(2\pi)^3} \int d^3k P(k) e^{ikr} . \quad (2.49)$$

To facilitate dimensional consistency²⁷, we introduce the dimensionless power spectrum [92]:

$$\Delta^2(k) \equiv \frac{k^3}{2\pi^2} P(k) . \quad (2.50)$$

The requirement for the two-point correlation to be a real function by definition leads to:

$$\begin{aligned} \xi(r) &= \frac{1}{2\pi^2} \int P(k) \frac{\sin(kr)}{kr} k^2 dk \\ &= \int \frac{dk}{k} \Delta^2(k) j_0(kr) , \end{aligned} \quad (2.51)$$

where j_0 represents the spherical Bessel function of order zero. Notably, the variance can be derived when $r = 0$:

$$\xi(0) = \int d\ln k \Delta^2(k) . \quad (2.52)$$

This observation elucidates that a constant dimensionless power spectrum results in scale-invariant, meaning that the amplitude of density fluctuations is the same on all length scales.²⁸.

²⁴This delta 'function' is not rigorously a function, as it is almost everywhere zero with respect to the Lebesgue measure and has a positive integral. A more rigorous definition of this is as a distribution.

²⁵Why is it called a 'power' spectrum? Because it quantifies the power of the amplitude for a given frequency \mathbf{k} .

²⁶This is beyond the scope of this project. However, this relation also holds for the n -th order polyspectrum (n -th order cumulants) with $\langle \delta(\mathbf{k}_1) \cdots \delta(\mathbf{k}_n) \rangle_c = (2\pi)^3 \delta_D^3(\mathbf{k}_1 + \cdots + \mathbf{k}_n) P^{(n)}(\mathbf{k}_1, \dots, \mathbf{k}_n)$.

²⁷Since the Dirac delta distribution has dimensions of \mathbf{k}^{-3} .

²⁸This scale-invariant power spectrum is called Harrison-Zeldovich spectrum [10]

2.3.3 Density Field on the Sphere

Observational cosmology, including surveys like the Dark Energy Survey [28], often lacks precise positional information for galaxies due to uncertainties in the redshift distribution of these tracers. Consequently, this thesis focuses on studying a projected density field. Specifically, the 2D projections of the density field $\delta_{m,2D}$ along a given redshift kernel $q(z)$ for a particular sky position $\hat{\mathbf{n}}$ can be represented through the line-of-sight integral [39],

$$\delta_{m,2D}(\hat{\mathbf{n}}) = \int_0^\infty q(z) \delta_m(\hat{\mathbf{n}} \cdot \mathbf{r}(z), z) . \quad (2.53)$$

In this context, $\hat{\mathbf{n}}$ is a unit vector in spherical coordinates ($\hat{\mathbf{n}} = (\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta)$). The projected density $\delta_{m,2D}$ resides on the surface of a 2D unit sphere and can be expanded into spherical harmonics Y_{lm} along with their corresponding spherical coefficients a_{lm} ,

$$\delta_{m,2D}(\hat{\mathbf{n}}) = \sum_{l=0}^{\infty} \sum_{m=-l}^l a_{lm} Y_{lm}(\hat{\mathbf{n}}) . \quad (2.54)$$

Here, the spherical harmonics are defined as exponential functions of ϕ [15],

$$Y_{lm} \equiv (-1)^m \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}} P_l^m(\cos\theta) \exp(im\phi) . \quad (2.55)$$

These spherical harmonics form an orthonormal basis [15], leading to the definition of the angular power spectrum C_l as the variance of the coefficients,

$$\langle a_{lm} a_{l'm'} \rangle = \delta_{ll'} \delta_{mm'} C_l . \quad (2.56)$$

This angular power spectrum C_l can also be interpreted as the square average of the coefficients a_{lm} over all possible m values,

$$C_l = \frac{1}{2l+1} \sum_{m=-1}^l |a_{lm}|^2 . \quad (2.57)$$

Using this framework, an analogous concept of the angular two-point correlation function can be defined [39],

$$\xi_{2D}(\triangleleft \hat{\mathbf{n}} \hat{\mathbf{n}}') = \langle \delta_{m,2D}(\hat{\mathbf{n}}) \delta_{m,2D}(\hat{\mathbf{n}}') \rangle . \quad (2.58)$$

Due to the assumption of statistical isotropy in our universe, the angular power spectrum can be described solely in terms of the angle $\triangleleft \hat{\mathbf{n}} \hat{\mathbf{n}}'$ between two unit vectors. Utilizing the properties of spherical harmonics, we derive the connection between the angular two-point correlation function and the angular power spectrum,

$$\xi_{2D}(\theta) = \sum_l \frac{2l+1}{4\pi} P_l(\cos\theta) C_l . \quad (2.59)$$

2.3.4 Limber Integration

The Limber integral plays a pivotal role in bridging the angular power spectrum in the sky (2D) with the power spectrum in 3D space, as initially formulated by Limber [78]. Within the context of photometric galaxy surveys, a quantity of paramount interest is the Limber quantity. This observable is the result of a projection through the Limber integral, primarily due to the inherent challenge in acquiring information directly along the line of sight. In the ensuing discussion, we delve into the derivation of the Limber integral.

The angular two-point correlation function, $\xi_{2D}(\theta)$, encapsulates the correlation between pairs of points separated by an angle θ on the celestial sphere. Its formulation reads as follows [39]:

$$\xi_{2D}(\theta) = \int_0^\infty dr_1 \int_0^\infty dr_2 q(r_1)q(r_2) \langle \delta_{3D}(\hat{\mathbf{n}}_1, \mathbf{r}_1, z_1) \delta_{3D}(\hat{\mathbf{n}}_2, \mathbf{r}_2, z_2) \rangle . \quad (2.60)$$

Here, $q(r)$ represents a projection kernel that captures the line-of-sight distribution, and $\delta_{3D}(\hat{\mathbf{n}}, \mathbf{r}, z)$ denotes the 3D density fluctuation field at position \mathbf{r} and redshift z in the direction defined by the unit vector $\hat{\mathbf{n}}$. To proceed, we introduce a novel vector $\delta\mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1$, while making use of the flat sky approximation where $\delta\mathbf{r} \ll r$. This enables us to make approximations such as $z(\mathbf{r}) \approx z(\mathbf{r} + \delta\mathbf{r})$ and $q(r) \approx q(\mathbf{r} + \delta\mathbf{r})$ ²⁹.

With these approximations in place, we can re-express ξ_{2D} in terms of the 3D correlation function, ξ_{3D} :

$$\xi_{2D}(\theta) = \int_0^\infty dr q(r) \int_0^\infty d\Delta r \xi_{3D}(|\hat{\mathbf{n}}_1 r - \hat{\mathbf{n}}_2(r + \Delta r)|) . \quad (2.61)$$

Furthermore, by utilizing the flat universe³⁰ and once again invoking the flat sky approximation to obtain the small angle approximation $\cos\theta \approx 1 - \frac{\theta^2}{2}$, we arrive at:

$$|\hat{\mathbf{n}}_1 r - \hat{\mathbf{n}}_1(r + \Delta r)|^2 \approx \Delta^2 r^2 + r^2 \theta^2 . \quad (2.62)$$

This enables us to recast ξ_{2D} in terms of the 3D power spectrum in Fourier space:

$$\begin{aligned} \xi_{2D}(\theta) &= \int_\infty^\infty dr q^2(r) \int \frac{d^2 k}{(2\pi)^2} k P_{3D}(z(r), k) \exp(ir\theta k) \\ &= \int_\infty^\infty dr q^2(r) \int \frac{d^2 k}{(2\pi)^2} k P_{3D}(z(r), k) J_0(kr\theta) , \end{aligned} \quad (2.63)$$

where J_0 denotes the zeroth-order spherical Bessel function. By introducing the relationship $l + \frac{1}{2} = kr$, we arrive at the angular power spectrum C_l :

$$C_l = \int d^2\theta \xi_{2D}(\theta) \exp\left(i\left(1 + \frac{l}{2}\right)\theta\right) \quad (2.64)$$

$$= \int_0^\infty dr \frac{q^2(r)}{r^2} P_{3D}\left(\frac{l + \frac{1}{2}}{r}, z(r)\right) . \quad (2.65)$$

²⁹Since the projection kernel decays faster.

³⁰Not necessarily assuming this; the derivation in [6] did not use the comoving distance for a flat universe.

The expression (2.65) embodies the concept of the Limber integral and signifies the line of projection. The projection kernel $q(r)$ allows us to incorporate the line-of-sight distribution of a specific quantity, such as the redshift distribution. Further insights into this will be presented in Chapter 4.

Chapter 3

Theory of Cosmology

This Master's thesis focuses on the regression problem involving the parameters of the Λ CDM model, which are inferred from a simulated projected density field. This inference process is motivated by the analysis of weak gravitational lensing. Consequently, a basic understanding of the Λ CDM theory is imperative.

The Λ CDM theory, an extension of the Big Bang theory, stands as the prevailing framework for numerous cosmological analyses [97], [28], [25]. This theory comprises two fundamental constituents: the cosmological constant¹, denoted as Λ , serving as late-time dark energy, and cold dark matter, abbreviated as CDM.

The triumphs of the Λ CDM cosmology are manifold, encompassing the explication of expansion history, as well as the formation of cosmic structures. It notably predicts phenomena such as the baryonic acoustic oscillation feature [32] and elucidates the observed peaks in the power spectrum of the cosmic microwave background [97]. Additionally, the framework furnishes constraints on cosmological parameters. Nevertheless, tensions between cosmological parameters, like H_0 and S_8 , have surfaced across various observational methods.

This chapter explores the basic principles that form the foundation of Λ CDM cosmology. It also provides an overview of how structures in our universe form within this theoretical framework. For this purpose, this chapter is mainly based on the books, [29], [89], [10], [122] and [104].

3.1 General Relativity

The foundation of the Λ CDM cosmology rests upon two fundamental hypotheses, in addition to the incorporation of two dark components. The initial postulate posits that the gravitational interactions governing cosmic structures adhere to the principles of general relativity. Ever since Erwin Hubble's observation of Cepheid in 1929, wherein he measured

¹One candidate for the cosmological constant is vacuum energy, but there is a difference of 120 orders of magnitude between the prediction from quantum field theory (QFT) and measurement (V. Mukhanov, Lecture on Cosmology, 2022, Munich).

the recessional velocities of galaxies [63], a wealth of evidence has amassed corroborating the expansion of our universe [95], [97], [28]. Consequently, the framework of general relativity is aptly applied in cosmology, given that our universe can be elegantly encapsulated within the construct of a 4-dimensional spacetime, encompassing temporal as well as spatial dimensions.

The second fundamental assumption revolves around the cosmological principle, which asserts the homogeneity and isotropy of the spatial distribution of matter on a grand scale within the universe. While this principle holds statistically for scales surpassing 100Mpc [94], it falters on smaller scales, where intricate, highly inhomogeneous structures—like galaxy clusters and individual galaxies—take form. These structures, embodying density fluctuations, bear testimony to the intricate tapestry of our universe's composition². Despite the localized deviations from homogeneity and isotropy, this assumption perseveres as a valid statistical framework.

Subsequently, our discourse delves into the intrinsic nature of general relativity within the framework of homogeneity and isotropy.

3.1.1 Geometry of Curved Space

The geometric characteristics of curved space-time are elucidated through the metric tensor denoted by $g_{\mu\nu}$. The curvature inherent to a Riemannian manifold finds quantification in the Riemann curvature tensor $R_{\mu\alpha\beta}^\sigma$, whose components can be expressed in terms of the Affine connection. This connection is encapsulated by the Christoffel symbol $\Gamma_{\mu\nu}^\sigma$ and is defined as follows:

$$\Gamma_{\mu\nu}^\sigma = \frac{1}{2}g^{\sigma\rho}(\partial_\mu g_{\nu\rho} + \partial_\nu g_{\rho\mu} - \partial_\rho g_{\mu\nu}) . \quad (3.1)$$

The Ricci tensor $R_{\mu\nu}$ emerges from the contraction of the Riemann tensor, as captured by the equation:

$$R_{\mu\nu} = R_{\mu\rho\nu}^\rho . \quad (3.2)$$

Subsequently, the Ricci scalar is established as the contraction of the Ricci tensor itself:

$$R = R_\mu^\mu = g^{\mu\nu}R_{\mu\nu} . \quad (3.3)$$

Leveraging equations (3.2) and (3.3), we formulate the Einstein tensor:

$$G_{\mu\nu} = R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} . \quad (3.4)$$

This Einstein tensor possesses 10 independent components within 4-dimensional space-time owing to its symmetry³, offers a profound representation of the gravitational field

²There is another framework called modified gravity (MOND) that may contribute to our understanding of the intricate tapestry of our universe's composition without invoking dark matter.

³The Einstein tensor has a total of 10 independent components in 4-dimensional space-time: 4 diagonal components and 6 symmetrically arranged non-diagonal components.

dynamics. An intriguing property emerges from the symmetries inherent in the Riemann tensor: the divergence of the Einstein tensor vanishes, manifesting as:

$$\nabla^\mu G_{\mu\nu} = 0 . \quad (3.5)$$

3.1.2 The Einstein Field Equations

The Einstein field equation establishes the profound connection between the curvature of space-time and the distribution of energy and momentum within it [29]. This relationship can be succinctly expressed through the energy-momentum tensor:

$$G_{\mu\nu} = 8\pi GT_{\mu\nu} , \quad (3.6)$$

In this equation, G represents Newton's gravitational constant. Analogous to the structure of (3.4), both sides of (3.6) manifest as 4-dimensional rank-2 symmetric tensors, giving rise to 10 distinct partial differential equations. Four of these equations correspond to conservation laws, as the divergence of the energy-momentum tensor is inherently fulfilled within the Einstein field equation due to (3.4):

$$\nabla^\mu T_{\mu\nu} = 0 . \quad (3.7)$$

The remaining six second-order partial differential equations, or equivalently, twelve first-order partial differential equations, encapsulate the dynamics of the space-time itself. Remarkably, owing to the Bianchi identities, and taking into account the four gauge freedoms arising from coordinate choices in four-dimensional space-time, we can condense the Einstein field equations to four independent equations. Consequently, to satisfactorily resolve this set of equations, an additional two second-order partial differential equations or four first-order partial differential equations are indispensable.

3.1.3 Friedmann–Lemaître–Robertson–Walker Metric

Utilizing the cosmological principle, we embrace the Friedmann–Lemaître–Robertson–Walker (FLRW) metric as a descriptor of the universe's geometry. In the spherical coordinate system (r, θ, ϕ) , employing natural units where $c = 1$ and adhering to the $(+, -, -, -)$ convention[89], the line element of this metric, denoted d^2s , is formulated as follows [89]:

$$ds^2 = g_{\mu\nu} dx^\mu dx^\nu = dt^2 - a^2(t) \left(\frac{dr^2}{1 - Kr^2} + r^2(d\theta^2 + \sin^2 \theta d\phi^2) \right) . \quad (3.8)$$

Here, μ and ν a natural number from 0 to 3, $a(t)$ represents the scale factor signifying the universe's relative size at a given time compared to a reference time, and K denotes a constant curvature that characterizes the topological nature of the spatial hypersurface in the Friedmann universe.

Introducing the radial spatial coordinate in comoving coordinates, we have:

$$d\chi = \frac{dr}{\sqrt{1 - Kr^2}} . \quad (3.9)$$

This gives rise to the expression:

$$ds^2 = dt^2 - a^2(t) [d\chi^2 + S_K^2(\chi)(d\theta^2 + \sin\theta d\phi^2)] = a^2(\tau)[d\tau + \gamma_{i,j}(x^k)dx^i dx^j] . \quad (3.10)$$

Here, $\gamma_{i,j}$ corresponds to the spatial part of the metric, and $S_K(\chi)$ is defined as:

$$S_K(\chi) = \begin{cases} \frac{1}{\sqrt{K}} \sin(\sqrt{|K|}\chi) & \text{if } K > 0 \\ \chi & \text{if } K = 0 \\ \frac{1}{\sqrt{K}} \sinh(\sqrt{|K|}\chi) & \text{if } K < 0. \end{cases} \quad (3.11)$$

This reveals the significance of the constant K , which takes on three values: (i) $K > 0$ corresponds to a closed 3D sphere with positive curvature K . (ii) $K = 0$ corresponds to a flat 3D space. (iii) $K < 0$ corresponds to an open 3D hyperbolic space with negative curvature K .

The concept of conformal time emerges as the duration required for light to traverse from an observer to the farthest distance within the expanding universe. Its definition is presented as [29]:

$$dt = a(\tau)d\tau . \quad (3.12)$$

With this, the FLRW metric can be reformulated as:

$$ds^2 = a^2(t) \left[d\tau - \left(\frac{dr^2}{1 - Kr^2} + r^2(d\theta^2 + \sin^2\theta d\phi^2) \right) \right] . \quad (3.13)$$

3.1.4 Friedmann Equations

In equation (3.8), we explored the underlying geometric assumption. Subsequently, we introduce another foundational premise known as the "perfect fluid" assumption, which posits a homogeneous distribution of matter and energy across the universe, a concept that holds true on cosmologically significant scales [89]. Under this assumption, the energy-momentum tensor takes the form:

$$T_{\mu\nu} = (\rho + p)u_\mu u_\nu + pg_{\mu\nu} , \quad (3.14)$$

Here, p denotes pressure, ρ signifies energy density, and u_μ represents the four-velocity of the corresponding fluid.

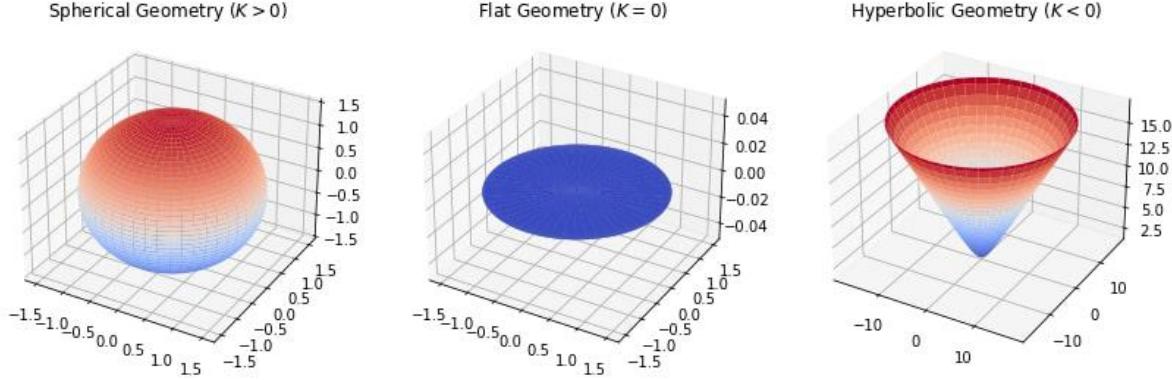


Figure 3.1: Illustration depicting the possible shapes of the universe as determined by the FLRW metric, with K representing the curvature parameter. Contemporary observations strongly favor a flat universe [28, 97].

Leveraging this energy-momentum tensor for the left-hand side of equation (3.6), and employing the Einstein tensor (3.4) derived from the FLRW metric (3.8) for the right-hand side, we deduce two ordinary differential equations from the non-vanishing components:

$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G}{3}\rho - \frac{K}{a^2} \quad (3.15)$$

$$\frac{\ddot{a}}{a} = -\frac{4\pi G}{3}(\rho + 3p) , \quad (3.16)$$

In this context, \dot{a} and \ddot{a} represent the first and second-time derivatives of the scale factor a .

Collectively, these differential equations are recognized as the Friedmann equations⁴, where (3.15) corresponds to the first Friedmann equation and (3.16) is the second Friedmann equation. These equations fundamentally elucidate how the universe's expansion (or contraction) is influenced by the energy content within it.

Before progressing to the subsequent subsection, we derive an additional equation by substituting the time derivative of (3.15) into (3.16), yielding:

$$\dot{\rho} + 3\frac{\dot{a}}{a}(\rho + p) = 0 . \quad (3.17)$$

This equation, known as the energy conservation equation⁵, establishes a pivotal link between the temporal evolution of the universe and the energy density of specific cosmic constituents.

⁴Pedagogically, it is possible to derive the Friedmann equation within a Newtonian framework, but the origin of K is never explicitly addressed.

⁵This equation is also known as 'third' Friedmann equation (V. Mukhanov, Lecture on Cosmology, 2022, Munich).

3.2 Homogeneous Universe

In the preceding section, we derived the cornerstone equations governing the behavior of a homogeneous universe—namely, the Friedmann equations. In this ensuing section, we embark on a comprehensive exploration of how the universe's expansion history is intricately influenced by its constituent energy components. Furthermore, we delve into the key observational quantities that serve as essential metrics in the realm of cosmological measurements.

3.2.1 The Expansion History of the Universe

We now delve into the unfolding expansion of our universe which is a phenomenon that intricately determines both the history and fate of our cosmos.

We begin by assuming the universe is composed of barotropic fluids, wherein the pressure is proportional to the energy density. This assumption gives rise to the equation of state:

$$p = w\rho , \quad (3.18)$$

Here, w is recognized as the equation of state parameter, which serves as a pivotal quantity in the cosmological context. The barotropic fluid's behavior is further characterized by a constant known as the speed of sound, denoted as c_s . This speed of sound can be expressed in terms of the equation of state parameter:

$$c_s^2 = \frac{dp}{d\rho} . \quad (3.19)$$

From this equation, it's evident that the sound speed c_s is equivalent to the square root of the equation of state parameter w . Utilizing the conservation of energy equation (3.17), we derive the time evolution of energy density:

$$\rho = \rho_0 a^{-3(w+1)} , \quad (3.20)$$

where ρ_0 signifies the energy density of the fluid in the present universe.

In reality, of course, our universe isn't comprised of a singular fluid. Therefore, we extend our analysis to encompass N distinct components, each characterized by an equation of state parameter w_i . Assuming these fluid components do not interact with each other, we can straightforwardly extend (3.20) to:

$$\rho = \sum_{i=1}^N \frac{\rho_{i,0}}{a^{3(1+w_i)}} . \quad (3.21)$$

Here, $\rho_{i,0}$ denotes the energy density of each fluid component in the present-day universe.

Introducing a fundamental quantity in cosmology, the Hubble parameter H is the ratio of the rate of expansion \dot{a} to the scale factor a of the universe

$$H \equiv \frac{\dot{a}}{a} . \quad (3.22)$$

With the aid of this Hubble parameter and equation (3.21), we reframe the first Friedmann equation:

$$H = \frac{8\pi G}{3} \sum_{i=1}^N \frac{\rho_{i,0}}{a^{3(1+w)}} - \frac{K}{a^2} . \quad (3.23)$$

Evaluating this equation for the present-day universe ($a_0 = 1$), we arrive at:

$$H_0^2 = \frac{8\pi G}{3} \sum_{i=1}^N \rho_{i,0} - K , \quad (3.24)$$

Where H_0 represents the Hubble constant, that is the current expansion rate of the universe with dimensions of $\text{kms}^{-1}\text{Mpc}^{-1}$ ¹⁶. In cosmology, the Hubble parameter is often expressed dimensionless, denoted as the parameter h [22]:

$$h = \frac{H_0}{100\text{kms}^{-1}\text{Mpc}^{-1}} . \quad (3.25)$$

Another crucial cosmological parameter is the critical density, denoted as ρ_c :

$$\rho_c = \frac{3H_0^2}{8\pi G} . \quad (3.26)$$

The fractional density parameter, Ω_i , can also be defined using this critical density:

$$\Omega_i = \frac{\rho_{i,0}}{\rho_c} . \quad (3.27)$$

With these quantities, we can restate the first Friedmann equation in terms of ρ_c and Ω_i :

$$K = H_0^2 \left(\frac{1}{\rho_c} \sum_{i=1}^N \rho_{i,0} - 1 \right) = H_0^2 \left(\sum_{i=1}^N \Omega_i - 1 \right) . \quad (3.28)$$

From the first equality, we discern that ρ_c represents the sum of all $\rho_{0,i}$ components and is zero in a flat universe ($K = 0$). The second equality underscores that the sum of all Ω_i must equal 1 in a flat universe.

In cosmology, you'll frequently encounter the dimensionless physical density parameter, which is a scaled version of the fractional density parameter. It is denoted as:

$$\omega_i = \Omega_i h^2 . \quad (3.29)$$

¹⁶The original measurement of H_0 by Hubble was $500\text{kms}^{-1}\text{Mpc}^{-1}$ [63].

3.2.2 The Energy Components of the Λ CDM Universe

In the previous subsection, we derived the equations that show the relation between the energy of our universe and its destiny. Now, we delve into how various energy components dictate the course of the universe's expansion.

Utilizing Equation (3.20) within the framework of (3.17), we formulate a differential equation with respect to the scale factor $a(t)$, assuming a flat universe comprised of a single fluid component. The solution to this time-dependent ordinary differential equation takes the form:

$$a(t) \propto t^{\frac{2}{3(1+w)}} . \quad (3.30)$$

This expression illuminates the profound interrelation between the dominant components of the universe and its expansion as a function of time. Let us now scrutinize the various energy components embraced within the Λ CDM cosmology.

Matter components Within the Λ CDM framework, the fractional energy density of matter, Ω_m , can be expressed as the sum of fractional densities attributed to cold dark matter (Ω_{CDM}) and baryonic matter (Ω_b)⁷:

$$\Omega_m = \Omega_{\text{CDM}} + \Omega_b . \quad (3.31)$$

In the cosmological context, the term "baryonic matter"⁸ pertains to particles that engage in photon interactions, predominantly composed of hydrogen and helium produced during Big Bang nucleosynthesis⁹. In contrast, dark matter solely interacts gravitationally. Contributions from neutrinos are disregarded for the purpose of this discussion. For a flat universe in a matter-dominated epoch ($\Omega_m = 1$ and $K = 0$), the evolution of the scale factor follows $a \propto t^{\frac{2}{3}}$, characterizing the so-called Einstein-de Sitter universe [89]. The Hubble parameter in this regime reads:

$$H(t) = \frac{2}{3t} . \quad (3.32)$$

Radiation components The radiation component comprises two primary constituents: photons and relativistic neutrinos. Accordingly, the fractional density of the radiation component Ω_r can be expressed as the sum of fractional densities attributed to neutrinos Ω_ν and photons Ω_γ :

$$\Omega_r = \Omega_\nu + \Omega_\gamma . \quad (3.33)$$

⁷Sometimes non-relativistic neutrinos are also considered as part of matter component.

⁸In the definition from Particle physics [84], Baryon is the particle which is composed of 3 quarks.

⁹The observed abundance of Deuterium is strong evidence in support of the Big Bang model. See detail in [20].

While the contribution from neutrinos is nearly negligible¹⁰, radiation also makes a minor contribution to the present-day energy composition. During the early stages of the universe, radiation dominated, influencing the transition to the subsequent matter-dominated era. In the context of a flat radiation-dominated universe, the evolution of the scale factor follows $a \propto t^{\frac{1}{2}}$ [89], yielding the corresponding Hubble parameter[89]:

$$H(t) = \frac{1}{2t} . \quad (3.34)$$

Cosmological Constant and Acceleration Observations of luminosity distance of supernovae [95] have unveiled the phenomenon of accelerated expansion. This acceleration necessitates the equation of state parameter $w_{\text{DE}} < -\frac{1}{3}$ for dark energy¹¹. In the Λ CDM framework, dark energy is represented by the cosmological constant. The Einstein equation (3.6) gives rise to the Friedmann equation with the cosmological constant (Λ):

$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G}{3}\rho - \frac{K}{a^2} + \frac{\Lambda}{3} \quad (3.35)$$

$$\frac{\ddot{a}}{a} = -\frac{4\pi G}{3}(\rho + 3p) + \frac{\Lambda}{3} . \quad (3.36)$$

Given the constancy of the cosmological constant, it follows that $w_\lambda = -1$. Considering a flat universe solely dominated by the cosmological constant ($\Omega_\Lambda = 1$, $K = 0$), solving the first Friedmann equation (3.35) yields:

$$a(t) \propto e^{H_0 t} . \quad (3.37)$$

It is evident from (3.37) that the Hubble parameter $H(t)$ remains constant in this scenario, and the universe experiences exponential expansion when the condition $\Lambda > 0$ holds true. This model, known as the "De Sitter universe," often serves as a representative early universe model.

Turning our attention beyond individual components, the overall energy composition of the universe encompasses the cosmological constant (Ω_Λ), matter ($\Omega_m = \Omega_b + \Omega_{\text{CDM}}$), and curvature ($\Omega_K = -\frac{K}{H_0^2}$). The Hubble parameter can be derived from the first Friedmann equation (3.35):

$$H(a) = H_0 \sqrt{\Omega_r a^{-4} + \Omega_m a^{-3} + \Omega_K a^{-2} + \Omega_\Lambda} . \quad (3.38)$$

Integrating (3.38) yields the age of the universe t_0 :

$$t_0 = \frac{1}{H_0} \int_0^1 \frac{1}{H_0 \sqrt{\Omega_r a^{-4} + \Omega_m a^{-3} + \Omega_K a^{-2} + \Omega_\Lambda}} da . \quad (3.39)$$

Observations of cosmic microwave anisotropy [97] and large-scale structure distribution [28] offer constraints on the geometry of our universe. These findings suggest a nearly flat geometry, thus for the purpose of this thesis, we consider a flat universe ($\Omega_K = 0$).

¹⁰However, this component is pivotal to include for precise calculation of the moment of matter-radiation equality.

¹¹This is known as the strong energy condition.

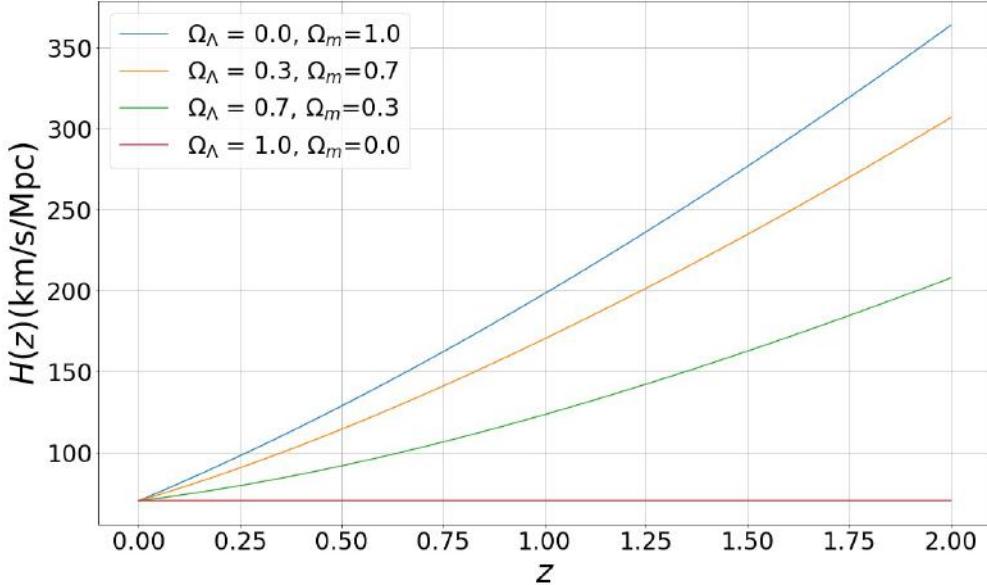


Figure 3.2: This figure depicts the Hubble parameter calculated using Equation (3.38) within a flat universe. Notably, the diverse energy components of the universe exert their influence on the expansion history.

3.2.3 Observational Quantities in Cosmology

In this subsection, we introduce fundamental observational quantities that characterize the expansion history of our universe.

Cosmological Redshift As light from distant astronomical objects, such as galaxies or quasars, traverses the expanding universe, its wavelength is stretched, resulting in a redshift in the electromagnetic spectrum. The cosmological redshift¹² z quantifies this effect and is defined as

$$z \equiv \frac{\lambda_o - \lambda_e}{\lambda_e} , \quad (3.40)$$

where λ_o is the observed wavelength at time t_0 , and λ_e is the emitted wavelength at an earlier time $t_e < t_0$. The cosmological redshift captures the change in wavelength due to the universe's expansion. By considering an isotropic universe and choosing the coordinates such that $d\phi = d\theta = 0$, we can utilize the FLRW metric (3.8) to establish the relation

$$(1 + z) = \frac{1}{a(t)} . \quad (3.41)$$

¹²In addition to cosmological redshift, there are other types of redshift. Doppler redshift, for instance, occurs when an object is moving away from an observer, causing the spectral lines of its emitted or reflected light to shift toward longer wavelengths. Similarly, gravitational redshift, in the presence of a strong gravitational field, as predicted by Einstein's theory of General Relativity, light can be redshifted as it climbs out of the gravitational well.

This equation succinctly connects the redshift to the scale factor of the universe.

Comoving Distance To model the relationship between redshift and distance, the concept of proper distance is introduced, which is then refined into the comoving distance. The comoving distance D_c represents the spatial separation between astronomical objects, irrespective of the scale factor, and is defined as

$$D_c = \frac{D_p}{a(t)} = \int_0^r \frac{dr'}{\sqrt{1 - Kr'^2}} = \int_0^z \frac{dz}{H(z)} . \quad (3.42)$$

Angular Diameter Distance The angular diameter distance D_A characterizes the angular separation between two objects, as measured from a fixed observer. In the FLRW metric's comoving coordinates, considering a proper distance δl associated with an angular separation $\delta\phi$, we have

$$\delta l = a(t)S_K(\chi)\delta\phi . \quad (3.43)$$

This leads to the definition of the angular diameter distance:

$$D_A(z) = \frac{\delta l}{\delta\phi} = a(t)S_K(\chi(z)) . \quad (3.44)$$

The angular diameter distance is instrumental for deducing the distance-redshift relationship from standard rulers, such as baryonic acoustic oscillations (BAO).

Luminosity Distance The luminosity distance D_L is a crucial observational quantity, particularly in the context of standard candles like Type Ia supernovae. For an object with known intrinsic luminosity or brightness, the luminosity distance is defined using the bolometric flux S and the observed bolometric luminosity L :

$$D_L = \sqrt{\frac{L}{4\pi S}} . \quad (3.45)$$

By employing Etherington's reciprocity theorem, the luminosity distance can be expressed in terms of the angular diameter distance $D_A(z)$:

$$d_L(z) = (1 + z)^2 D_A . \quad (3.46)$$

The measurement of luminosity distances, such as those of Type Ia supernovae, played a pivotal role in establishing the universe's accelerated expansion and the dominance of dark energy [95].

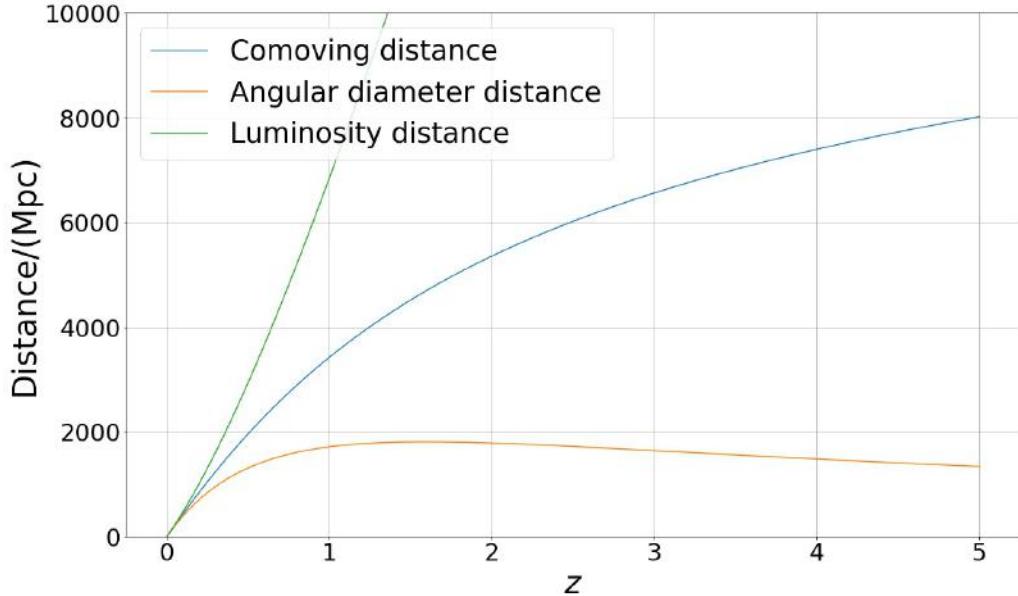


Figure 3.3: Comparison of various distances in cosmology computed using equations (3.42), (3.44), and (3.46). The calculations assume a flat Λ CDM model with cosmological parameters $\Omega_m = 0.14$, $\Omega_\Lambda = 0.31$, and $\Omega_\gamma = 4.12 \times 10^{-5}$.

3.3 The Origin of Density Fluctuation

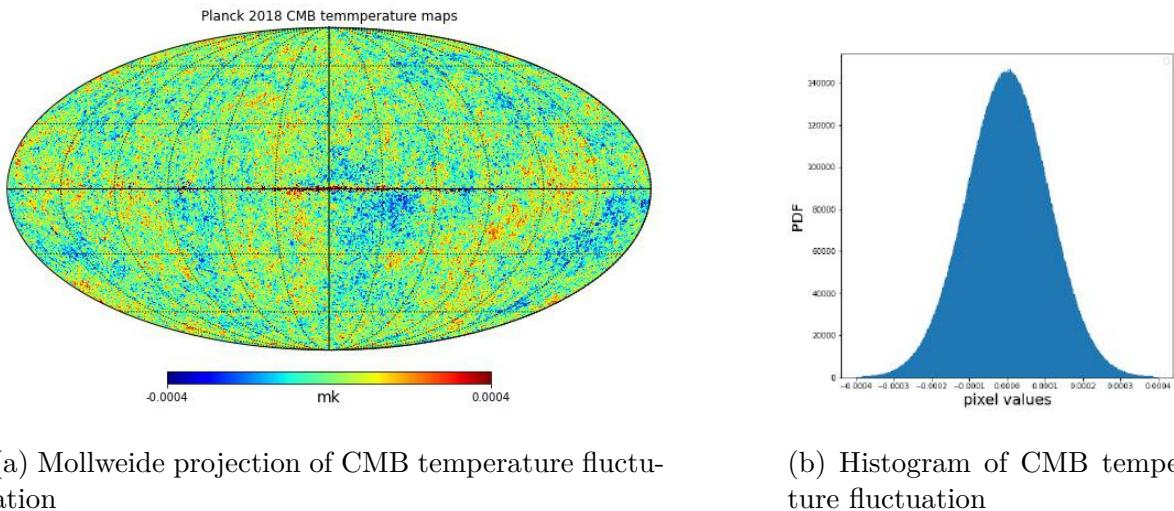
So far, we have discussed the concept of a homogeneous universe based on the cosmological principle. However, our actual universe exhibits inhomogeneities and structures on various scales, such as cosmic voids and filaments, which arise due to the gravitational evolution of initial density fluctuations. These fluctuations are the seeds of cosmic structure and can be understood by analyzing the temperature anisotropies in the cosmic microwave background (CMB). The theoretical framework describing the initial density field and the dynamics of the very early universe is provided by inflation theory. The motivation behind inflation theory stems from addressing several issues not resolved by the Big Bang Theory, namely the Flatness problem, Horizon problem, Magnetic monopole problem, and the origin of fluctuations [104]. For a detailed discussion of each problem, refer to literature such as [89], [10], and [94].

One possible explanation for the origin of density inhomogeneities was proposed by V. Mukhanov and G. Chibisov in 1981, suggesting that these initial perturbations originated from quantum fluctuations [90]. Heisenberg's uncertainty principle leads to vacuum-generated quantum fluctuations, which then grow beyond the Hubble horizon to macroscopic scales. If these fluctuations originated from a single energy component during inflation, they are adiabatic in nature. Current measurements of the cosmic microwave background support this adiabatic fluctuation [98], aligning with the concept of single-field inflation¹³, where the perturbation originates from a scalar field known as the inflaton

¹³The single scalar field inflation generates not only scalar perturbations but also tensor perturbations.

[80].

A key prediction of single-field inflation is that the density field follows a Gaussian probability distribution, which agrees remarkably well with measurements of CMB temperature anisotropies [97]. Another relevant prediction is the spectral index, denoted as n_s , characterizing the slope of the linear matter power spectrum with respect to the wave number k , and a scale-invariant spectrum corresponds to $n_s = 1^{14}$. Single-field inflation predicts a slight deviation from the scale-invariant spectrum [89], and the observed scalar spectral index also indicates a small deviation from this spectrum [97].



(a) Mollweide projection of CMB temperature fluctuation

(b) Histogram of CMB temperature fluctuation

Figure 3.4: Commander temperature map from Planck 2018 data [97]. As we see in the histogram on the right-hand side, the CMB temperature fluctuation is extremely close to Gaussian. Therefore, the assumption of Gaussianity of initial density fluctuation is a valid assumption. However, the observed CMB fluctuation has tiny non-Gaussianity because of the secondary anisotropy such as CMB lensing. The single inflation model produces also tensor perturbation, and this perturbation generates the primordial gravitational wave. This signal appears in the B-mode polarization of CMB. Detecting this signal is one of the biggest challenges of modern cosmology in the next decades.

3.4 Linear Structure Formation

In the previous chapter, we explored the origin of cosmic structures arising from initial density fluctuations. These fluctuations, initially almost Gaussian in nature, evolve over time under the influence of gravity, leading to the formation and growth of large-scale

These tensor perturbations give rise to primordial gravitational waves, which have the potential to induce non-Gaussianity in the density field of the early universe. However, as of now, there has been no detection of non-Gaussianity originating from gravitational waves [23].

¹⁴This scale-invariant power spectrum is called Harrison-Zeldovich spectrum [10]

structures. During the early stages of structure formation, these fluctuations are relatively small, allowing us to employ the linear approximation $\delta \ll 1$ to describe the evolution of the density contrast field. This section introduces the linear description of the density field and the theoretical model of the power spectrum, connecting theoretical predictions to observable large-scale structure cosmology.

3.4.1 Linear Perturbation Theory

The scalar perturbations generated by single-field inflation introduce small distortions to the Friedmann-Lemaître-Robertson-Walker (FLRW) metric due to gravitational effects. This can be expressed as follows [39]:

$$ds^2 = a^2(\tau) \left[(1 + 2\Phi)d\tau^2 + (1 - 2\Psi)\gamma_{ij}dx^i dx^j \right], \quad (3.47)$$

where Φ and Ψ are the gauge-invariant Bardeen potentials. By considering perturbations to the Einstein tensor and energy-momentum tensor, we derive the linearized Einstein equations¹⁵ [39]¹⁶. Specifically, the relativistic Poisson equation¹⁷¹⁸ takes the form:

$$\nabla^2\Phi - 3\mathcal{H}(\Phi' + \mathcal{H}\Phi) = 4\pi G a^2 \delta\rho, \quad (3.48)$$

where $\mathcal{H} = aH$ is the conformal Hubble parameter. Furthermore, we obtain the relativistic continuity and Euler equations for the evolving perturbations:

$$\delta' + 3\mathcal{H}(c_s^2 - w)\delta + (1 + w)(\nabla_i v^i - 3\Phi') = 0, \quad (3.49)$$

$$v'_i + \mathcal{H}(1 - 3c_s^2)v_i + \frac{c_s^2}{1 + w}\nabla_i\delta + \nabla_i\Phi = 0. \quad (3.50)$$

These relativistic hydro-equations govern the evolution of perturbations during different eras of the universe¹⁹.

3.4.2 Matter Power Spectrum

The matter power spectrum is a fundamental statistical tool that quantifies the abundance of large-scale structures in the universe. By solving the linearized Einstein and hydrodynamical equations for sub-horizon scales, we can derive the linear matter power spectrum $P(k, a)$. This power spectrum captures the amplitude and shape of the matter density fluctuations, with its amplitude influenced by the growth of perturbations.

¹⁵This is derived in the conformal Newtonian gauge(longitudinal gauge).

¹⁶In Appendix A of [39], there is a extremely detailed derivation.

¹⁷Here, the assumption $\Phi = \Psi$ is used, which is valid during inflation; however, it becomes inaccurate during radiation domination due to the presence of anisotropic stress arising from radiation pressure.

¹⁸This equation converges to the Poisson equation in the subhorizon scale ($k \ll \mathcal{H}$).

¹⁹More detailed discussion in [88].

The linear growth factor $D_l(a)$ ²⁰ describes the evolution of matter density perturbations. In a Λ CDM cosmology, the linear matter power spectrum is given by:

$$P(k, a) = A_s \left(\frac{D_l(a)}{D_l(a_0)} \right)^2 \left(\frac{k}{k_0} \right)^{n_s} T^2(k), \quad (3.51)$$

where A_s is the amplitude of the power spectrum, $D_l(a_0)$ is the linear growth factor at present time, and k_0 is the pivot wavenumber.

The shape of the power spectrum is influenced by the matter-radiation equality, characterized by the transfer function $T(k)$ ²¹. The transfer function accounts for the change in the curvature perturbations as modes enter the horizon and is responsible for the features of the power spectrum.

The linear power spectrum can be computed using Boltzmann solver like **CLASS** [76] or **CAMB** [77], taking into account various physical effects. The shape and amplitude of the power spectrum provide crucial information about the universe's composition and evolution.

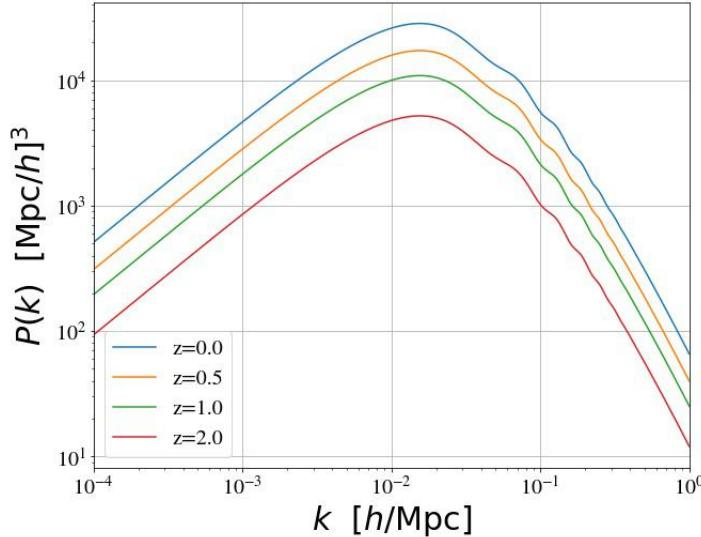


Figure 3.5: The linear matter power spectrum in flat Λ CDM with cosmological parameters from WMAP 9-year results [61], produced by **CLASS**. The amplitude of the power spectrum decreases with increasing redshift z .

3.4.3 The Power Spectrum and σ_8

The linear power spectrum is not only characterized by its shape and amplitude but also by a crucial parameter known as σ_8 [61]. This parameter quantifies the root-mean-square

²⁰Often, due to the rapid decay of the decaying mode, it is ignored in many analyses.

²¹The transfer function for Λ CDM model is derived by Eisenstein and Hu [33]

(RMS) fluctuation of the density field in spheres with a comoving radius of $8h^{-1}$ Mpc. It serves as a key indicator of the amplitude of density fluctuations on the scale of galaxy clusters [93].

The mass variance $\sigma^2(R)$, representing the variance of the density contrast within a sphere of radius R , is related to the linear power spectrum $P_L(k)$ through integration [33]. The parameter σ_8 is defined as the value of σ when $R = 8h^{-1}$ Mpc [31]. Mathematically, we have:

$$\sigma^2(R) = \frac{1}{2\pi^2} \int dk k^2 P_L(k) \tilde{W}^2(kR), \quad (3.52)$$

where $\tilde{W}(y)$ is the Fourier transform of the top-hat window function. The value of σ_8 encapsulates the information about the large-scale structure and the normalization of the power spectrum.

In a Λ CDM cosmology, the linear power spectrum can be written in terms of σ_8 [97]:

$$P_L(k) = P(k, a_0) = A_s \left(\frac{k}{k_0} \right)^{n_s} T^2(k), \quad (3.53)$$

where $P(k, a_0)$ is the power spectrum at present time, and A_s is determined by σ_8 :

$$A_s = \frac{2\pi^2}{k_0^{n_s+3}} \sigma_8^2. \quad (3.54)$$

Thus, σ_8 provides a direct link between the amplitude of density fluctuations on the scale of galaxy clusters and the parameters characterizing the linear power spectrum.

3.5 Non-Linear Structure Formation

In this section, I provided a concise overview of linear perturbation theory, a fundamental framework for understanding the initial growth of cosmological structures. However, it's important to note that the linear assumption, where the density fluctuations are small ($\delta \ll 1$), holds only on relatively large scales. On smaller scales, such as those encompassing galaxy clusters or individual galaxies, this assumption breaks down as the gravitational interactions lead to significant density fluctuations. This departure from linearity has a profound impact on structure formation, giving rise to processes like gravitational collapse and the formation of dark matter halos.

In addressing the non-linear aspects of structure formation, various theoretical approaches have been developed. One common strategy involves modifying the power spectrum through non-linear perturbation theory. This theory can be pursued in both Eulerian or Lagrangian formulations²², which respectively track the evolution of perturbations at fixed points in space or by following the trajectories of fluid elements. By considering the coupling terms of Fourier modes, corrections to the power spectrum can be introduced.

²²This is beyond the focus of this master thesis, however the good review for this topic is [11].

However, it's crucial to highlight that these corrections remain valid only within the quasi-linear regime, where the density contrast is approximately equal to 1. Beyond this regime, more sophisticated techniques are needed to accurately model non-linear structure formation.

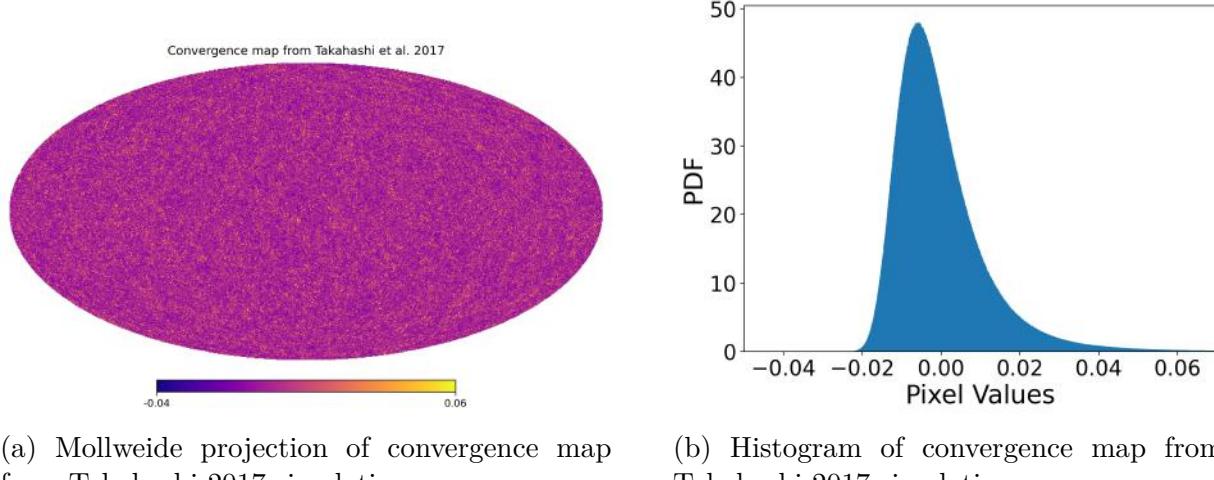


Figure 3.6: The density field projection is obtained from the Takahashi 2017 simulation [114] at a resolution of $NSIDE = 2048$. In the depicted figures, the left panel displays a histogram of the density field. Notably, the Gaussianity in the density fluctuations of the present-day universe is not preserved.

3.5.1 N-body Simulation

Accurate predictions of density fluctuations are achieved through numerical simulations known as N-body simulations. In these simulations, the density field is represented as particles, each possessing specific attributes—position and velocity. Commencing with initial conditions—often inferred from observed density fluctuations in the early universe, such as those gleaned from cosmic microwave background radiation measurements—these simulations compute the aforementioned quantities, continuously updating them based on gravitational interactions. The output typically furnishes data on redshifts, positions, and velocities. By employing this numerical technique, an array of complex large-scale structures stemming from non-linear structure formation, including cosmic webs, filaments, and voids, can be meticulously investigated. Comparing the power spectrum from cosmological N-body simulations to the linear power spectrum shows small differences on smaller scales

In this Master's thesis, `Halofit` [12] is employed to transition from the linear to the non-linear power spectrum. The underlying premise is that the non-linear power spectrum can be expressed as a summation of the linear power spectrum and corrective terms. Within the `Halofit` model, these corrective terms encapsulate the repercussions of non-linear gravitational evolution on smaller scales, encompassing phenomena like non-linear

gravitational collapse and the emergence of dark matter halos. This study incorporates cosmological models that integrate the influence of these non-linear effects on the power spectrum. By fitting these theoretical model functions using N-body simulations, a comprehensive alignment between theory and observation is established.

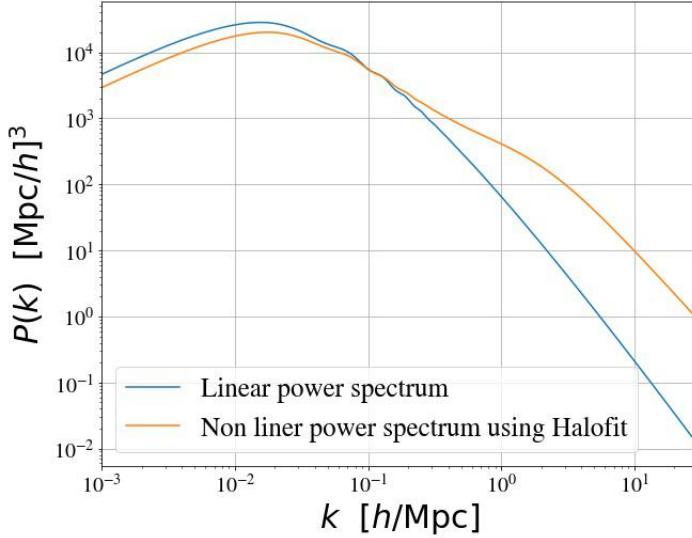


Figure 3.7: The linear matter power spectrum and non-linear power spectrum in the flat Λ CDM model with cosmological parameters obtained from the WMAP 9-year data [61]. These spectra were generated using the CLASS code at a redshift of $z = 0$. The non-linear power spectrum was computed using the **Halofit** model. Notably, the discrepancy between the two spectra is more pronounced on smaller scales, highlighting the substantial impact of non-linear gravitational interactions.

3.5.2 The Statistics beyond 2-point Functions

In exploring the theoretical aspects, it becomes apparent that the emergence of non-Gaussian features in the cosmic density field due to non-linear structure formation [11] underscores the necessity for extending statistical analyses beyond the realm of 2-point statistics. While the conventional approach to data analysis in large-scale structure cosmology relies heavily on 2-point statistics, this practice inadvertently disregards valuable information embedded in the non-Gaussian tail of the density field at smaller scales. Recognizing this, various avenues of investigation have emerged to transcend the constraints of 2-point statistics: (i) exploration of the 3-point correlation function [52], [51], [45], [53] (ii) formulation of models for the probability density function (PDF) of mass or convergence [42], [8], [14], (iii) utilization of density split statistics [49], [41], and (iv) adoption of field-based inference techniques [37], [36], [38], [13], [99].

Within the scope of this master's thesis, a distinctive approach known as field-level inference is employed. As previously elaborated in Chapter 2, the lognormal field stands

as an effective model to capture the characteristics of convergence maps. Building upon this insight, a simulation pipeline has been developed using the `Flask` framework [124], facilitating the generation of lognormal maps tailored to specific cosmologies. In contrast to the conventional abstraction of 2-point functions, this thesis explores graph convolutional neural networks for direct estimation of cosmological parameters without utilizing any summary statistics. This approach aims to extract cosmological information from the lognormal maps, thereby enhancing the precision of parameter estimation for test datasets.

3.6 Summary of Λ CDM Cosmology

This section provides a concise overview of the Λ CDM cosmology, which forms the backdrop of the regression problem concerning cosmological parameters within this master's thesis. The section begins by presenting the key Λ CDM parameters that are typically derived from diverse cosmological measurements. Subsequently, attention is drawn to some inherent challenges within the Λ CDM cosmological framework.

3.6.1 The Λ CDM Parameters

The foundational step in comparing theoretical predictions with observational data is parameterizing the underlying theory. However, observed data is influenced not only by cosmology but also by various astronomical and instrumental effects. Thus, the parameters in cosmological analyses can be categorized into cosmological parameters that encapsulate fundamental cosmological properties, astrophysical parameters that quantify astronomical effects, and instrumental parameters that manage measurement systematics.

Within the context of Λ CDM cosmology, the focus is on six cosmological parameters:

- Ω_m : Representing the energy density of cold dark matter with an energy state of $w_m = 0$. Occasionally, ω_m is employed instead of Ω_m due to h uncertainty.
- Ω_b : Signifying the energy density of baryonic components. Similarly, ω_b can be used.
- H_0 : Denoting the Hubble parameter, which characterizes the current universe's expansion rate.
- n_s : Describing the scalar spectral index, which captures minor deviations from a perfectly scale-invariant power spectrum. Confining this parameter necessitates constraining theories related to the early universe.
- A_s : This parameter characterizes the amplitude of the primordial power spectrum and is often expressed as $\ln 10^{10} A_s$ due to its typically small values. An alternative parameterization for capturing the amplitude of matter fluctuations is σ_8 ²³. σ_8 represents the root mean square density contrast of matter fluctuations within spheres of

²³Sanchez (2020) [105] propose using σ_{12} instead of σ_8 in order to solve the tension.

radius $8h^{-1}$. Another choice for constraining the amplitude of matter fluctuations in the universe is S_8 , defined as $S_8 = \sqrt{\frac{\Omega_m}{0.3}}$. The rationale for favoring S_8 over σ_8 alone lies in its ability to alleviate degeneracies between various cosmological parameters.

- τ : Denoting the optical depth of photons to the recombination surface, offering insights into reionization processes, structure formation, and intergalactic medium properties.

Astrophysical parameters, influenced by astronomical effects, and instrumental parameters, controlling measurement systematics, also contribute to the comprehensive analysis:

- b : Representing galaxy bias, accounting for the disparity between the galaxy density contrast field and the matter density contrast field due to various feedback processes. Galaxy bias b^{24} is defined as the ratio of galaxy density fluctuation field to matter density contrast field.
- A_{IA} : Signifying the amplitude of intrinsic alignment, observed in gravitational weak lensing surveys, where the correlation between galaxy shape/orientation and the surrounding large-scale structure is utilized. Tidal fields during galaxy formation introduce additional correlations between shape and nearby density fields.

Additionally, instrument-related parameters exist. In this master's thesis, focus is placed on just two parameters— Ω_m and σ_8 —due to computational limitations. Other parameters are considered nuisance parameters, held constant for the simulation pipeline.

3.6.2 Challenges in Λ CDM Theory

Despite its status as the prevailing standard model of cosmology, Λ CDM theory faces challenges that cannot be fully reconciled with experimental data. These challenges signify the potential divergence between measurements from various cosmological tracers, offering a gateway to more profound insights into our observations and theories.

Two prominent tensions within Λ CDM theory are:

- Hubble Tension: Discrepancies in the Hubble parameter H_0 exist across different measurements. Notably, H_0 values from nearby galaxies, such as supernovae or Cepheid variables, deviate by five sigma from the values derived from cosmic microwave background (CMB) and large-scale structure measurements [1]. Although lensing could potentially address H_0 constraints, challenges remain.
- S_8 Tension: Disagreement in the S_8 parameter, characterizing the amplitude of matter density fluctuations on large scales, persists between the late-time universe and the early universe. This tension is of significance, indicating possible discrepancies or systematic uncertainties [1].

²⁴This constant is from the linear order of the Kaiser bias model [69]

This thesis addresses $S8$ tensions by constructing cosmological contours in the $S8 - \Omega_m$ (In Appendix C) and $\sigma_8 - \Omega_m$ parameter space through a field-based approach on diverse test datasets. This methodology reveals substantial biases and potential inaccuracies stemming from specific approximations employed during analyses.

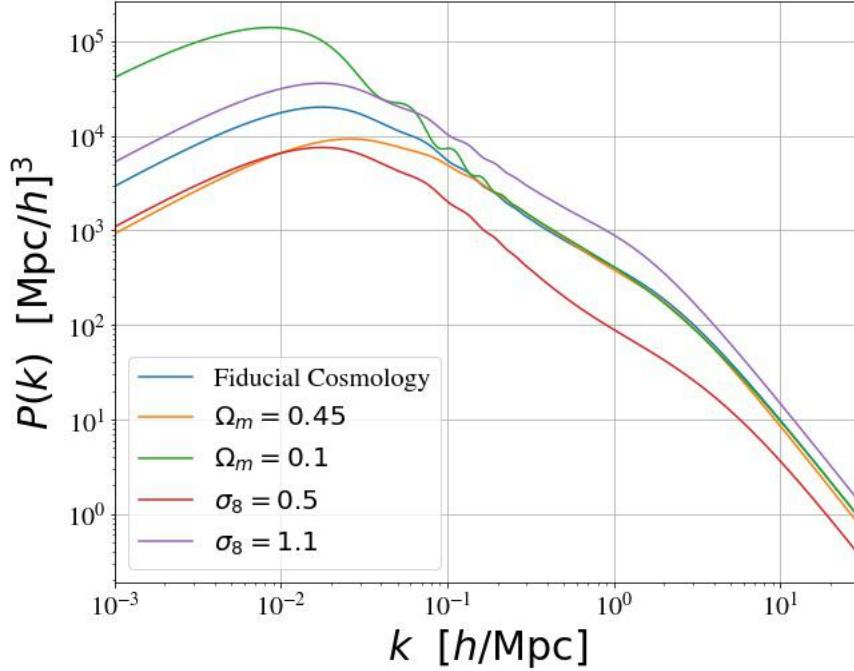


Figure 3.8: The non-linear matter power spectrum within the framework of flat Λ CDM cosmology, generated using **CLASS**. The fiducial parameter values are based on the cosmological parameters derived from the WMAP 9-year results [61]. Variations in cosmological parameters lead to changes in both the overall shape and amplitude of the matter power spectrum.

Chapter 4

Weak Gravitational Lensing

In the preceding sections, we delved into briefly the theoretical framework of how Λ CDM explains the density fluctuations in the universe. In this section, our focus shifts to the practical aspect of measuring these density fluctuations through observations. Specifically, my Master's thesis centers around the gravitational convergence lensing field, which manifests cosmology through the distortion of galaxy shapes—a phenomenon measurable through sophisticated shape measurement statistical techniques.

The principle underlying gravitational lensing is very simple: the distribution of matter alters the trajectory of photons as they traverse between a distant light source and an observer, often leading to magnified or multiple images of remote sources. Within the realm of weak lensing, the light from galaxies in the background is subtly distorted by the intervening large-scale structure—ubiquitous throughout our universe. This gravitational bending induces a systematic distortion in the shapes of these background galaxies, a signal that can be precisely measured and subjected to statistical analysis.

Exploiting this lensing phenomenon allows us to construct maps of the matter distribution responsible for the observed galaxy shape distortions. This distribution of matter can be mathematically described using statistical models rooted in cosmological theory. As a result, we gain the capacity to constrain cosmological parameters or ascertain the credibility of various cosmological models. This determination is approached from a Bayesian perspective, whereby we evaluate the likelihood of underlying data derived from shape measurements against different cosmological scenarios.

Within this section, I elucidate the fundamental concepts of gravitational weak lensing, presenting essential formulas and theoretical constructs.

This section is based on the reviews and the article [6], [106], [71], [39], [7].

4.1 Gravitational Lensing

Before we delve into the basics of weak lensing, it's imperative to grasp the overarching concept of gravitational lensing itself. This phenomenon arises from the distortion induced in the path of photons as they traverse space under the influence of gravity, ultimately

reaching an observer. To shed light on this concept, let's scrutinize the phenomenon of light bending due to gravity.

For illustrative purposes, we shall first consider the simplest scenario: the bending of light caused by a point mass. In this scenario, we can aptly describe the spacetime trajectory of a photon using the Schwarzschild metric. This metric's line element, in spherical coordinates, takes the form [39]:

$$ds^2 = \left(1 - \frac{2Gm}{r}\right) dt^2 - \frac{dr^2}{1 - \frac{2Gm}{r}} + r^2(d\theta^2 + \sin^2\theta d\phi^2). \quad (4.1)$$

Here, m represents the mass of the point mass, and r denotes its location. Notably, two singularities emerge at $r = 0$ and $r = 2m$, with the latter being termed the Schwarzschild radius. This radius marks the event horizon, which is the point where photons cannot escape the gravitational pull. An interesting aspect of the Schwarzschild metric is its convergence to the Minkowski metric as $r \rightarrow \infty$.

Crucially, the Schwarzschild metric manifests another captivating property: it deviates from the Minkowski metric when not within a singularity. As a result, the paths of objects in space-time curve instead of following straight lines, causing the bending of photon paths when they interact with gravitational fields.

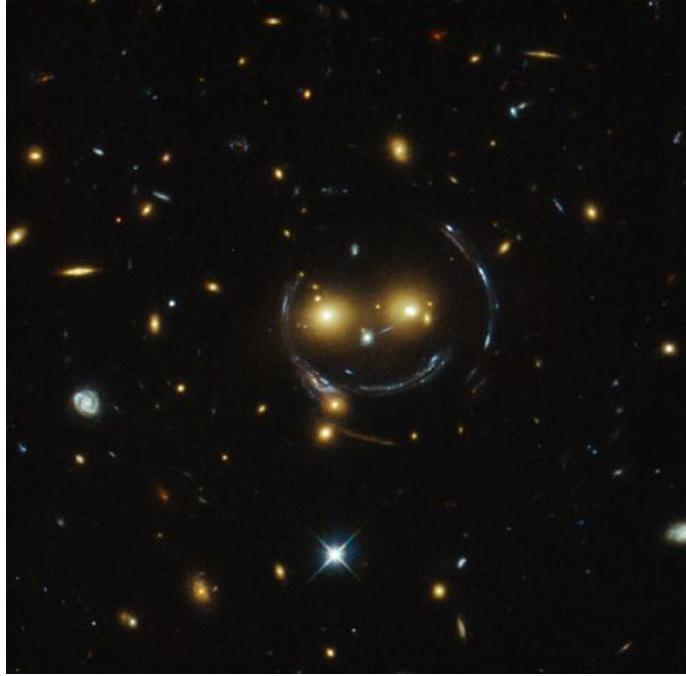


Figure 4.1: The Cheshire Cat image of galaxy cluster (SDSS J1038+4849) by Hubble Space Telescope. The Einstein ring is caused by the high symmetric distribution of background galaxies which causes the tangential distortion into the ring-like image. Image credit: Hubble Sees a Smiling Lens, NASA/ESA <https://www.nasa.gov/content/hubble-sees-a-smiling-lens>

4.1.1 The Lens Equation

This subsection explores the lens equation, a fundamental concept in gravitational lensing theory. The lens equation is derived here within the context of a lens system which is configured in figure 4.2, where distinct planes - the lens plane housing the gravitational lens, and the source plane containing the emitting source. These planes are separated by angular distances: the observer's distance to the lens plane at redshift z_d is D_d and from the lens plane to the source plane at the same redshift is D_{ds} . These angular diameter distances culminate in the observer-source angular distance, defined as D_s^1 .

This lensing scenario employs a two-dimensional coordinate system: ξ for the lens plane and η for the source plane. The angles θ , β , and $\hat{\alpha}$ are integral to this configuration [6]. Specifically, $\hat{\alpha}$ is termed the deflection angle, which is expressed as the integral over the transverse gradient of the gravitational potential Φ :

$$\hat{\alpha} = 2 \int_l \nabla_\perp \Phi dl . \quad (4.2)$$

The thin lens approximation becomes relevant when the extended object's size is significantly smaller than D_s and D_{ds} , rendering these angular distances minuscule compared to the Schwarzschild Radius r_s . This approximation construes the lensing event to be confined to the lens plane, leading to the thin lens deflection angle $\hat{\alpha}(\xi)$:

$$\hat{\alpha}(\xi) = 4 \int d^2\xi' \Sigma(\xi') \frac{\xi - \xi'}{|\xi - \xi'|^2} . \quad (4.3)$$

Here, the surface density $\Sigma(\xi)$ emerges from the line of sight integration of the three-dimensional density distribution ρ .

Employing the flat sky approximation² , the lens system can be derived as follows:

$$\beta = \theta - \frac{D_{ds}}{D_s} \hat{\alpha}(\theta) . \quad (4.4)$$

To simplify matters, we introduce the rescaled deflection angle α , yielding:

$$\beta = \theta - \alpha(\theta) , \quad (4.5)$$

This lens equation establishes a connection between the angular positions of a distant source, the gravitational deflection caused by a lensing mass, and the observer's viewpoint in gravitational lensing.

4.1.2 Shear and Convergence

This subsection delves into two pivotal observational quantities within gravitational lensing which are shear and convergence. We embark on establishing the convergence field κ , a

¹Since the angular diameter distance is not additive quantity.

²All involved angles are small enough to approximate the angle in the 2D plane.

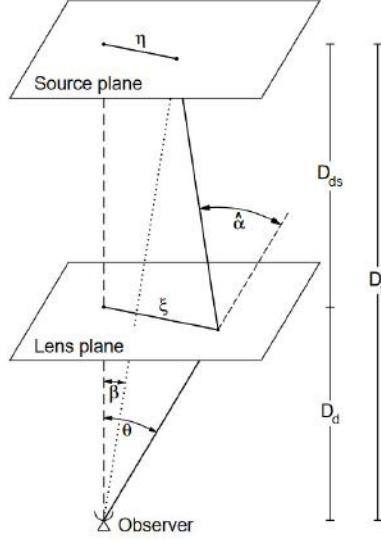


Figure 4.2: The sketch of the lens system from [6]

dimensionless measure of surface mass density [106]:

$$\kappa(\boldsymbol{\theta}) = \frac{\Sigma(\boldsymbol{\theta})}{\Sigma_{\text{crit}}} . \quad (4.6)$$

Here, Σ_{crit} denotes the critical surface density which is defined as follows,

$$\Sigma_{\text{crit}} = \frac{1}{4G\pi} \frac{D_s}{D_{ds} D_d} . \quad (4.7)$$

$\Sigma(\boldsymbol{\theta}) > \Sigma_{\text{crit}}$ is sufficient condition for the multiple solution of lens equation (4.5). In this case, the multiple images of the background galaxy can be observed and called strong lensing.

The relationship between the rescaled deflection angle α and the convergence field κ is articulated, along with the deflection potential ϕ :

$$\alpha(\boldsymbol{\theta}) = \nabla \phi(\boldsymbol{\theta}), \quad \phi(\boldsymbol{\theta}) = \frac{1}{\pi} \int d^2\boldsymbol{\theta}' \kappa(\boldsymbol{\theta}') \ln(|\boldsymbol{\theta} - \boldsymbol{\theta}'|) . \quad (4.8)$$

Furthermore, the convergence κ is linked to the deflection potential ϕ through:

$$\kappa(\boldsymbol{\theta}) = \frac{1}{2} \nabla^2 \phi(\boldsymbol{\theta}) . \quad (4.9)$$

The framework transitions into the realm of extended light sources, culminating in the linearized lens mapping equation:

$$\boldsymbol{\beta}(\boldsymbol{\theta}_0) + \mathbf{A}(\boldsymbol{\theta}_0) \cdot (\boldsymbol{\theta} - \boldsymbol{\theta}_0) . \quad (4.10)$$

With \mathbf{A} as the Jacobian matrix containing components related to the convergence κ and shear components γ_1 and γ_2 as follows,

$$\mathbf{A} = \begin{pmatrix} 1 - \kappa - \gamma_1 & -\gamma_2 \\ -\gamma_2 & 1 - \kappa + \gamma_1 \end{pmatrix}. \quad (4.11)$$

where γ_1 and γ_2 are defined,

$$\begin{aligned} \gamma_1(\boldsymbol{\theta}) &= \frac{1}{2}(\partial_1 \partial_1 - \partial_2 \partial_2)\phi \\ \gamma_2(\boldsymbol{\theta}) &= \partial_1 \partial_2 \phi. \end{aligned} \quad (4.12)$$

Shear and convergence, essential aspects of gravitational lensing, intricately shape the appearance of distant galaxies. Shear, described by the complex $\gamma(\boldsymbol{\theta})$, is decomposed into tangential and cross components, altering galaxy shapes along and perpendicular to the lensing axis. Convergence, denoted by $\kappa(\boldsymbol{\theta})$, magnifies and distorts galaxies uniformly, while the interplay between shear and convergence, quantified by the reduced shear g^3 , determines the degree of elongation or compression observed in lensed galaxy images.

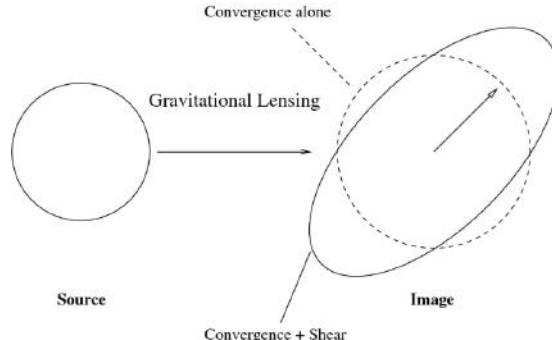


Figure 4.3: This illustration [118] depicts the mapping resulting from the gravitational lensing effect, transitioning from the source plane to the lensed plane. The convergence by itself induces isotropic magnification of a circular source, while the shear is responsible for generating varying sizes and shapes of the resulting ellipses.

4.2 Weak Lensing

In the preceding section, it was established that the condition $\Sigma > \Sigma_{\text{crit}}$ signifies strong lensing, yielding phenomena like multiple images. However, these striking manifestations, such as arcs or Einstein rings, are relatively rare in our universe. More commonly, most lines of sight fall within the regime of weak lensing which is $\Sigma < \Sigma_{\text{crit}}$. In this context, light deflection isn't discernible from a single source due to the underlying condition. Yet, the

³Reduced shear is defined as $g = \frac{\gamma}{1-\kappa}$. It is a normalized quantity with respect to an isotropic stretching of the image by convergence.

ubiquity of matter distribution still imparts a vital influence—gravitational lensing distorts the shape of galaxies through the presence of foreground matter. This distortion, observed across wide sky areas in weak lensing cosmology, presents an avenue to extract valuable insights into matter distribution, ultimately refining our grasp of cosmological theories by a stochastic model.

4.2.1 Weak Lensing Basics

The shapes of galaxies, serving as tracers of the foreground density field, can be estimated from the surface brightness, denoted as $\mathbf{I}(\boldsymbol{\theta})$. Utilizing the flat sky approximation, the image center $\bar{\boldsymbol{\theta}}$ is expressed as [106]:

$$\bar{\boldsymbol{\theta}} = \frac{\int d^2\theta \mathbf{I}(\boldsymbol{\theta}) q_I[\mathbf{I}(\boldsymbol{\theta})] \boldsymbol{\theta}}{\int d^2\theta \mathbf{I}(\boldsymbol{\theta}) q_I[\mathbf{I}(\boldsymbol{\theta})]} , \quad (4.13)$$

where q_I us the cut-off scale for low-intensity images in order to avoid the contribution of detection noise to the integral. The second-order brightness moments, embodied by Q_{ij} [106], can be written as:

$$Q_{ij} = \frac{\int d^2\theta \mathbf{I}(\boldsymbol{\theta}) q_I[\mathbf{I}(\boldsymbol{\theta})] (\theta_i - \bar{\theta}_i)(\theta_j - \bar{\theta}_j)}{\int d^2\theta \mathbf{I}(\boldsymbol{\theta}) q_I[\mathbf{I}(\boldsymbol{\theta})]} . \quad (4.14)$$

These moments yield the ellipticity of the light distribution $\mathbf{I}(\boldsymbol{\theta})$, characterized by:

$$\varepsilon = \frac{Q_{11} - Q_{22} + 2iQ_{12}}{Q_{11} + Q_{22} + 2(Q_{11}Q_{22} - Q_{12}^2)^{\frac{1}{2}}} . \quad (4.15)$$

The interplay of the shear field and the observed ellipticity ε can be understood via [107]:

$$\varepsilon = \frac{\varepsilon^{\text{int}} + g}{1 + g * \varepsilon^{\text{int}}} . \quad (4.16)$$

In the weak lensing regime ($\kappa \ll 1$), the reduced shear approximates the complex shear ($g \approx \gamma$). In light of the overwhelming contribution of shape noise to the cosmic shear signal, a statistical assumption is applied, positing random orientations for unlensed galaxies. This leads to the following approximation⁴:

$$\langle \varepsilon \rangle \approx g \approx \gamma . \quad (4.17)$$

⁴This approximation simplifies the analysis by neglecting effects such as intrinsic alignments and gravitational lensing by galaxy shear. These additional complexities are beyond the scope of this Master's thesis.

4.2.2 Cosmic Shear and Convergence Field

In this subsection, we delve into the modeling of the cosmic shear and convergence field, two crucial aspects in gravitational weak lensing.

We commence our exploration with the Born approximation, asserting that the deflection caused by lensing can be described through linear perturbations. Focusing our attention on the linearized general relativistic Poisson equation (3.48) operating within the subhorizon regime ($k \ll \mathcal{H}$), we have:

$$\Delta\Phi = 4\pi a^2 \bar{\rho}_m \delta_m . \quad (4.18)$$

Drawing an analogy with the Newtonian Poisson equation, we recognize that the term $a^2 \bar{\rho}_m \delta_m$ serves as the source responsible for the gravitational potential in an expanding universe. This motivates us to define the conformal matter density [91] as:

$$\delta_{\text{conf}} = a^2 \bar{\rho}_m \delta_m . \quad (4.19)$$

Leveraging equation (3.27), we arrive at the expression for the conformal matter density:

$$\delta_{\text{conf}} = \frac{3H_0^2 \Omega_{m,0}}{8\pi} \frac{\delta_m}{a} . \quad (4.20)$$

The exploration continues by considering infinitesimal shells $d\chi'$ of comoving distance around the observer. This allows us to express the conformal surface density as:

$$d\Sigma_{\text{conf}}(\boldsymbol{\theta}, \chi') = d\chi' \delta_{\text{conf}}[\boldsymbol{\theta}, \chi'] = d\chi' \frac{3H_0^2 \Omega_0}{8\pi} \frac{\delta_m[\boldsymbol{\theta}, \chi']}{a(\chi')} . \quad (4.21)$$

Moving forward, we consider a light source situated at the comoving distance χ . Assuming a flat universe, we express the surface critical density in terms of comoving distances χ and χ' :

$$\Sigma_{\text{crit}}(\chi', \chi) = \frac{1}{4\pi} \frac{\chi}{\chi'(\chi - \chi')} . \quad (4.22)$$

Subsequently, we formulate the lensing convergence within a certain shell as follows:

$$d\kappa(\boldsymbol{\theta}, \chi', \chi) = \frac{d\Sigma(\boldsymbol{\theta}, \chi')}{\Sigma_{\text{crit}}(\chi', \chi)} . \quad (4.23)$$

This lays the foundation for defining the effective convergence field:

$$\kappa_{\text{eff}}(\boldsymbol{\theta}, \chi) \equiv \int_0^\chi d\chi' d\kappa(\boldsymbol{\theta}, \chi', \chi) = \frac{3H_0^2 \Omega_0}{2} \int_0^\chi d\chi' \frac{\chi'(\chi - \chi')}{\chi} \frac{\delta_m[\boldsymbol{\theta}, \chi']}{a(\chi')} . \quad (4.24)$$

Introducing a source galaxy redshift distribution $p(z)$, we define $g(\chi)$ as a galaxy redshift distribution with χ -dependency ($g(\chi)d\chi = p(z)dz$). This enables us to express the effective convergence field in terms of only $\boldsymbol{\theta}$:

$$\kappa_{\text{eff}}(\boldsymbol{\theta}) = \int_0^{\chi_{\text{max}}} d\chi g(\chi) \kappa_{\text{eff}}(\boldsymbol{\theta}, \chi) . \quad (4.25)$$

Having established a theoretical understanding of the convergence field through the density contrast δ , we now explore its connection to the observable cosmic shear signal. The convergence field, denoted as $\kappa = \kappa_E + i\kappa_B$ ⁵, represents a scalar field⁶, while the shear field $\gamma = \gamma_E + i\gamma_B$ assumes a tensorial nature⁷. Harmonic space provides a convenient framework to describe both fields. The shear field γ can be expanded using harmonic coefficients $\hat{\gamma}_{lm}$:

$$\gamma = \sum_{lm} \hat{\gamma}_{lm} Y_{lm}^2 , \quad (4.26)$$

where Y_{lm}^2 are the spin-weight 2 spherical harmonics basis functions, and $\hat{\gamma}_{lm}$ are defined as:

$$\hat{\gamma}_{lm} = \int d\Omega \gamma(\theta, \varphi) Y_{lm}^{2*}(\theta, \varphi) . \quad (4.27)$$

Similarly, we describe the convergence field κ and the lensing potential ϕ using harmonic coefficients $\hat{\kappa}_{lm}$ and $\hat{\phi}_{lm}$. The relationship between these coefficients is given by:

$$\hat{\gamma}_{lm} = -\frac{(l-1)(l+2)}{l(l+1)} \hat{\kappa}_{lm} . \quad (4.28)$$

In a small sky region, the flat sky approximation allows us to establish the Kaiser-Squire inversion relation [70]⁸:

$$\gamma(l) = \frac{(l_x + il_y)^2}{l^2} \kappa(l) = \exp(2i\phi_l) \kappa(l) , \quad (4.29)$$

where $l = \sqrt{l_x^2 + l_y^2}$ and $\phi = \arctan\left(\frac{l_y}{l_x}\right)$.

4.2.3 Weak Lensing Convergence Power Spectrum

Now, I would like to introduce one of the most important quantities in our simulation pipeline: the convergence power spectrum. We begin with the effective convergence field, denoted as $\kappa_{\text{eff}}(\boldsymbol{\theta})$, and then reframe our approach as follows [6]:

$$\kappa_{\text{eff}}(\boldsymbol{\theta}) = \frac{3H_0^2 \Omega_{m,0}}{2} \int_0^\chi d\chi W(\chi) \chi \frac{\delta_m[\boldsymbol{\theta}, \chi]}{a(\chi)} , \quad (4.30)$$

where

$$W(\chi) = \int_\chi^{\chi_{\text{max}}} d\chi' g(\chi') \frac{\chi' - \chi}{\chi'} . \quad (4.31)$$

⁵The imaginary part is zero under the Born approximation, eliminating systematic effects.

⁶It corresponds to spin-weight 0 [113]

⁷It corresponds to spin-weight 2 [113]

⁸The ultimate goal of this project is to apply field-based cosmological parameter inference to real convergence maps obtained from measured cosmic shear. Detailed discussions about the conversion from the observed shear signal to convergence maps are provided in [67].

Then, we can observe that this equation takes a form similar to that of equation (2.65), sharing a resemblance with the projection kernel $q(r)$, given by:

$$q(\chi) = \frac{3H_0^2\Omega_{m,0}}{2} \frac{W(\chi)\chi}{a(\chi)} . \quad (4.32)$$

Next, employing the Limber approximation, we can derive the convergence power spectrum as follows:

$$C_l = \frac{9H_0^2\Omega_{m,0}^2}{4} \int_0^{\chi_{\max}} d\chi \frac{W(\chi)^2}{a(\chi)^2} P_{3D} \left(\chi, \frac{l}{\Phi_\kappa(\chi)} \right) . \quad (4.33)$$

As evident from the equation above, the weak lensing convergence power spectrum arises as a result of the line-of-sight projection of the 3D matter power spectrum.

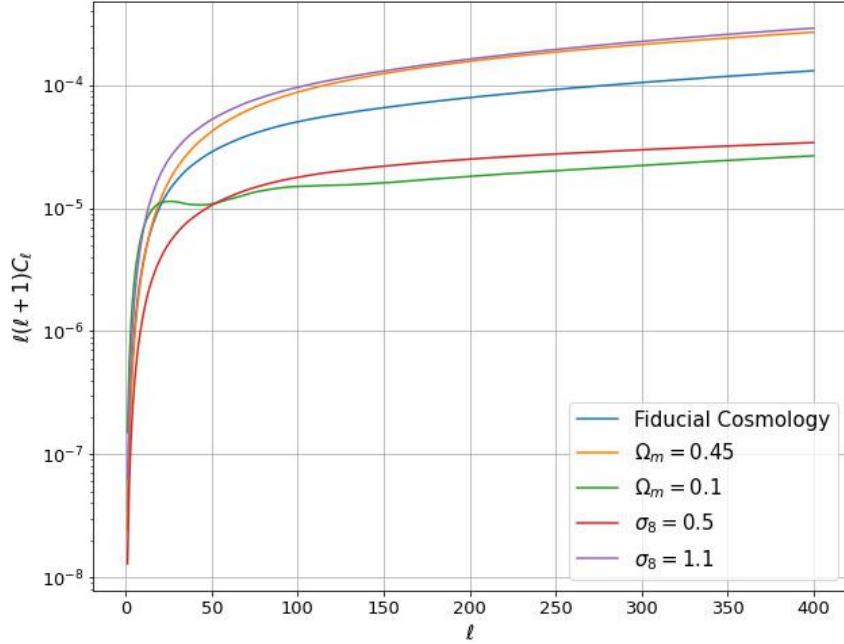


Figure 4.4: The non-linear matter power spectrum within the framework of a flat Λ CDM cosmology is generated using the CLASS code. For this study, we adopt fiducial cosmological parameters based on the WMAP 9-year results as presented in [61]. Additionally, we employ the lensing kernel from the second bin of [45]. Variations in the cosmological parameter values result in changes to both the overall shape and amplitude of the convergence power spectrum.

Chapter 5

Simulation Pipeline

My Master's thesis revolves around two distinct statistical pipelines: the simulation pipeline and the inference pipeline. Within the simulation pipeline, I focus on generating convergence density fields corresponding to a set of cosmological parameters. On the other hand, the inference pipeline operates in the reverse manner. It involves estimating cosmological parameters based on provided density fields. To facilitate the estimation process, I employ Graph Convolutional Neural Network (GCNN) models [96], which will be discussed in the upcoming chapter. The effectiveness of these models hinges on the availability of appropriately generated training datasets. Moreover, I leverage separate test and validation datasets to gauge the robustness and reliability of the neural network outcomes. The overarching objective of our research is to reconstruct cosmological posterior distributions using data from the N-body simulation [114]. This investigation aims to determine if log-normal fields can accurately reconstruct cosmological parameters from N-body simulations. For comparative analysis, I also incorporate a neural network model which is trained with Gaussian fields. This comparative approach serves to quantify any bias and discrepancies inherent in lognormal field-trained neural networks. By utilizing established 2-point statistics, it is possible to assess the consistency and reliability of the neural network outputs. In this section, I present the simulation pipeline along with its validation of outputs from this pipeline.

5.1 Simulation Setup

Before delving into the detailed structure of our simulation, I would like to present our experimental setup. In this master’s thesis, I am conducting an analysis of cosmological parameters using field-based inferences from three distinct test data sets. The objective is to comprehend and evaluate our field-based inference. Among the test datasets, one comprises a full-sky convergence map derived from the N-body simulation by Takahashi et al. (2017)[114]. The other two datasets consist of lognormal and Gaussian fields generated from **Flask** simulations [124], which are elaborated upon in this chapter. The T17 simulations encompass 108 realizations covering different random realizations for a single cosmology, which is based on the WMAP 9-year estimation [61]. Additionally, I have generated 300 lognormal and Gaussian fields with the same fiducial cosmology, constituting our test datasets. Accurate estimation of parameters within high-dimensional parameter spaces requires substantial computational resources [4], mainly due to the curse of dimensionality. Therefore, our focus narrows down to two crucial cosmological parameters: Ω_m and σ_8 for the estimation. Meanwhile, we keep other cosmological parameters fixed at the fiducial 9-year WMAP values [61]. Both Ω_m and σ_8 benefit from stringent constraints derived from prior weak lensing analyses [28], [5], [54]. Additionally, σ_8 faces tension with respect to the CMB experiment [97], while the strong degeneracy observed in the Ω_m - σ_8 plane [105] emphasizes their importance in our analysis. To conduct successful map-based inference using machine learning models, we must prepare qualitative and quantitative training and validation datasets. In this context, we generate lognormal and Gaussian fields using the **Flask** simulation [124]. While conventional projects involving statistical inference of cosmic density fields often rely on maps from N-body simulations [68], [120], we opt for **Flask** [124] due to its computational efficiency¹ and flexible experimental setup. We choose a prior range of $\Omega_m = [0.1, 0.45]$ and $\sigma_8 = [0.5, 1.1]^2$. Within this prior range, we sample independently 1000 cosmologies for training datasets and 250 cosmologies for validation datasets using Latin Hypercube Sampling (LHS) [112]. Furthermore, we generate 10 lognormal and 10 Gaussian random realizations for each cosmology to estimate accurately cosmic variance.

¹Whole simulation takes 32 seconds from the cosmological parameters to generate 10 lognormal and 10 Gaussian random realizations.

²Initially, we tested our simulation and inference pipeline using the lowest resolution ($NSIDE = 32$) with DES priors [28], which revealed two σ contours. Based on this observation, we selected a broader prior range, considering the common issue in neural networks where predictions tend to be less accurate at the boundaries of training dataset ranges.

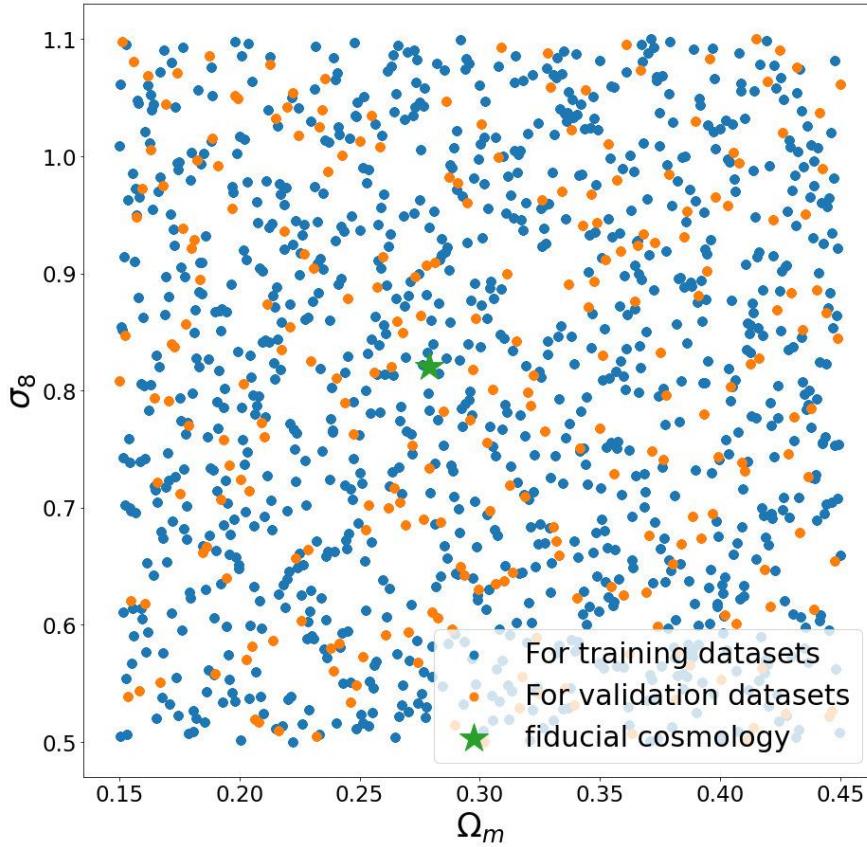


Figure 5.1: From the prior range ($\Omega_m = [0.1, 0.45]$, $\sigma_8 = [0.5, 1.1]$), we employ LHS sampling as outlined in the methodology from [112]. The resulting figure visually displays each sampled cosmology for both the training and validation datasets, along with the fiducial cosmology of the test datasets, within the 2D $\Omega_m - \sigma_8$ plane.

5.2 Structure of Simulation Pipeline

To begin, I wish to provide an overview of our simulation pipeline’s structure. The pipeline initiates with a set of cosmological parameters sampled through Latin Hypercube Sampling (LHS), as elaborated upon in the preceding subsection. The subsequent steps unfold as follows 5.2:

- (i) **Matter Power Spectrum Generation:** Firstly, I generate the power spectrum for the provided cosmological parameters using the Boltzmann solver, which is `CLASS` [76].
- (ii) **Projection Integral:** The line of sight integration of matter power spectrum yields the convergence angular power spectrum. Here is employing the source galaxy redshift distribution from Gong et al. (2023) [45].

- (iii) **Lognormal Shift Parameter:** To realize the lognormal field, a lognormal shift parameter is indispensable, as discussed in Chapter 1. This parameter is computed for the given cosmologies using **CosMomentum**, a tool to model the PDF of cosmic density fields introduced by Friedrich et al. (2020) [42].
- (iv) **Generation of Convergence Maps:** Lastly, I generate full-sky lognormal convergence maps along with corresponding Gaussian convergence maps for the training and validation datasets. These maps are produced using the input power spectra and log-normal shift parameters, employing **FLASK**, a tool introduced by Xavier et al. (2016) [124].

Each of the aforementioned steps—power spectrum generation (i), convergence angular power spectrum computation (ii), lognormal shift parameter calculation (iii), and convergence maps generation (iv)—is expounded upon in the subsequent subsections.

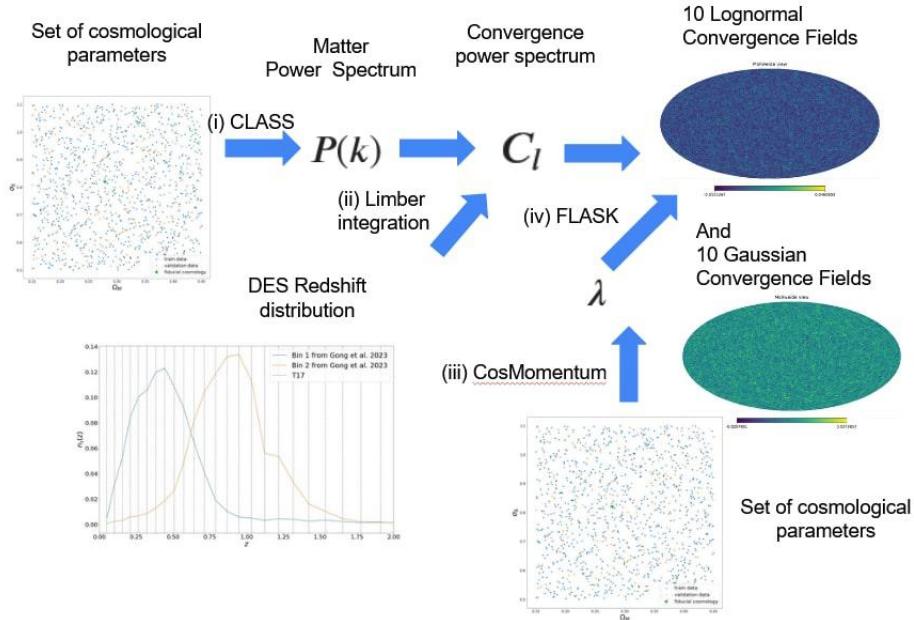


Figure 5.2: The sketch of the simulation pipeline in this thesis. The numbers such as (i) correspond to the number of text above.

5.2.1 Matter Power Spectrum Generation

The initial phase of the simulation pipeline involves the generation of the matter power spectrum for given sets of cosmological parameters. While an analytical approach exists for calculating the matter power spectrum, accomplished either manually by hand as outlined in Mukhanov (2004) [88], or through the utilization of fitting functions derived from observations and numerical simulations as demonstrated by Eisenstein and Hu (1998) [33], both methods necessitate certain approximations. However, for tasks demanding high precision

and encompassing a broad range of cosmological models inclusive of gravitational interactions, radiation, and thermodynamics, enhanced accuracy is attainable by employing comprehensive Boltzmann solvers such as **Class**(Cosmic Linear Anisotropy Solving System) [76] or **CAMB** [77]. These Boltzmann solvers are computer programs that numerically solve a set of equations comprising linearized Einstein field equations and the Boltzmann equation of photons and neutrinos. This enables the modeling of the evolution of linear perturbations within the universe, ultimately facilitating the prediction of observables pertinent to Cosmic Microwave Background (CMB) or large-scale structure cosmology. Given these considerations, our Master thesis adopts the utilization of **CLASS** to compute the matter power spectrum. In the realm of weak lensing data analysis, the presence of non-linearity within cosmic density fields demands the computation of a non-linear matter power spectrum. To address this need, the phenomenological semi-analytical fitting formula known as **Halofit** [12] is adopted within the **CLASS**. The **Halofit** model provides corrections, founded on the modeling of dark matter halos within N-body simulations, which offers an effective correction from the linear to the non-linear matter power spectrum.³ The outcomes stemming from this phase of the simulation are illustrated in figures 3.7 and 3.8 in Chapter 4.

5.2.2 Projection Integral

Once the computation of the matter power spectrum is concluded, the subsequent step involves projecting this power spectrum along the line of sight. This projection is achieved through the utilization of the Limber integration (4.33), which can be expressed in terms of redshifts z for practical computation as follows:

$$C_l = \int_0^{z_{\max}} \frac{1}{H(z)} dz \frac{q^2(z)}{\chi^2(z)} P_{3D} \left(|\mathbf{k}| = \frac{|l|}{\chi(z)}, z \right). \quad (5.1)$$

Here, the projection kernel⁴ (4.32) is rewritten by the equation in terms of redshift:

$$q(z) \frac{3H_0^2 \Omega_m}{2} (1+z) \chi(z) \int_0^z dz' n_s(z') \frac{\chi(z') - \chi(z)}{\chi(z)}. \quad (5.2)$$

During practical computation, equation (5.1) is evaluated using trapezoidal integration, and the integral portion of (5.2) is approximated as a summation across all involving redshift bins. This approach is adopted due to the smooth nature of both approximated functions, significantly enhancing the computational efficiency of the double integral. The source redshift distribution $n_s(z)$ is used as the bin2 distribution from Gong et al. (2023)

³Notably, the halo model is employed to conceptualize the universe with all matter content confined within dark matter halos. This model further decomposes the power spectrum into one- and two-halo terms, where the former characterizes autocorrelation and the latter delineates correlations with other dark matter halos. A comprehensive elucidation of this concept is presented in Takahashi (2012) [115].

⁴In the actual code, I compute the division of the projection kernel by the comoving distance first, as opposed to the definition below. This is done to avoid the division of an extremely large number.

[45]. These redshift bins are a combination of four DES Y3 redshift source bins, depicted in Figure 5.3, and the same bins are employed for projecting T17 simulations. It is important to note that a more intricate analysis [45] than the one presented in this thesis takes into account photometric redshift uncertainties. These uncertainties are represented by a single shift parameter Δz , where the source redshift distribution $n_s^i(z)$ is related to the estimated distribution \hat{n}_s^i for a single source redshift bin i through the relation:

$$n_s^i(z) = \hat{n}_s^i(z + \Delta z^i). \quad (5.3)$$

However, such considerations fall beyond the scope of the present thesis.

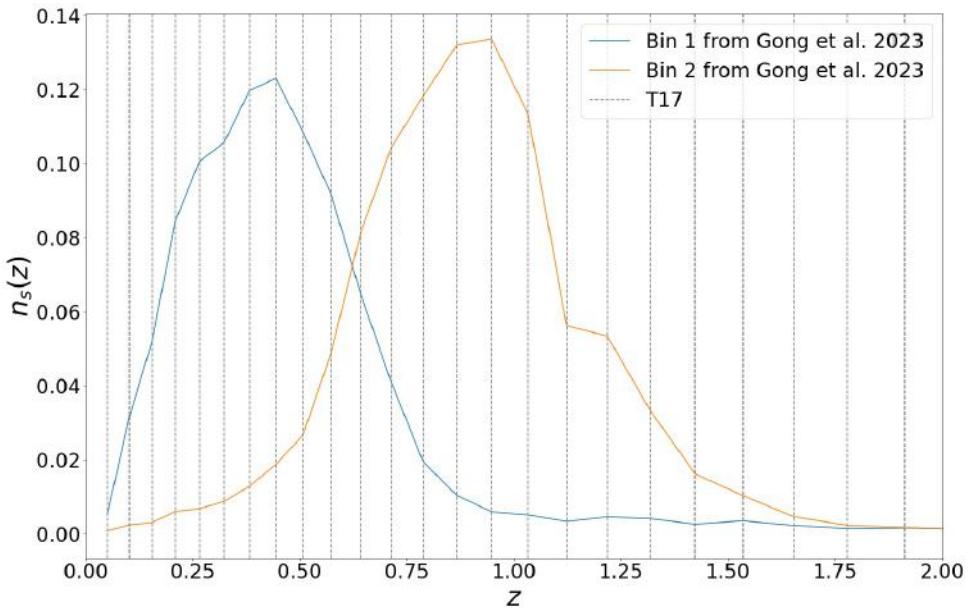


Figure 5.3: The two source redshift distribution from [45] is plotted in this figure. The gray dotted line corresponds to the binning of the redshift distribution for generating T17 convergence maps. Bin2 is used for our simulation pipeline.

5.2.3 Lognormal Shift Parameter

To accurately define the lognormal distribution of convergence maps for a given cosmology, one must consider both the convergence power spectrum and the cosmological dependency of the lognormal shift parameter. To fulfill this purpose, I am making use of the `CosMomentum` tool as presented in [42]. This code takes as its input a set of cosmological parameters, which I've sampled using Latin Hypercube Sampling⁵ from a predefined

⁵While it is often optimal to employ sophisticated techniques for sampling in high-dimensional parameter spaces, it's important to note that in this thesis, we are working with only two parameters. In such cases, alternative methods like grid sampling or random sampling may not yield significantly different results.

range of prior values. The result of running this code is determining the lognormal shift parameter specific to the chosen cosmological configuration. In essence, **CosMomentum** is a software code designed to generate the PDF and moments of the cosmic density field by leveraging principles from perturbation theory⁶.

5.2.4 Generation of Convergence Maps

Finally we can generate the random field from lognormal or Gaussian PDF from the convergence power spectrum and lognormal shift parameter. This is achieved through the application of the **Flask**(Full-sky Lognormal Astro-fields Simulation Kit) [124], a publicly available code designed to produce a **Healpy** sky map [47] that embodies a realization of a lognormal random variable. The internal steps of Flask can be summarized as follows[52]:

$$C_l \xrightarrow{(2.59)} \xi_{2D} \xrightarrow{(2.59)} \xi_{2D,\log} \xrightarrow{\text{treatas}} \xi_{2D,Gauss} \xrightarrow{(2.27)} C_{l,Gauss} \xrightarrow{(2.64)} HEALPY \rightarrow \delta_{Gauss} \xrightarrow{(2.23)} \delta_{\log}$$

Within this sequence, δ_{\log} and δ_{Gauss} signify the lognormal and Gaussian random fields respectively. Both δ_{\log} and δ_{Gauss} are produced through 10 separate random realizations for every single cosmological configuration. These generated fields subsequently serve as datasets for training, validation, and testing in the construction of the model using GCNN. The motivation behind constructing the GCNN model using lognormal convergence fields stems from the pursuit of computational efficiency and its notable congruence with the 1-PDF derived from the projected density in N-body simulations [41]. Indeed, a critical aspect is that an abundant collection of random and cosmological realization is required for effectively training the neural network. It is important to note that the accuracy of lognormal field approximations is constrained on smaller scales. Consequently, investigating the precision and impact on the posterior distribution of cosmological parameters through the field-based method using a lognormal field, in comparison to the more accurate N-body simulation, presents a non-trivial task. Within the scope of this thesis, the Gaussian field is also employed for modeling another variant of the GCNN model. This allows for a direct comparison between the GCNN trained with lognormal fields and the Gaussian-field-based model, while also providing a robust assessment through 2-point statistics for the GCNN itself.

⁶It is also possible to calculate the PDF starting from non-Gaussian initial conditions.

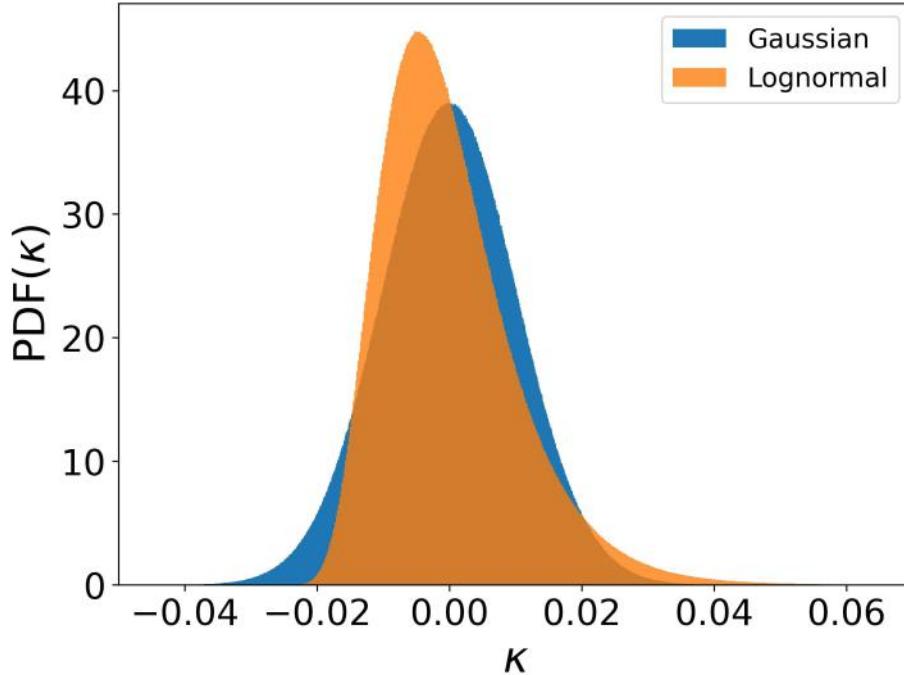


Figure 5.4: PDF of pixel values of lognormal and Gaussian `Flask` maps in $NSIDE = 2048$. Both are generated from the same input as the power spectrum.

5.2.5 Execution Time of the Simulation Pipeline

Efficient execution time of the simulation pipeline plays a pivotal role in this project, particularly considering the necessity to generate a substantial number of maps for training the GCNN model. Throughout the course of this project, we generated a total of 10,000 lognormal maps (comprising 1,000 different cosmological parameter sets, each with 10 random realizations) and 10,000 Gaussian maps, all at an angular resolution of $NSIDE = 128$. Additionally, for validation purposes, 2,500 lognormal maps (from 250 different cosmological parameter sets, again with 10 realizations each) and 2,500 Gaussian maps were generated at the same resolution.

The execution time for a single set of cosmological parameters averages around 30 seconds when utilizing 30 CPU cores. It's worth noting that parallel processing can further enhance the speed. The breakdown of this time includes 6 seconds for generating the matter power spectrum using `Class`⁷, 5 seconds for the Limber integral, 7 seconds for the `CosMomentum` component, and an additional 1 second for each random realization generated using `Flask`.

Our observations indicate that this total execution time scales linearly with the number of multipoles l . To manage this, I've capped the maximum l at 3000. While this speed

⁷Further acceleration can be anticipated by substituting the model parameter σ_8 with A_s .

suffices for generating training datasets with just two parameters, expanding the parameter space will necessitate even greater speed improvements. It is anticipated that the most time-consuming component will be the Boltzmann solver. Therefore, future research may involve employing an emulator for the power spectrum generated by the Boltzmann solver⁸, such as COSMOPOWER [112], or emulating the entire process to achieve further efficiency gains.

5.3 Validation of Simulation Pipeline

The primary determinant of a successful machine learning project lies in the quality and quantity of training datasets employed. Moreover, comprehending the inherent distinctions between lognormal fields and more realistic matter density fields derived from N-body simulations holds significance. Consequently, a thorough evaluation of the output yielded by the simulation pipeline becomes imperative. This evaluation entails statistical attributes, such as the convergence power spectrum, as manifested in the convergence maps originating from both the simulation pipeline under consideration and the N-body simulation. To achieve this objective, it becomes indispensable to assimilate the correcting factors impacting the convergence maps in the context of N-body simulations into the simulation pipeline under study. This endeavor necessitates the incorporation of these factors into the simulation process. This section commences with an elucidation of the T17 simulation by Takahashi et al. (2017)[114], which constitutes a pivotal foundation for this undertaking. Subsequently, the focus transitions to the introduction of two correction factors that have been seamlessly integrated into the simulation pipeline. The viability and soundness of the simulation pipeline are thereby demonstrated and validated.

5.3.1 T17 N-body Simulation

Prior to elucidating the validation test, I wish to introduce the T17 simulation. The present investigation utilizes publicly accessible synthetic datasets that have been derived from the T17 cosmological N-body simulations by Takahashi et al. (2017) [114]. These simulations were primarily designed for gravitational weak lensing surveys. The T17 simulations are constructed based on a cold dark matter (CDM) cosmological N-body framework, encompassing a set of periodic cubic volumes, with side lengths progressively increasing ($L, 2L, 3L, \dots, 14L$), where L equals $450\text{MPC}/h$. These volumes are situated around a common vertex. Each individual cubic volume contains 2048^3 particles. The initial conditions are established through second-order Lagrangian perturbation theory, aligned with a flat ΛCDM cosmology defined by the WMAP 9-year cosmological parameters [61]. The temporal evolution of particle positions and velocities is governed by the N-body gravity solver known as GADGET2 [111]. Subsequently, the trajectory of evolved particle distributions in each cubic volume is traced using the multi-lens plane ray-tracing algorithm denoted as GRAYTRIX [55]. This process enables the generation of comprehensive lensing maps, such as convergence maps, across the entire celestial sphere. These lensing maps are obtained

⁸Replacing the remaining components from CPU to GPU can be accomplished with relative ease.

by employing several Dirac- δ source redshift bins. To ensure the statistical validity of the findings, a multitude of simulations were executed, leading to the generation of 108 distinct realizations for each data product. Comparisons between the mean matter power spectra extracted from these realizations and the theoretical angular power spectra associated with cosmic shear and cosmic microwave background (CMB) anisotropies were conducted. The results revealed an accord within 5% for angular scales up to $l = 3000$.

5.3.2 Validation with T17 Simulations

Incorporation of correction factors is imperative within the simulation pipeline to ensure alignment between the computed power spectrum and the observed power spectrum obtained from the T17 simulation. In the Appendix section of Halder et al. (2021) [52], important three factors are delineated as follows:

- (i) **Finite simulation-box-size effect:** The precision of the power spectrum is influenced by the limited size of the simulation volume. Spatial frequencies that exceed the limit of Nyquist frequency, defined as $k < k_{\text{Nyq}} = \frac{2\pi}{L}$, manifest as aliased frequencies within the observable range due to the discretized nature of the simulation grid. A resolution to this issue is proposed by Takahashi et al. (2017)[114], where $P(k, a) = 0$ serves as the remedy in this context.
- (ii) **Finite lens-shell effect:** When employing the ray tracing method via lens shells, it becomes imperative to account for the finite thickness inherent to each individual lens shell. This phenomenon significantly influences the characteristics of the power spectrum. In this regard, [114] offers a fitting formula obtained through the convolution of the power spectrum and the window function of the lens shells, yielding the subsequent expression:

$$P(k, a) \rightarrow \frac{(1 + c_1 k^{-\alpha_1})^{\alpha_1}}{(1 + c_2 k^{-\alpha_2})^{\alpha_3}} P_{3D}(k, a) , \quad (5.4)$$

where $c_1 = 9.5171 \times 10^{-4}$, $c_2 = 5.1543 \times 10^{-3}$, $\alpha_1 = 1.3063$, $\alpha_2 = 1.1475$ and $\alpha_3 = 0.62793$.

- (iii) **Finite angular resolution of sky-maps:** The measured angular power spectrum C_l demonstrates an underestimation in relation to the corresponding theoretical spectrum. This discrepancy is primarily attributed to the inherent angular resolution limitations of the utilized `Healpix` map [47]. In response to this, Takahashi et al. suggest the incorporation of a damping factor at smaller scales to effectively mitigate this angular resolution effect. This proposition can be concisely expressed as follows:

$$C(l) \rightarrow \frac{C(l)}{1 + (\frac{l}{l_{\text{res}}})^2} , \quad (5.5)$$

where $l_{\text{res}} = 1.6 \cdot NSIDE$.

We proceed with an assessment of these correction factors. The impact of these correction formulae on the theoretical power spectrum, particularly at $NSIDE = 2048$, is depicted in Figure 5.5. Insights drawn from this plot include the following observations:

Firstly, it becomes evident that the contribution stemming from the finite lens-shell effect within our simulation pipeline is almost negligible. However, to ensure comprehensiveness, I have also incorporated this effect into our simulation pipeline. Conversely, the finite angular resolution effect significantly influences the alignment between the theoretical angular power spectrum and the measured counterpart, particularly with pronounced effects on smaller scales.

Despite these considerations, it is important to note that the overall agreement between our theoretical power spectrum and the measured power spectrum remains less than optimal, particularly in the context of smaller scales. However, ultimately, the level of agreement at higher l values does not hold significant importance for the inference pipeline. This is because I am constrained to downsample all the acquired maps from $NSIDE = 2048$ to $NSIDE = 128$ due to computational limitations imposed by the training process. Consequently, the finer-scale structures are inherently smoothed out during this downsampling procedure⁹. Figure 5.6 demonstrates the comparison between the angular power spectra obtained from the outputs of our **Flask** simulation pipeline and the T17 simulations [114], encompassing l values up to 400^{10} . This plot succinctly showcases the exceptional concordance achieved within the desired scales of interest.

Once we validate the second moments by angular power spectra, our attention should shift towards the higher moments of the PDF, as these aspects also play a significant role in characterizing non-Gaussian distributions. Figure 5.7 illustrates the fractional differences in the measured skewness and kurtosis between the **Flask** lognormal simulation [124] and the T17 simulation [114]. Notably, at larger $NSIDE$ values that correspond to smaller scales, the fractional differences exceed 50%. This effect arises due to the fixed angular resolution that I employed in the **CosMomentum** simulation. In our inference pipeline, I maintain a constant angular resolution of $NSIDE = 128$. This results in an approximate 35% difference for skewness and an approximate 50% difference for kurtosis. However, these disparities are insufficient for accurately assessing the inherent distinctions between the lognormal approximation and N-body simulations. I intend to rectify this limitation in future work.

⁹A comprehensive and detailed analysis of these findings is expounded upon in Chapter 7.

¹⁰Indeed, the **healpy** function [126] is designed to calculate up to $l = 3 \cdot NSIDE - 1$. It's important to note that $l = 400$ lies significantly above this range, surpassing the angular resolution capabilities inherent to **Healpix** [47].

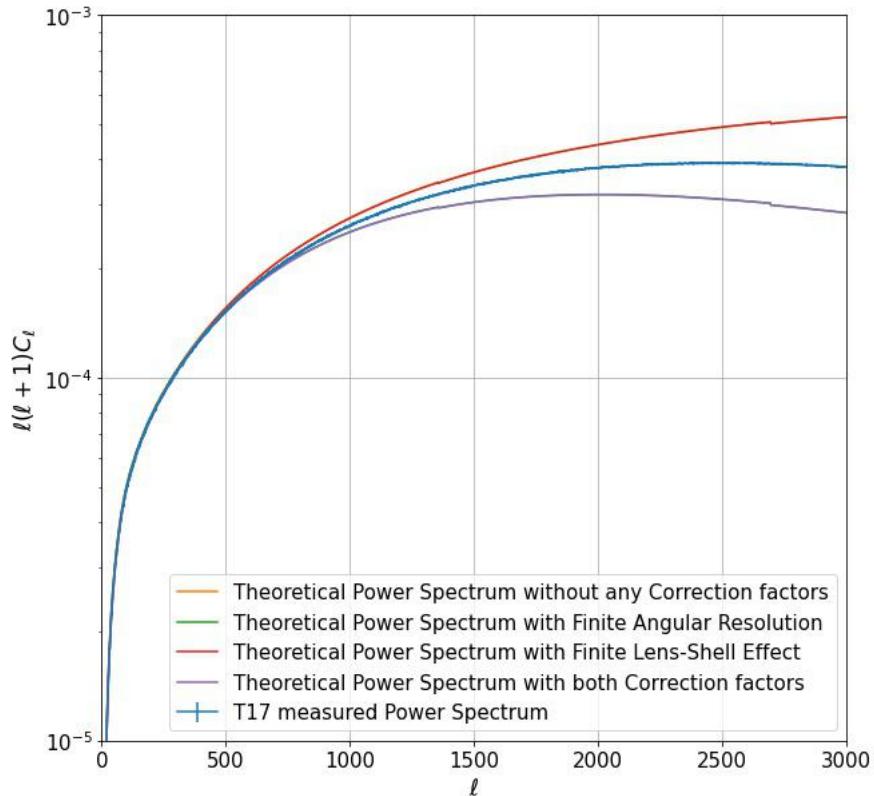


Figure 5.5: The plotted blue line illustrates the mean of the measured power spectrum, derived from the convergence power spectrum of 108 realizations within the T17 simulation by Takahashi et al. (2017) [114] at $NSIDE = 2048$. The remaining lines depict various theoretical angular power spectra, each incorporating distinct correction factors at the same angular resolution. Prior to their integration into the `Flask` software [124], these theoretical spectra are subjected to these correction factors for comparative analysis.

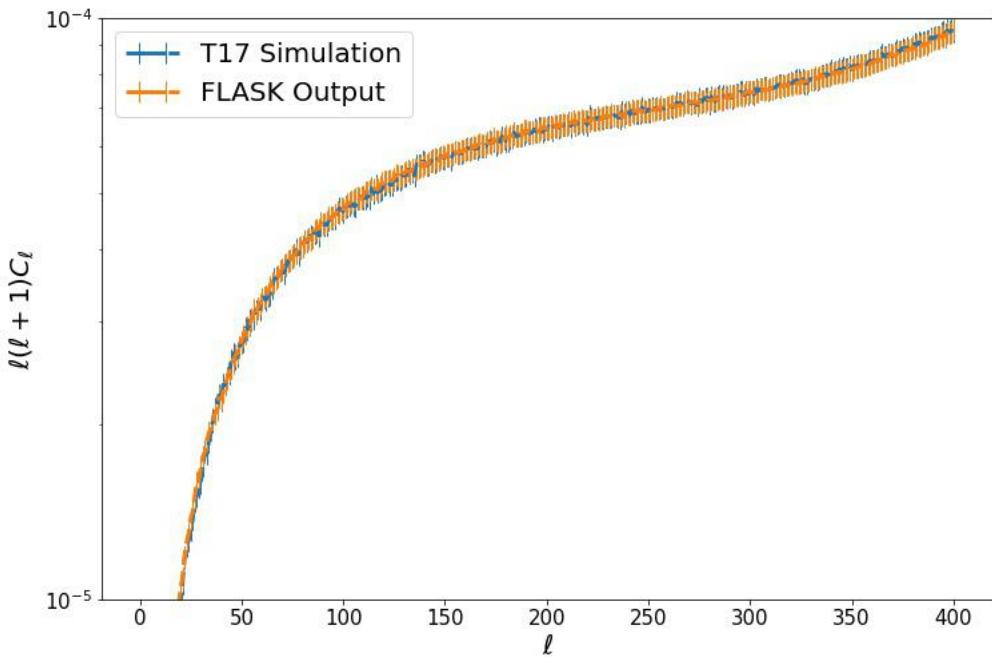


Figure 5.6: The dashed blue line corresponds to the average of the measured angular power spectrum derived from 108 realizations of the T17 simulation by Takahashi et al. (2017) [114]. For this comparison, the angular resolution is reduced to $NSIDE = 128$. In contrast, the dashed orange line is generated from the average of the measured angular power spectrum obtained from 300 realizations produced by our simulation pipeline. In this case, as well, the maps are degraded to $NSIDE = 128$ resolution. The associated error bars represent the measured variance in each case.

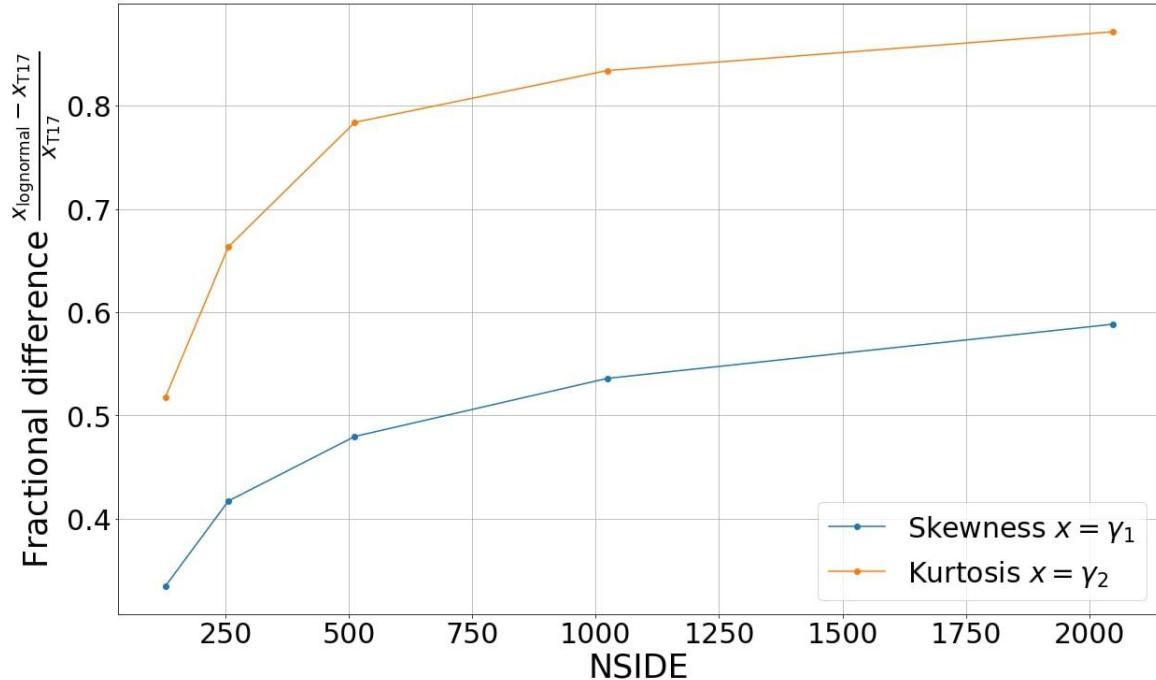


Figure 5.7: The y-axis displays the fractional difference in higher moments between the `Flask` lognormal simulation and the T17 simulation, while the x-axis represents the angular scale given in $NSIDE$. The measured skewness is represented by the blue curve, and the orange curve represents the kurtosis. These curves depict the averages of 300 realizations for `Flask` simulations and 108 realizations for T17 simulations, all conducted for the same cosmology.

Chapter 6

Theory of Neural Networks

The central focus of this Master’s thesis revolves around the implementation of inference methods that extend beyond traditional 2-point statistics. Hence, the utilization of field-based inference methodologies, using the power of artificial intelligence, is employed to tackle the regression of Λ CDM cosmological parameters. To achieve this, I am utilizing the `DeepSphere` framework [27], [96], a graph convolutional neural network optimized for processing spherical data such as `Healpix` maps [47]. Before delving into the details of our inference pipeline, it’s essential to establish a foundational understanding of artificial intelligence, particularly the realm of graph convolutional neural networks.

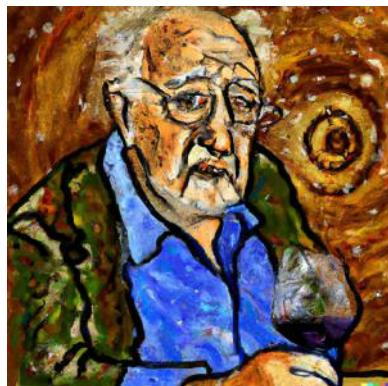


Figure 6.1: This illustration is generated by DALL-E 2 [100], a sophisticated generative deep neural network developed by OpenAI (accessible at <https://openai.com/dall-e-2>). Leveraging natural language descriptions as prompts, these deep learning models create digital images. In this instance, the prompt "An old professor from Eastern Europe who is enjoying the wine from Chianti, since Planck(Satelite) measured " $n_s = 0.96$ "", is painted in the style of Vincent van Gogh" was used to produce the depicted image.

Artificial intelligence is being used across various domains. This approach has brought about significant advancements not only in academia but also in everyday life. Examples include cancer detection [34], self-driving cars [35], and recommendation systems like those of YouTube [24] and Netflix [18]. The question arises: Why has artificial intelligence

become so remarkably successful currently? The answer lies in the convergence of hardware capabilities that can handle substantial computational loads¹ and the availability of user-friendly APIs, such as **Keras**[19] and **Scikit-Learn**[50], that simplify the implementation process.

This section draws from a compilation of reviews and books, including references such as [19], [50], [46], [57], [27], and [96].

6.1 Terminology

Let's begin by clarifying the terminology surrounding "Artificial Intelligence" (AI), "Machine Learning" (ML), and "Deep Neural Networks" (DNN), as these terms are often used interchangeably but carry distinct meanings. The Venn diagram in Figure 6.2 provides a visual representation of their relationships.

"Artificial Intelligence" (AI) encompasses the broader concept of automating tasks that traditionally require human intelligence. It's the overarching field focused on creating machines capable of intelligent behavior. One of the approaches within AI is "Machine Learning" (ML), where machines learn from data to improve their performance on specific tasks. ML is a subset of AI, targeting the development of algorithms and models that can learn patterns and make predictions from data.

"Deep Learning" is a subset of "Machine Learning" that specifically refers to the use of deep neural networks. These networks consist of multiple layers of interconnected nodes that process and transform data in a hierarchical manner. Deep learning has proven to be remarkably effective in capturing complex patterns and representations from data, making it a powerful tool in various AI applications.

6.1.1 Artificial Intelligence

The concept of "Artificial intelligence" was invented in the 1956 proposal of young professor John McCarthy [19]. During the early stages of computer science, the fundamental question revolved around whether computers can think like humans. The Turing test² emerged as a prominent example, assessing a machine's ability to mimic human behavior convincingly. AI encompasses various techniques, including machine learning, while also encompassing non-learning methods like Symbolic AI, where many rules are explicitly programmed without the need for training. However, the latter approaches faced challenges when dealing with intricate and ambiguous problems like computer vision and natural language processing, as explicit rule coding proved impractical. As a solution, machine learning emerged, enabling systems to learn from complex datasets [19].

¹Commercial CPUs have become roughly 5000 times faster between 1990 and 2010 [19]. Furthermore, GPUs and TPUs have emerged as specialized processors for parallel calculations, such as matrix multiplications.

²Originally known as the imitation game [117]

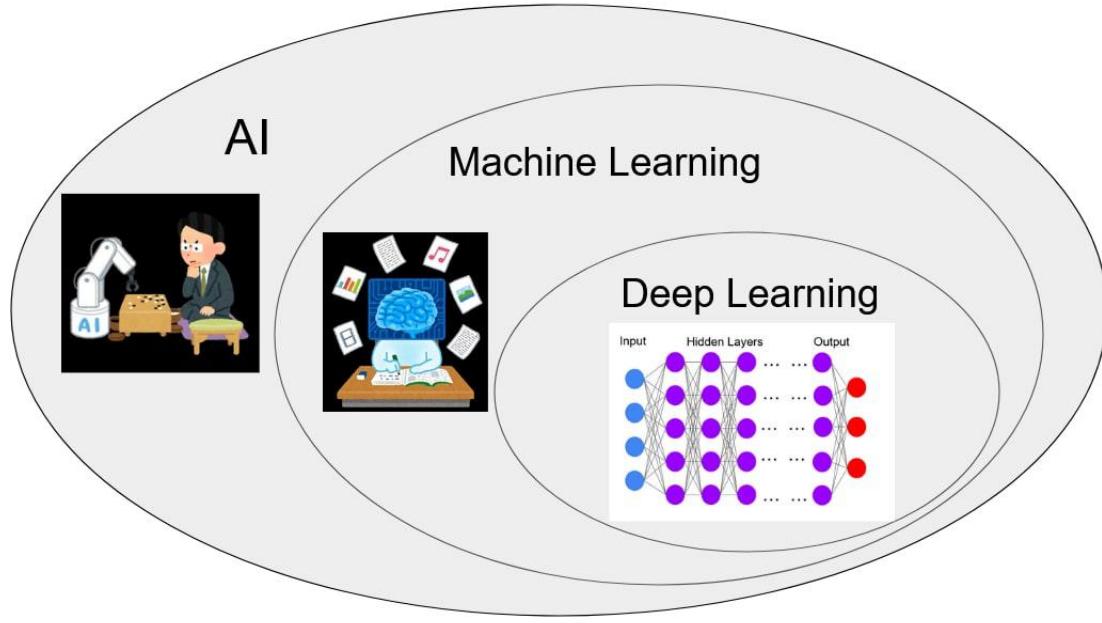


Figure 6.2: The Venn diagram which shows the ideas of "AI", "Machine Learning" and "Deep Learning. The illustrations are used from <https://www.irasutoya.com/>.

6.1.2 Machine Learning

Let's explore the precise distinctions between a conventional computer program and machine learning³. The foundational type of computer is the analytical engine, conceived by Charles Babbage in the 1840s [19]. This machine was programmable via punch cards, and Ada Lovelace, his collaborator, devised a method for calculating Bernoulli numbers using it [56]. The modern computer operates on the von Neumann architecture, which stores program instructions in electrically accessible memory—a departure from the analytical engine's design [121]. Both machines could only execute tasks explicitly programmed by human programmers. Could machines generate rules from input data? Arthur Samuel defined machine learning as the field enabling computers to learn without explicit programming [103]⁴. This process of improving program ability using training data is termed "training the machine." During training, the machine's adjustment is guided by a metric measuring the distance between output and expected output. The machine then learns the optimal representation of input data by exploring suitable algebraic operations in the

³While many researchers, including the author of this thesis, might not hesitate to employ computer programs such as Boltzmann solvers or MCMC for natural science, however, there is often a more cautious stance when it comes to embracing machine learning techniques. Thus, the aim is to delve into the precise origins of this hesitancy and examine the underlying factors that contribute to this apparent aversion.

⁴A more pragmatic definition by Tom Mitchell in 1997 states that a computer program learns from experience E for a task T with performance measure P if its performance on T , measured by P , improves with experience E [87].

hypothesis space.

So, how do we train the network? Broadly speaking, there are two categories for training machines⁵. The first is supervised learning, in which input data is accompanied by corresponding "labels," representing the desired outputs. The machine learns the general mapping rule from inputs to outputs by minimizing a defined loss function. In contrast, unsupervised learning involves learning from input data alone, without explicit labels. Instead, the machine focuses on the statistical structures within the data, often identifying commonalities through methods like clustering. This thesis employs supervised learning. I generate numerical simulations with associated cosmological parameters and train machine learning models using this simulated data, using the numerical simulations as inputs and the cosmological parameters as labels.

6.1.3 Deep Neural Network

Then, let's delve into the concept of neural networks. The inspiration behind neural networks stems from the attempt to model the intricate networks of biological neurons in our brains. This mathematical modeling commenced with McCulloch and Pitts' proposal [83] of the MP neuron, which constitutes a foundational binary input to binary output transformation. The MP neuron decision threshold governs whether the binary output surpasses it. This initial neuron model can express Boolean functions; however, a drawback emerges: MP neurons cannot undergo training.

The perceptron, an extension of the MP neuron⁶ [102], adopts the Heiside function, involving input-weight multiplications. The perceptron's crucial innovation resides in its introduction of a learning algorithm via backpropagation, which adjusts weights based on the discrepancy between predicted and target outputs. This algorithm minimizes the training loss. By tailoring and combining perceptrons, the early version of a deep neural network⁷ emerges: the multilayer perceptron (MLP)⁸ [66]. Comprising multiple perceptron layers utilizing non-linear activation functions, MLP's advantage over simple perceptrons lies in its ability to handle non-linearly separable problems through the interplay of hidden layers and non-linear activation functions.

But how can the sequence of tensor algebra, as observed in deep learning, acquire a grasp of intricate feature representation? To address this question, a topological interpretation of deep learning is introduced. The manifold hypothesis [101] postulates that *the real-world data presented in high dimensional spaces is likely concentrated in the vicinity of non-linear sub-manifolds of much lower dimensionality*. If this proposition holds true, then deep learning approximates suitable representations of low-dimensional, complex subsets within high-dimensional data, allowing for interpolation between two input points [19].

⁵The book [50] introduces more methods, such as reinforcement learning.

⁶The single-layer perceptron corresponds to logistic regression.

⁷The definition of 'deep learning' is somewhat variable. Some sources [46] also include multilayer perceptrons under deep learning. To avert confusion, this thesis escapes from strictly defining the term 'deep learning'.

⁸Also referred to as a fully connected network.

Consequently, deep learning's capacity to generalize during training emerges.

In the realm of generalization, there exists a balance to be struck between the two directions⁹ — generalization and overfitting — referred to as the bias-variance tradeoff¹⁰. Finding the optimum equilibrium between underfitting and overfitting through hyperparameter tuning for deep learning architectures remains an open challenge. Although no universal approach exists, various techniques are discussed in [109].

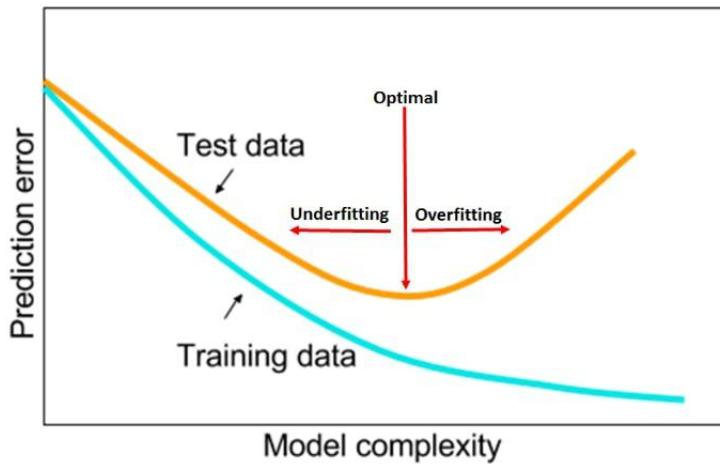


Figure 6.3: This visual representation [109] clarifies the concepts of overfitting and underfitting. In the overfitting regime, increasing model complexity leads to a rise in prediction error for test data, while in the underfitting regime, the reverse occurs. Achieving a balance between these two states hinges on the careful adjustment of hyperparameters¹¹. The paper [109] introduces a systematic methodology for fine-tuning hyperparameters in neural networks.

6.2 Structure of Deep Neural Networks

In the previous section, we discussed the basics idea of deep neural network. In this section, we will explore the algorithmic execution of this process within the context of a regression problem. To aid our understanding, let's refer to Figure 6.4, which illustrates the training process of neural networks.

The training process involves several key steps, each contributing to the network's ability to learn complex relationships and patterns:

- **Input Transformation:** The input data, denoted as X , undergoes an initial transformation. It is first subjected to multiplication by weights w , and the resulting

⁹Underfitting signifies inadequate generalization of input-output relations by the model.

¹⁰Machine learning's overarching task is to minimize the loss function's expectation value's global minimum, which engenders the issue of underfitting and overfitting in machine learning.

products are summed. Additionally, a bias term b is associated with each neuron. This bias term introduces a constant value to the weighted sum, ensuring that the network can still learn even if input values are all zeros. The bias plays a crucial role in activating neurons effectively.

- **Non-Linear Activation:** The linear combination obtained in the previous step is then passed through a non-linear activation function $\sigma(x)$. This function enables the network to capture intricate patterns and relationships by introducing non-linearity into the calculations. It's important to note that in specific network architectures like Convolutional Neural Networks or Graph Convolutional Neural Networks, additional operations such as convolutions or graph convolutions may occur before the non-linear activation.
- **Layer Outputs:** The output of the activation function, denoted as Z , constitutes the output for that specific layer. This process of input transformation and activation is repeated across multiple layers, typically N times. The layering structure allows the network to gradually extract and learn complex features from the input data.
- **Loss Computation:** Once the network produces an output, it calculates a loss function that quantifies the discrepancy between the network's prediction \hat{Y} and the actual label Y . This loss function serves as a guide for optimization, indicating how well the network is performing.
- **Weight Adjustment:** With the loss value computed, an optimizer comes into play. The optimizer employs an optimization algorithm to adjust the weights within the network. Through an iterative process, the optimizer fine-tunes these weights to minimize the loss function and enhance the network's predictive capabilities.

The overarching goal of this training process is to minimize the loss function, thereby improving the network's predictive accuracy. This iterative cycle of input transformation, non-linear activations, loss calculations, and weight adjustments equips the deep neural network with the ability to learn intricate data patterns. By adaptively tuning its weights and biases, the network becomes adept at approximating complex relationships present in the data. In the upcoming subsection, we will delve into a more detailed exploration of the model architecture within deep neural networks. This discussion will encompass essential components such as activation functions, loss functions, and optimizers.

6.2.1 Activation Functions

I aim to introduce one of the most important techniques within deep neural networks, namely the non-linear activation function, which holds immense expressive power. Linear activation layers perform Affine transformations from input to output, resulting in a hypothetical space defined solely by linear transformations. This restriction renders the hypothetical space narrow. In contrast, the non-linear activation function capitalizes on

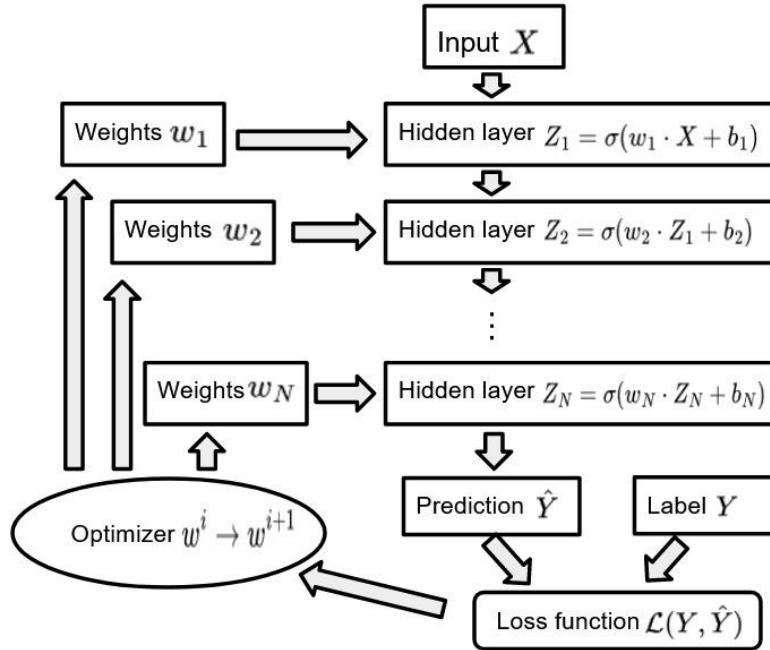


Figure 6.4: This diagram portrays the architecture of deep neural networks. The optimizer iteratively updates the weights from w^i to w^{i+1} through a process of minimizing the loss function.

multiple layers, thereby widening the scope. Generally, non-linear activation functions are kept simple due to computational constraints. Here, I present the activation functions I experimented with throughout the course of this Master's project.

- (1) **Sigmoid:** The sigmoid function gained popularity in the early stages of deep neural network research. It is defined as:

$$f(x) = \frac{1}{1 + e^{-x}} = \frac{\tanh(\frac{x}{2}) + 1}{2} . \quad (6.1)$$

- (2) **ReLU (Rectified Linear Unit):** ReLU is one of the most prevalent activation functions, especially in convolutional neural networks. It can be described as a one-dimensional spline:

$$f(x) = \max(0, x) . \quad (6.2)$$

- (3) **Leaky-ReLU:** Leaky-ReLU was introduced to overcome the problem of ReLU ignoring negative inputs. The parametric-ReLU, with a parameter a , is defined as:

$$f(x) = \begin{cases} x, & \text{if } x > 0 \\ ax, & \text{if } x \leq 0 \end{cases} . \quad (6.3)$$

The specific case where $a = 0.01$ is commonly referred to as Leaky-ReLU

Initially, the ReLU activation function was employed in our GCNN pipeline across all GCNN layers. However, the subsequent replacement of ReLU with parametric ReLU yielded notably improved performance. It's important to note that while this change led to enhanced results, the conclusion is not absolute. Due to computational and temporal constraints, a comprehensive grid search across all possible hyperparameter combinations was not conducted. Since, the selection of the non-linear activation function is intricately linked with other hyperparameters, introducing further complexity to the decision-making process.

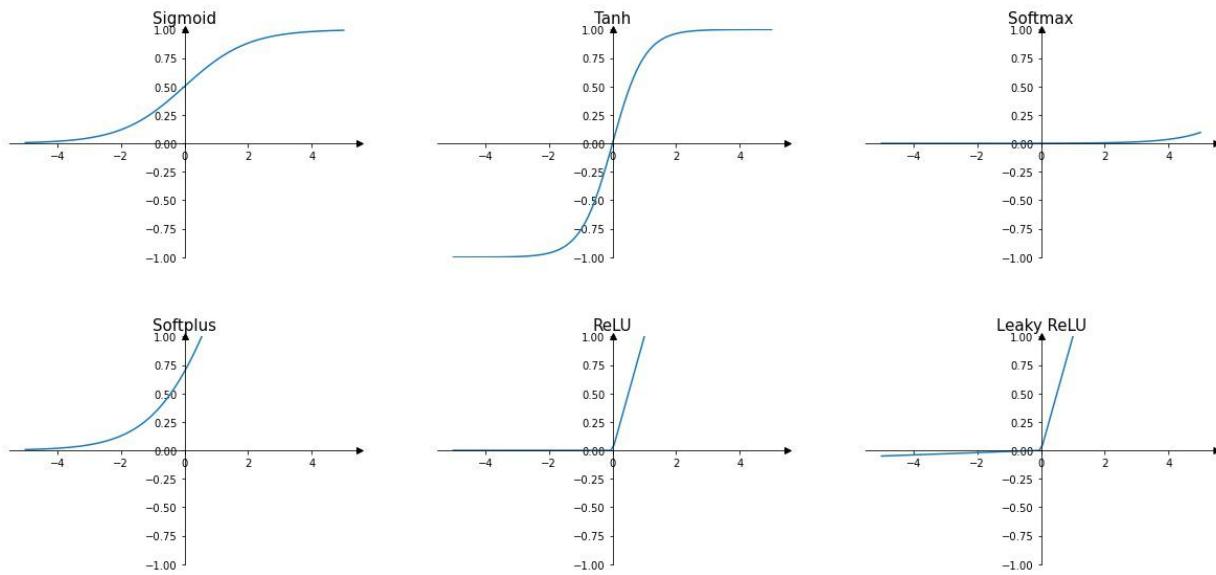


Figure 6.5: Comparison of Different Non-linear Activation Functions in Neural Networks. The activation functions including Sigmoid, Tanh, Softplus, ReLU, and Leaky-ReLU were generated using the Keras library. The Softmax function is employed from the Tensorflow library. Notably, the `keras.layers.LeakyReLU` represents the parametric ReLU. Leaky ReLU can be visualized by setting `alpha=0.01`.

6.2.2 Loss Function

To enable effective training of a machine using supervised learning, the role of the loss function is pivotal. This function serves as a metric for quantifying the discrepancy between the model's predictions and the real data, thereby enabling the model to refine its performance iteratively through the optimization process. In this subsection, I will introduce two distinct functions that were employed throughout the duration of this project.

MSE loss

The Mean Squared Error (MSE) serves as a widely employed loss function in machine

learning, particularly in the context of regression tasks. The MSE quantifies the average squared deviation between the model's predictions, denoted as \hat{y} , and the true observed values, represented as y , within the dataset. In the case of a dataset comprising n data points, the MSE is calculated as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 . \quad (6.4)$$

In this equation, \hat{y}_i signifies the predicted value for the i -th data point, while y_i corresponds to the actual observed value. It is important to note, however, that the use of MSE as a loss function does not guarantee that the trained neural networks will yield an unbiased estimator of the model parameters¹² [38].

NLL loss

An alternative loss function employed in this thesis is the negative loglikelihood (NLL) loss, chosen for distinct reasons. This loss function derives its roots from the concept of maximum likelihood estimation, aiming to ascertain model parameters that optimize the likelihood of observed data.

Expressed mathematically for a dataset encompassing 'n' data points, the negative loglikelihood loss is formulated as:

$$NLL = -\frac{1}{n} \sum_{i=1}^n \log(p_{\text{model}}(y_i|x_i)) . \quad (6.5)$$

Here, $p_{\text{model}}(y_i|x_i)$ signifies the forecasted probability of the correct class label y_i given input data x_i .

Within this master's thesis, the combination of NLL loss and Gaussian likelihood is harnessed for generating posterior distribution plots. This selection is substantiated by two primary considerations. Firstly, the utilization of NLL loss, in conjunction with Gaussian likelihood, facilitates the direct estimation of posterior distributions for cosmological parameters from the output of the GCNN model [37]. In our configured setup, the GCNN model output encompasses not only the estimation of cosmological parameters but also elements contributing to the estimation of covariance matrices. While MSE loss can also yield posterior distributions via methods like kernel density estimation, the NLL loss permits posterior distribution extraction even from single test datasets—rendering outcomes more applicable to real-world weak lensing data analysis¹³.

The second rationale underpinning the adoption of NLL loss pertains to bias mitigation within our inference pipeline. Throughout the experiments, a notable trend emerged, that our inference setup displayed a predisposition towards biased predictions when employing the MSE loss function. Yet, leveraging the NLL loss, the prevalence of biases lessened. This could be attributed to the machine learning process assimilating not only point estimates but also predictive errors, which potentially contributes to the mitigation of biases.

¹²Due to page limitations in this thesis, we have chosen not to include any results obtained using the MSE loss function.

¹³Since our observable universe is only one.

6.2.3 Optimizer

Having calculated the loss function, the next step is to facilitate the learning process of the machine. One prevalent approach is achieved through updating algorithms based on gradient-based optimization. In this algorithm, the weights in the upcoming iteration (w^i) are adjusted using the gradient of the loss function (\mathcal{L}) and the current weights (w^{i-1}) as follows[50]:

$$w^i = w^{i-1} - \alpha \frac{\partial \mathcal{L}}{\partial w^i}. \quad (6.6)$$

Here, α signifies the learning rate which is a crucial parameter that governs the step size of weight updates. A well-chosen learning rate enables efficient convergence while avoiding overshooting or slow convergence.

While gradient descent is a fundamental optimization technique, its direct application is often not ideal due to challenges like convergence speed and avoiding local minima. To address these, stochastic gradient descent (SGD) emerges as a popular choice. SGD employs a stochastic approximation of the gradient by using random subsets of data in each iteration [40]. It balances exploration and exploitation, making it suitable for complex optimization landscapes.

In this Master's thesis, the optimization process is facilitated using the Adam (Adaptive Moment Estimation) optimizer [72] which is an extension of SGD. Adam adapts learning rates individually for each parameter by maintaining running averages of both the first moment and second moment of the gradients. It incorporates the advantages of two other optimizers, AdaGrad[30] and RMSProp[62] while addressing issues like vanishing and exploding gradients. This capability contributes to more stable convergence during training.

The Adam optimizer is characterized by three primary hyperparameters [72]:

- (i) α : Learning rate.
- (ii) β_1 : Governs the exponential decay rate of first moment estimates.
- (iii) β_2 : Governs the exponential decay rate of second-moment estimates.

The adaptive learning rates of the Adam optimizer accommodate varying gradient magnitudes across parameters, enhancing convergence and stability. It is important to note that while Adam has been employed in this work, the optimal choice of optimizer is often determined through hyperparameter tuning [109]. Currently, the focus has been on using Adam; however, future research endeavors may explore other optimization strategies.

6.3 Convolutional Neural Networks

In this thesis, the Graph Convolutional Neural Network (GCNN) model is used to extract the information of our universe. While the focus of this thesis is on the GCNN, which is an advanced evolution of the Convolutional Neural Network (CNN), it's essential to delve into

the fundamentals of CNN. This approach is taken to establish a clear foundation before delving into the intricacies of the more advanced GCNN.

The image stands as one of humanity's most remarkable inventions. It serves as a medium through which scientific discoveries and personal memories can be documented, information can be preserved, and communication can take place in an intuitive manner. In the realm of astrophysics, the utilization of digital images has been instrumental in the exploration and analysis of celestial or cosmological phenomena. These digital images are represented as grid-like data structures, and it is within this context that CNNs find their significance. CNNs are uniquely suited to process such grid-like data, offering the capability to capture local features or patterns inherent in the input data through the use of convolutional kernels [19]. Additionally, they have the ability to extract spatial hierarchical structures in the data using pooling techniques [19]. This makes CNNs an invaluable tool in the field of astronomy for tasks such as image classification, object detection, and feature extraction from astronomical images [110].

In this section, we discuss first the structure of convolutional neural networks, then I introduce several techniques to avoid overfitting, which can also be applied to GCNN structures.

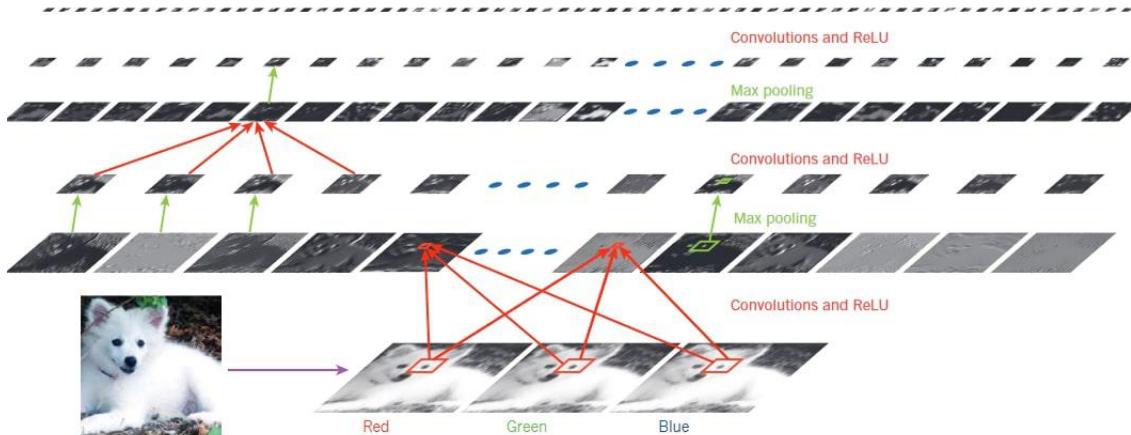


Figure 6.6: This figure illustrates the internal architecture of a standard convolutional network, applied to an RGB image of a Samoyed dog [75].

6.3.1 Structure and Network Layers of CNN

The sequential structure of a CNN plays a pivotal role in its functioning. Each layer's output serves as the input for the subsequent layer, resulting in a cohesive progression through the network. This sub-section provides a step-by-step exposition of the fundamental components within a CNN.

Input layer

The inaugural layer of a CNN is the input data, often an image. This input is represented

as a grid of pixels, conveniently organized as a tensor with dimensions (number of inputs) x (input height) x (input width) x (input channels) [19]¹⁴.

Convolutional layer

The convolutional layer stands as a cornerstone within a CNN. Unlike fully connected layers that capture global patterns, the convolutional layer excels at capturing local features. It employs convolution operations with receptive fields, which are small segments of the input data, and filters, which are learnable parameters typically presented as square matrices (e.g., 3×3 or 5×5). These filters traverse the input data, performing elementwise multiplications and aggregating results to generate feature maps. Subsequently, a non-linear activation function is applied to introduce non-linearity. The process can be summarized as follows [50]:

$$z_{i,j,k}^l = \sigma^l \left(b_k^l + \sum_{u=0}^{f_h-1} \sum_{v=0}^{f_w-1} \sum_{k'=0}^{f_{n'}-1} x_{i',j',k'}^{l-1} \times w_{u,v,k',k} \right) \text{ with } \begin{cases} i' = i \times s_h + u \\ j' = j \times s_w + v \end{cases}. \quad (6.7)$$

In this equation:

1. $z_{i,j,k}^l$ is the output of neuron after introducing the non-linear activation function σ^l at row i , column j in feature map k at layer l .
2. f_h and f_w are the height and width of receptive field, and $f_{n'}$ is the number of feature maps at the layer $l - 1$.
3. s_h and s_v are called the vertical and horizontal stride which controls the step size of the kernel as it slides over the input at the layer $l - 1$.
4. $x_{i',j',k'}^{l-1}$ is the input of layer l and the output of layer $l - 1$ at row i' , column j' , and the feature map k' .
5. b_k is the bias term of feature map k at layer l and σ^l is non linear activation function which you choose for layer l .
6. $w_{u,v,k',k}$ is the weight between any neuron in feature map k at layer l and input which is located at row u , column v , and feature map k' .

Pooling layer

The pooling layer, is essential for managing computational complexity and avoiding overfitting, downsamples feature maps, and reducing the spatial dimensions. For instance, Max Pooling retains the maximum value within each pooling region.

¹⁴For instance, an RGB image has three channels, corresponding to the red, green, and blue color components [19].

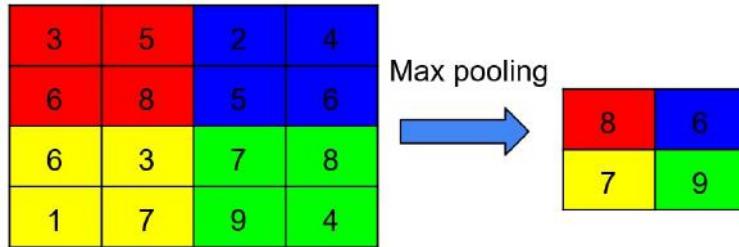


Figure 6.7: Illustration of a max pooling layer with a 2×2 kernel.

Fully connected layer

Following convolutional and pooling layers, fully connected layers¹⁵ are often incorporated. These layers amalgamate abstract features learned from earlier stages to form higher-level representations, which are then evaluated with a loss function. Then I would like to introduce the two interesting features of convolutional neural networks.

- (1) CNN is translational invariance. It means CNNs can recognize patterns or features in an image regardless of their precise location. Due to this symmetry, CNNs require fewer samples to generalize in general with respect to multilayer perceptron which doesn't have this property.
- (2) CNN can learn the spatial hierarchy of input data. For example, the first convolutional layer can learn the small local pattern such as edge, and the second convolutional layer learns the larger pattern of the input layer.

6.3.2 Regularization of CNNs

Addressing the challenge of overfitting is a vital concern in the realm of deep neural networks, including CNNs. To counter this issue, various regularization methods have been devised to curb overfitting, foster generalization, and enhance a network's performance on unseen data.

Dropout

A renowned regularization technique, dropout, involves randomly deactivating neurons during each training epoch. This dropout layer's mechanism nullifies specific neurons to prevent them from heavily relying on particular features, thereby encouraging the network to learn more robust and diversified representations.

Kernel Regularization

Kernel regularization entails appending a penalty term to the loss function based on the weights. L2 regularization introduces the square root of the weight magnitudes as the penalty function, while L1 regularization employs the absolute magnitude of the weights as the penalty function.

¹⁵It is same as MLPs.

Data Augmentation

Data augmentation encompasses applying various transformations to the training data, such as random rotations, flips, and shifts. These transformations aim to equip the network with the capability to recognize features that remain invariant despite such alterations.

Early Stopping

In early stopping, a fraction of the training data, usually around 20% [19], is reserved as validation data for loss cross-validation in each training epoch. This method halts the training process when the validation loss begins to rise, thereby forestalling overfitting.

Batch Normalization

Batch normalization¹⁶ standardizes the inputs of each layer before activation, ensuring a mean of zero and a variance of one across a mini-batch of data. This recentering and rescaling operation stabilizes the training process of artificial neural networks [64].

In this thesis, the regularization techniques¹⁷ of dropout, kernel regularization, early stopping, and batch normalization are put to the test within the **DeepSphere** GCNN architecture, in combination with the MSE loss. However, barring batch normalization and early stopping, substantial enhancements were not observed. Regarding the **DeepSphere** model employing the NLL loss, the existing challenge centers on generalization. Due to computational limitations, training beyond overfitting is infeasible. As a result, regularization techniques, except for early stopping, have not been implemented.

6.4 Graph Convolutional Neural Networks

In contrast to CNNs designed for typical Euclidean domains, such as images, cosmological data is often presented in spherical formats, such as the **HEALPix** method [47]. This master's thesis revolves around working with **HEALPix** maps. However, implementing CNNs in non-Euclidean domains like the sphere is not straightforward. One approach involves using local patches of the full sky, treating the manifold as locally Euclidean by projecting a local region into 2D and applying CNNs. Yet, this method becomes less optimal when dealing with larger observed sky areas¹⁸.

This thesis takes a different approach. It tackles the problem by embracing the spherical nature of the data and defining convolutions on a graph. The central technique employed here is the Graph Convolutional Neural Network (GCNN), specifically the "**DeepSphere**" architecture [27], [96], tailored for spherical data like **HEALPix** maps [47]. This GCNN

¹⁶The term "normalization" in batch normalization refers to a linear transformation that is applied and does not imply any connection with the normal distribution.

¹⁷The original **DeepSphere** architecture's Chebyshev layer does not include the capability for kernel regularization. Nonetheless, you can enable batch normalization by setting the parameter *use.bn* to True.

¹⁸DES Y3 approximately covers about $\frac{1}{8}$ th of the full sky [28], while Euclid is projected to cover approximately $\frac{9}{25}$ of the entire sky [9].

is utilized to directly derive the posterior distribution of cosmological parameters from HEALPix maps [47].

This section serves as an introduction to the fundamental concepts necessary for comprehending GCNNs and the specific "DeepSphere" network employed in this study. For readers unfamiliar with spectral graph theory, Appendix A of this thesis or the review by Hammond et al. (2009) [57] is recommended reading.

6.4.1 Graph Convolutional Neural Networks

Before delving into the details of GCNN, it's crucial to elucidate the distinctions between images and graphs as illustrated in Figure 6.8. Unlike images, where pixels have a specific order and grid-like arrangement, graphs lack a predefined order for each node. Moreover, the number of connected nodes in a graph can differ significantly. This structural difference highlights the unique challenges and considerations when designing neural networks for graph data compared to the more structured world of images. Now, let's delve into

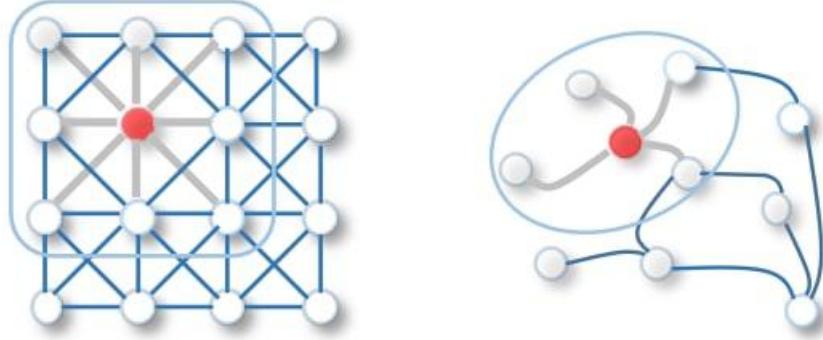


Figure 6.8: This illustration [123] provides a visual comparison between images. On the left-hand side, each pixel is depicted as a vertex in an image. In contrast, the graph on the right-hand side represents a different scenario. In a graph, the neighbors of a vertex lack a specific order, and the number of connected neighbors can vary greatly for each vertex.

the concept of graph convolutional neural networks (GCNNs). To begin, we explore the convolution operation between the input vector x and the filter g , employing the Parseval theorem on graphs (A.13).

$$x * g = \mathcal{F}^{-1}(\mathcal{F}(x * g)) = \mathcal{F}^{-1}(\mathcal{F}(x) \odot \mathcal{F}(g)) = U(U^T g \odot U^T x) . \quad (6.8)$$

ext, we employ the Hadamard product denoted by \odot . We then proceed to define the filter in the spectral domain as $g_w = U^T g$, which leads to the following expression:

$$x * g = U g_w U^T x . \quad (6.9)$$

Let's proceed with deriving the graph convolutional neural network, analogous to the formula (6.7). We consider the input vector x^k , which has dimensions of $|\Omega| \times f_{k-1}$, where

$|\Omega|$ represents the index set of the weighted graph, and f_{k-1} is the number of filters. The output x^{k+1} is also of dimensions $|\Omega| \times f_k$. The convolutional layer of the graph convolutional neural network, without subsampling, can be expressed as follows [16]:,

$$x_j^{k+1} = \sigma \left(U \sum_{i=1}^{f_k-1} F_{k,i,j} U^T x_{k,i} \right) \text{ with } (j = 1, \dots, f_k) . \quad (6.10)$$

Here, σ signifies the non-linear activation function, and $F_{k,i,j}$ represents a diagonal matrix. However, this representation faces a challenge: most graphs possess meaningful eigenvectors primarily at the top of the spectrum [16]. Furthermore, this formulation involves convolving the entire graph, which neglects local structural information. Addressing this, a solution necessitates subsampling and the definition of local filters.

Distinct from CNNs, establishing localized filters for GCNNs is not straightforward, mainly due to the absence of a general fast Fourier transform expression on graphs. A feasible approach to defining local filters is by utilizing a polynomial approximation. In this approach, the filter from equation (6.9) is approximated as a polynomial in terms of the eigenvalues of the graph Laplacian Λ [26]:

$$g_w(\Lambda) \approx \sum_{k=0}^{K-1} \theta_k \Lambda^k . \quad (6.11)$$

Here, $\theta \in \mathbb{R}^K$ represents a vector of polynomial coefficients. This expression yields a K th-order polynomial Laplacian filter that is K -localized in the spectral domain [26]. However, it's important to note that this filter still suffers from a computational cost of $\mathcal{O}(n^2)$ due to the multiplication involving the Fourier basis U , given that $x * g = U g_w U^T x$.

6.4.2 ChebNet

An alternative approach to defining localized filters involves leveraging wavelets. Wavelet transformation can be seen as a localized counterpart to Fourier transformation. In the work by Hammond et al. (2009) [57], the scaled wavelet operator is defined on the graph, and computation using Chebyshev polynomials to approximate the wavelet kernel is demonstrated. In this methodology, the spectral filter is approximated as follows [26]:

$$g_w(\Lambda) \approx \sum_{k=0}^{K-1} \theta_k T_k(\Lambda) . \quad (6.12)$$

Here, T_k represents the Chebyshev polynomial of order k . The Chebyshev polynomial $T_k(x)$ is defined recursively as follows:

$$T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x) , \quad (6.13)$$

with initial conditions $T_0 = 1$ and $T_1 = x$. By employing the spectral filter using Chebyshev Polynomials, expression (6.10) can be streamlined as follows [26]:

$$x^{k+1}j = \sum_{i=1}^{f_k-1} g_{\theta_{i,j}}(\Lambda)x_i^k \text{ with } (j = 1, \dots, f_k) . \quad (6.14)$$

In this context, $\theta_{i,j} \in \mathbb{R}^K$ denotes the layer's trainable parameters, specifically Chebyshev coefficients. Notably, this approach reduces the computational cost of filter calculation to $\mathcal{O}(N)$, as the spectral filter is computed recursively [26][57]. This GCNN layer is referred to as **ChebNet**, and within the architecture of **DeepSphere**, **ChebNet** is utilized for the computation of the Chebyshev layer, which corresponds to the conventional convolutional layer in CNNs.

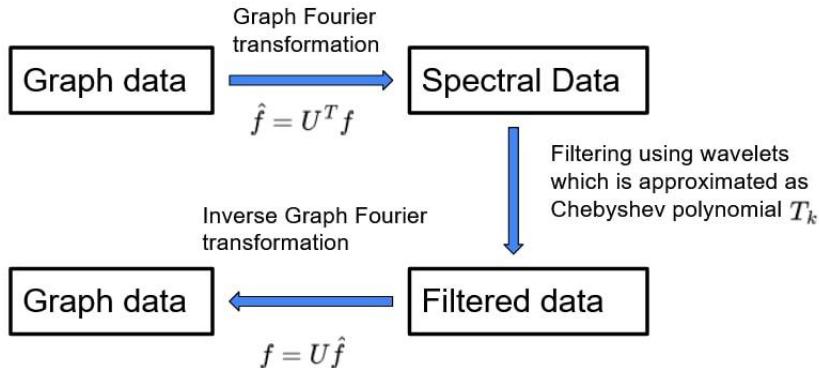


Figure 6.9: This illustration visualizes the concept of filter localization in graph convolutional neural networks. Designing a localized filter within the spectral domain of the graph presents challenges due to the non-local nature of the Fourier transformation. As a solution, wavelet transformation is introduced to establish a localized filter, and this kernel is further approximated through a Chebyshev polynomial in the computational framework of **ChebNet**.

6.4.3 DeepSphere

In this Master's thesis, we leverage the inference pipeline with components from the **DeepSphere** architecture [27], [96]. **DeepSphere** is a distinctive graph convolutional neural network tailored for spherical data representation, specifically designed to accommodate formats like **HEALPix** maps [47]. A distinctive aspect of **DeepSphere** is its ability to maintain rotational equivariance, a feature that sets it apart from conventional convolutional neural networks. In this section, we will delve into the fundamental attributes of **DeepSphere**.

The foundation of **DeepSphere** involves the construction of a graph from spherical data, exemplified by **HEALPix** sampling [47]. To illustrate this process, consider a map with N_{pix} pixels. **DeepSphere** constructs a weighted undirected graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathbf{W})$, wherein each

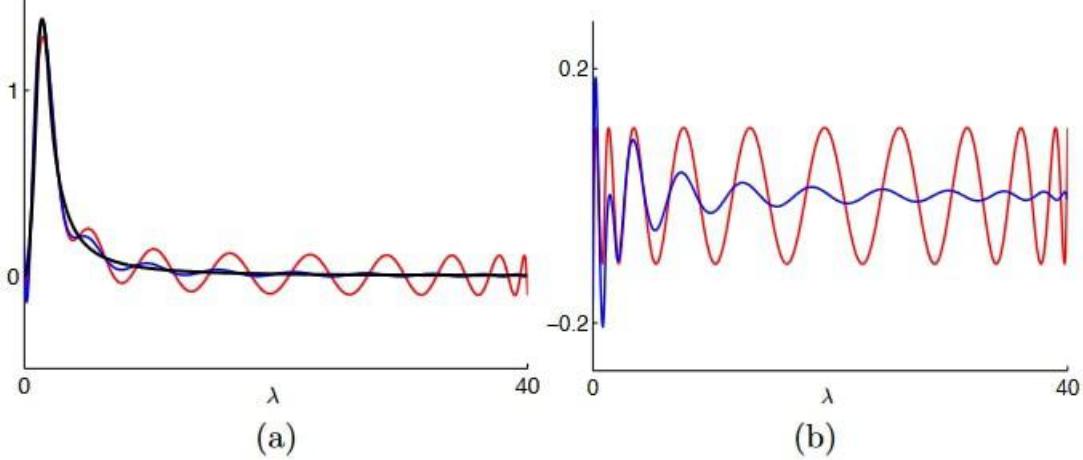


Figure 6.10: (a) The wavelet kernel $g(\lambda)$, along with the truncated Chebyshev approximation (blue) and the minimax polynomial approximation (red). The caption and figure have been adapted from Hammond et al.'s work [57].

pixel i corresponds to a vertex $v_i \in \mathcal{V}(|\mathcal{V}| = N_{\text{pix}})$ [96]. The connections between vertices are established such that each vertex v_i is linked to 8 or 7 neighboring vertices, leading to edges $(v_i, v_j) \in \mathcal{E}$ [96]. These edges are equipped with weights defined by a weighted adjacency matrix $W \in \mathbb{R}^{N_{\text{pix}} \times N_{\text{pix}}}$ [96]:

$$W_{ij} = \begin{cases} -\exp\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\rho^2}\right) & \text{if pixels } i \text{ and } j \text{ are neighbors,} \\ 0 & \text{otherwise,} \end{cases} \quad (6.15)$$

where \mathbf{x}_i is a vector representing the 3D coordinates of pixel i , and ρ is computed as:

$$\rho = \frac{1}{|\mathcal{E}|} \sum_{(v_i, v_j) \in \mathcal{E}} \|\mathbf{x}_i - \mathbf{x}_j\|^2. \quad (6.16)$$

Although there is no general formal proof demonstrating that this adjacency matrix leads to the convergence of the graph Laplacian to the Laplace-Beltrami operator, empirical evidence has shown that this approach indeed achieves such convergence [26].

Now, let's delve into the network layers within **DeepSphere**, which bear resemblance to the conventional convolutional and pooling layers found in typical CNNs. Within the framework of **DeepSphere**, a pivotal component is the Chebyshev layer which is a layer we previously elucidated using equation (6.12), drawing a parallel to the functionality of a conventional convolutional layer. The parameter k , determining the order of the Chebyshev polynomial, is user-defined and corresponds to the number of k -nearest neighbors considered [27]. Adjusting the value of k provides a trade-off [27]: as k increases, the calculations more faithfully uphold rotational equivalence, yet this comes at the expense of heightened memory consumption and slower computations. Consequently, k becomes a noteworthy hyperparameter within the context of our inference pipeline.

In the context of DeepSphere, the pooling layer plays a vital role in facilitating hierarchical sampling, a process illustrated in Figure 6.11 [27]. In the HEALPix scheme [47], the pixel division entails the subdivision of a pixel into four child sub-pixels. During pooling, a choice is made between either capturing the maximum value among these sub-pixels or computing their average. In this thesis, the approach adopted is max-pooling.

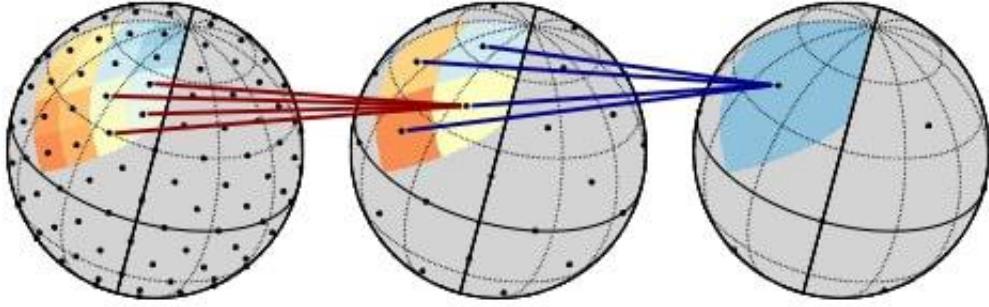


Figure 6.11: This figure illustrates the pooling process utilized in DeepSphere [96]. Four cell groups are merged into a single cell using either Max or Average pooling techniques.

Unlike traditional CNNs, GCNNs are inherently designed to be rotationally equivariant, implying that the graph Laplacian \mathcal{L} and rotation operator $\mathbf{R}(g)$, where $g \in SO(3)$, commute with each other. However, due to factors such as pixelization and approximations, perfect rotational equivariance may deviate slightly. In the context of DeepSphere, this deviation is quantified using the normalized equivariance error, which is computed for a signal \mathbf{f} and rotation g as follows [27]:

$$E_{\mathcal{L}}(\mathbf{f}, g) = \left(\frac{\|\mathbf{R}_V(g)\mathcal{L}\mathbf{f} - \mathcal{L}\mathbf{R}_V(g)\mathbf{f}\|}{\|\mathcal{L}\mathbf{f}\|} \right)^2. \quad (6.17)$$

In the work by Defferrard et al. (2020) [27], it was observed that decreasing the value of k in the k -th order Chebyshev polynomial also reduces this normalized error.

Another crucial consideration is computational efficiency. Perraudeau et al. (2019) [96] demonstrated the theoretical computational cost of graph filtering is related to the number of pixels N_{pix} with a complexity of $\mathcal{O}(N_{\text{pix}}^{\frac{3}{2}})$. Both studies emphasize the need to strike a balance between accuracy and computational efficiency.

Chapter 7

Inference Pipeline

In this section, we delve into the inference pipeline employed to generate the results in this thesis. Our primary focus centers on using the benefits of field-based inference methods, specifically through machine learning models. When dealing with data residing in non-Euclidean spaces, conventional CNNs often struggle. Hence, we have turned to GCNNs as our model of choice, leveraging the capabilities of `DeepSphere` [27], [96].

Two GCNN models, based on the `DeepSphere` framework, have been developed in this Master’s thesis. These models share identical hyperparameters and architecture, differing only in their input data. The first, referred to as the "Lognormal model" henceforth, is trained using Lognormal fields. These fields offer a cost-effective yet accurate means of representing the density field of the late-time universe. The second model, known as the "Gaussian model," serves as a robustness check for our approach. It employs Gaussian fields and acts as a benchmark, allowing us to compare the resulting posterior distributions with those derived from Fisher’s analysis.

Both model architectures produce outputs that estimate cosmological parameters and the essential elements for computing estimations of parameter covariance. After training, these architectures are capable of predicting posterior distributions for cosmological parameters, with a specific focus on Ω_m and σ_8 in this thesis. The ensuing section delves into the details of this inference pipeline, encompassing data processing procedures, model architecture specifics, training methodologies, and an evaluation of model performance.

7.1 Data Preprocessing

Our neural network can directly process inputs consisting of convergence maps generated by our simulation pipeline. However, considering the computational demands during the training phase, downsampling becomes a necessary step. Consequently, in this section, we delve into the downsampling methods and the potential effects stemming from this process.

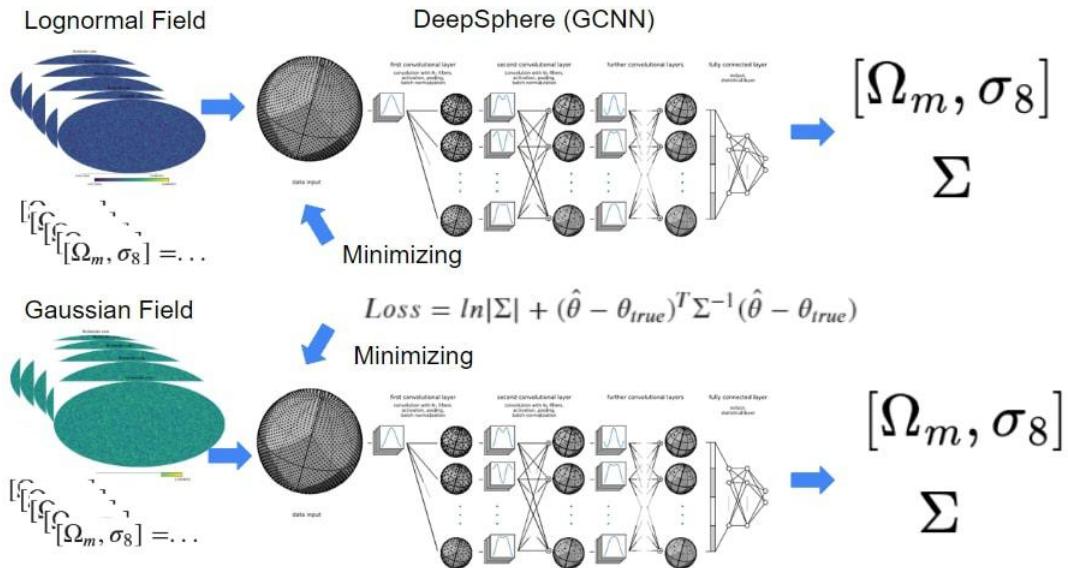


Figure 7.1: This diagram illustrates the inference pipeline utilized in this thesis, employing the **DeepSphere** framework [27], [96]. The neural network undergoes supervised learning and takes input convergence maps \mathcal{M} alongside their corresponding true cosmological parameters θ_{true} . Throughout the training process, the negative-log Gaussian likelihood serves as the loss function. The network's outputs include estimates of cosmological parameters $\theta \in [\Omega_m, \sigma_8]$ and essential elements required for computing the parameter covariance matrix Σ . The figure from [96] is used in this illustration.

7.1.1 Downsampling

First, let's discuss the downsampling process applied to the input data. The output of our simulation pipeline consists of convergence maps with an initial resolution of $NSIDE = 2048$. Here, the $NSIDE$ parameter determines the number of subdivisions used to divide the spherical surface into equal-area pixels [47]. Higher $NSIDE$ values correspond to finer resolutions with smaller pixels, while lower values result in coarser resolutions with larger pixels.

To put this into perspective, the relationship between $NSIDE$ and the number of pixels, denoted as N_{pix} , for a full-sky map is given by:

$$N_{pix} = 12 \cdot NSIDE^2 . \quad (7.1)$$

Using this formula, we can understand the computational demands involved. For instance, convergence maps with $NSIDE = 2048$ contain a staggering 50,331,648 pixels, each holding real-number values. During training, handling tensors for 10,000 maps, each with 50,331,648 pixels, becomes computationally prohibitive for this Master's thesis¹.

¹To gain a deeper understanding of this issue, let's draw a parallel with the process of building an emulator. Imagine you are working on an emulator for the angular power spectrum, and you need to

To address this issue, I employed downsampling, reducing the $NSIDE$ to 128, which results in maps consisting of 196,608 pixels and corresponding to the angular scale 27.5arcmin. This downsampling was performed using the `healpy.pixelfunc.ud_grade` function, which calculates the superpixel's value as the mean of its child pixels. This reduction in resolution helped manage the computational complexity of the project².

7.1.2 Effect of Downsampling

In the previous subsection, we delved into the downsampling process, which, in a way, goes against the core concept of this Master's thesis. The essence of this thesis revolves around the idea of pioneering improved Bayesian parameter inferences compared to traditional 2pt statistics. This improvement hinges on the ability to extract non-Gaussian information hidden on small scales, which conventional 2pt statistics cannot access. Unfortunately, the downsampling procedure, particularly when achieved by averaging subpixels, leads to the degradation of information at these small scales.

How can we quantify this statement, or what should we anticipate regarding the resulting posterior distribution given this angular resolution? We can begin by constructing a normalized histogram, which can be thought of as a 1-point PDF derived from the values observed in a single pixel within an ensemble of size $N = N_{pix}$. To further assess the dissimilarity between probability distributions, we introduce the Kullback-Leibler divergence (KL divergence)³. The KL divergence quantifies the difference between two probability distributions, denoted as $P(x)$ and $Q(x)$, and is expressed as follows [125]:

$$D_{KL}(P\|Q) = \int_{-\infty}^{\infty} dx P(x) \log \left(\frac{P(x)}{Q(x)} \right). \quad (7.2)$$

In essence, when two distributions closely resemble each other, the KL divergence approaches 0, signifying their similarity. Conversely, if the distributions diverge significantly, the KL divergence takes on larger values, indicating substantial differences between them.

Figure 7.2 depicts the dependency of KL divergence of projected density field on the distribution concerning the Gaussian distribution across different angular scales represented by $NSIDE$. This figure conveys that at higher resolutions, there exists more non-Gaussian information. Consequently, using high-resolution maps, it is reasonable to anticipate obtaining a more tightly constrained posterior distribution by using this non-Gaussian information. Nevertheless, for $NSIDE = 128$, the KL divergence is not precisely zero (KL

predict values up to $l = 9000$. If you decide to use 10,000 power spectra to train this emulator, your tensor calculations would revolve around these 10,000 power spectra, each containing 9,000 values. Therefore, field-based inference tasks are generally computationally expensive. Incorporating GPU parallelization is necessary.

²Another crucial aspect of data preprocessing involves the format of HEALPix maps. To ensure compatibility between the `Flask` and `DeepSphere` frameworks, format conversion is required due to their utilization of different HEALPix pixelization schemes, `RING` and `NEST`, respectively, which have distinct pixel ordering patterns [47].

³The KL divergence is not symmetric; therefore, it does not satisfy the properties of a distance metric.

divergence between T17 and Gaussian is 6.59, and lognormal to Gaussian is 3.12 at this resolution).

Our attention now turns to 7.3. Given the asymmetric nature of KL divergence, it is meaningful to make comparisons step by step. First, let's compare the blue curve and the green curve. From this comparison, we can infer that the lognormal realizations from `Flask` are closer to a Gaussian distribution than the T17 convergence maps are to a Gaussian distribution. This holds true across all angular resolutions. Next, we compare the blue curve versus the orange curve. In the cases of $NSIDE = 128$ and $NSIDE = 256$, the lognormal distribution is closer to a Gaussian distribution than the T17 simulation. However, at $NSIDE = 512$, this relationship reverses. Please note that this test neglects the scale dependence of the shifted lognormal distribution, so it is not absolutely accurate.

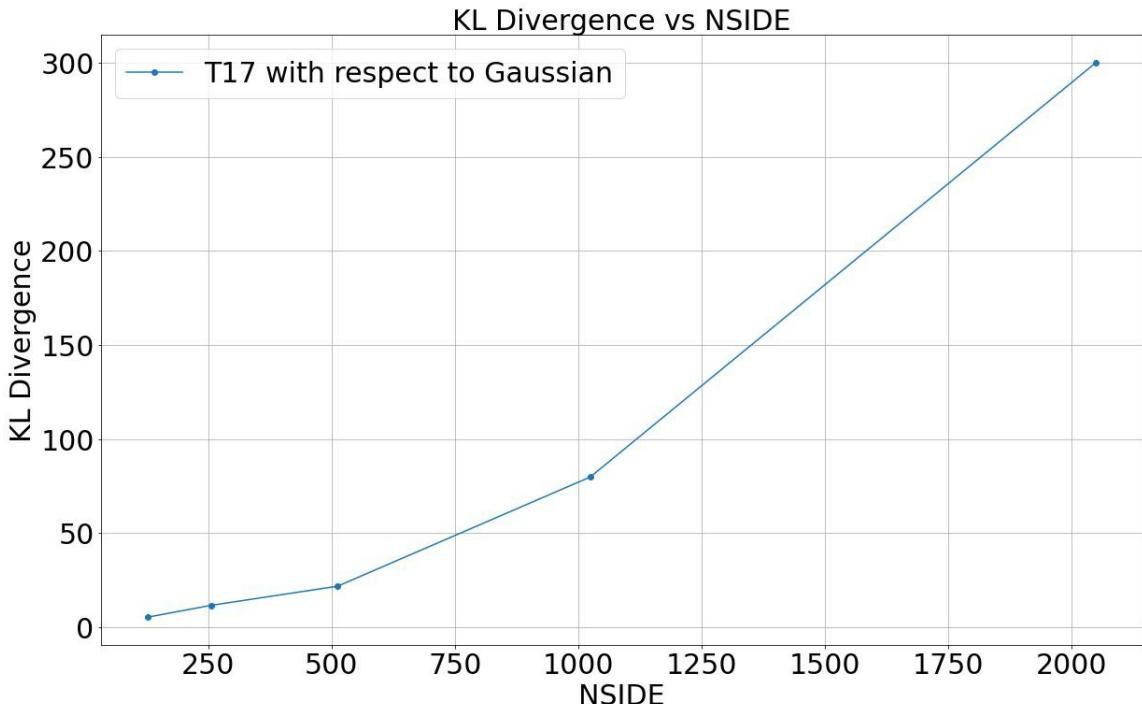


Figure 7.2: The y-axis illustrates the KL divergence concerning the Gaussian distribution, while the x-axis denotes the angular scale presented in $NSIDE$. Gaussian distributions are derived from the simulation pipeline in Chapter 5, employing the WMAP 9-year cosmology [61], consistent with T17 simulations [114].

7.2 Structure of Inference Pipeline

In the following section, we delve into the model architecture of the GCNN models and present derivation of the negative-log Gaussian likelihood loss.

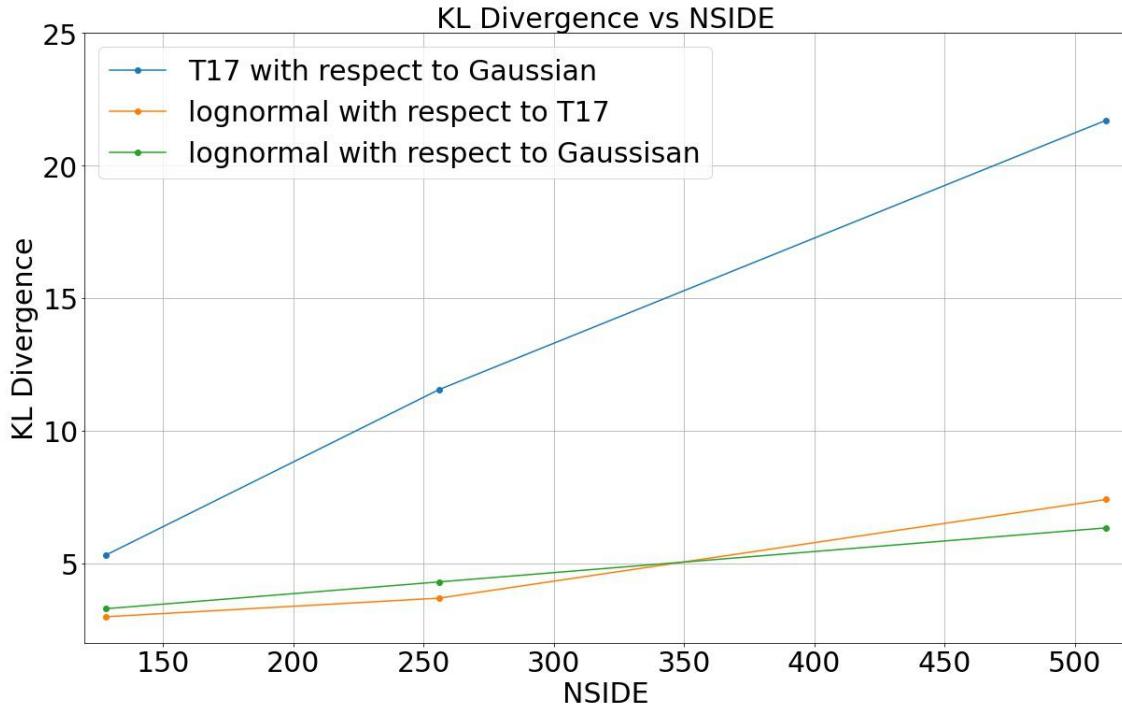


Figure 7.3: The y-axis illustrates the KL divergence concerning the Gaussian distribution, while the x-axis denotes the angular scale presented in $NSIDE$. Both Gaussian and lognormal distributions are derived from the simulation pipeline in Chapter 5 analogous to 7.2. The blue curve indicates the KL divergence of the convergence field from T17 to Gaussian, while the orange curve represents lognormal to T17, and the green curve represents lognormal to Gaussian. Please note that, for plotting the lognormal distribution, a downsampled map from the simulation pipeline is employed. In this approach, the scale dependency of the lognormal shift parameter is disregarded. Consequently, the lognormal curve is less accurate, except for $NSIDE = 128$.

7.2.1 Model Architecture

In this Master’s thesis, I have designed two separate models that utilize identical model architectures and hyperparameters. These neural networks take convergence maps as their primary input, with an angular resolution of $NSIDE = 128$, and generate five real-number outputs. The secondary input comprises cosmological parameters, which serve as labels. Within these outputs, the first two offer estimates for cosmological parameters, while the remaining three provide essential components required for computing the parameter covariance matrix.

The GCNN model architecture is constructed using the `tf.keras.Sequential` method and can be visualized using `tf.keras.model.summary`, as depicted in Figure 7.4. Let’s delve into the model’s components, starting with the initial layer.

The first layer of the model employs the `HealpyChebyshev` layer, a component de-

Layer (type)	Output Shape	Param #
<hr/>		
chebyshev (Chebyshev)	(None, 196608, 10)	50
healpy_pool (HealpyPool)	(None, 49152, 10)	0
chebyshev_1 (Chebyshev)	(None, 49152, 10)	500
healpy_pool_1 (HealpyPool)	(None, 12288, 10)	0
chebyshev_2 (Chebyshev)	(None, 12288, 10)	500
healpy_pool_2 (HealpyPool)	(None, 3072, 10)	0
chebyshev_3 (Chebyshev)	(None, 3072, 10)	500
healpy_pool_3 (HealpyPool)	(None, 768, 10)	0
chebyshev_4 (Chebyshev)	(None, 768, 10)	500
healpy_pool_4 (HealpyPool)	(None, 192, 10)	0
chebyshev_5 (Chebyshev)	(None, 192, 10)	500
healpy_pool_5 (HealpyPool)	(None, 48, 10)	0
chebyshev_6 (Chebyshev)	(None, 48, 10)	500
flatten (Flatten)	(None, 480)	0
dense (Dense)	(None, 5)	2400
<hr/>		
Total params: 5,450		
Trainable params: 5,450		
Non-trainable params: 0		

Figure 7.4: This output is generated using the `keras.model.summary()` function and provides an overview of the architecture of our GCNN model.

rived from DeepSphere [27], [96], serving as an analog to the traditional convolutional layer, as discussed in Chapter 6. This layer is characterized by three crucial parameters. First, K determines the order of the Chebyshev polynomial. A higher K enhances the model’s rotational equivariance but also incurs a greater computational cost. Second, $Fout$ specifies the number of output channels. In the presented model, 10 output channels are employed. You can refer to Figure 7.4 to observe the model’s architecture. The first output is a tensor with dimensions `None` (number of maps) \times 196,608 (corresponding to $NSIDE = 128$) \times 10 (output channels). Finally, the *activation* parameter designates the choice of activation function. Also In this thesis, the parametric ReLU (6.3) is employed, with $\alpha = 0.2$. Although the conventional choice for a convolutional layer’s activation function is the ReLU function, significant performance improvements were observed by utilizing the parametric ReLU, justifying this choice. Subsequent to the

HealpyChebyshev layers, the output undergoes pooling layers. Here, I employ HealpyPool from DeepSphere [27], [96], which identifies maximum values among four sub-pixels, effectively reducing the number of pixels by a factor of 4. This pooling process is applied multiple times in conjunction with the HealpyChebyshev layers. Following the hierarchical application of HealpyChebyshev and HealpyPool layers⁴, the data flows into the `tf.keras.layers.Flatten` layer, a standard component in CNNs. This layer converts the 2D output from previous layers into a 1D vector, preparing it for the final fully connected layer. The final layer, `tf.keras.layers.Dense`, produces five outputs, which are then compared to the true cosmological parameters as labels to compute the loss function. The loss will be discussed in the following subsection.

7.2.2 Negative-Log Gaussian Likelihood Loss

Combining a lot of point estimations allows us to derive the posterior distribution. However, in practice, we can observe only one universe through our observations. The use of neural networks with negative-log Gaussian likelihood loss offers the ability to derive the posterior distribution, even from a single realization. This is very nice because the model provides predictions for the parameter uncertainty, which, in turn, estimates both the parameters and their covariance. This capability serves as the main motivation for employing this loss function in this thesis. Another motivation for using this loss function is its robustness against bias. In this project, I conducted a thorough comparison of bias between Mean Squared Error and this loss functions, with the Negative-Log Gaussian Likelihood Loss yielding superior results⁵. The implementation of the Negative-Log Gaussian Likelihood Loss involves small tricks. In this subsection, I will provide a derivation of this loss function as it is implemented in our code. Let's start with the outputs from the GCNN model. The GCNN models in this thesis provide five outputs denoted as y_0, y_1, y_2, y_3, y_4 . The primary focus of this work centers on estimating two parameters, Ω_m and σ_8 . Therefore, the estimation of these parameters from the GCNN model is expressed as:

$$\hat{\boldsymbol{\theta}} = [y_0, y_2] . \quad (7.3)$$

These GCNN models are trained using supervised learning, and as such, we define the true label $\boldsymbol{\theta}_{\text{true}}$ as:

$$\boldsymbol{\theta}_{\text{true}} = [y_{0,\text{true}}, y_{1,\text{true}}] . \quad (7.4)$$

⁴In this architecture, there are a total of 7 HealpyChebyshev and 6 HealpyPool layers. This configuration leads to $N_{pix} = 48$, which corresponds to the minimum resolution that DeepSphere can effectively compute, equivalent to $NSIDE = 4$.

⁵Despite conducting hyperparameter tuning for the model using Mean Squared Error loss with the assistance of `tf.keras_tuner.BayesianOptimization`, I achieved superior results using the Negative-Log Gaussian Likelihood Loss. It's worth noting that I couldn't perform extensive hyperparameter tuning for the latter due to its long processing time.

Next, we consider the covariance matrix Σ . Since Σ is a Hermitian positive-definite matrix, Cholesky decomposition can be applied [113]:

$$\Sigma = \mathbf{L}\mathbf{L}^T , \quad (7.5)$$

where \mathbf{L} is a lower triangular matrix. Considering that the diagonal elements of the covariance matrix must be positive definite, we employ the exponential function for these elements, resulting in the definition of \mathbf{L} as follows:

$$\mathbf{L} = \begin{pmatrix} e^{y_2} & 0 \\ y_4 & e^{y_3} \end{pmatrix} . \quad (7.6)$$

Now, let's discuss the likelihood assumption. We assume that the likelihood function of the estimated parameters follows a multivariate Gaussian distribution (2.15). In this case, the negative loglikelihood $-\ln\mathcal{L}$ in equation (6.5) can be written as follows:

$$-\ln\mathcal{L} = \frac{1}{2}(\ln|\det(\Sigma)| + (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_{\text{true}})^T \Sigma^{-1}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_{\text{true}})) . \quad (7.7)$$

The objective of the loss function is to minimize this equation (7.7). Thus, the loss is defined as:

$$Loss = \underbrace{\ln|\det(\Sigma)|}_{\log\det\Sigma} + \underbrace{(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_{\text{true}})^T \Sigma^{-1}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_{\text{true}})}_{\chi^2} . \quad (7.8)$$

The second term represents the chi-squared (χ^2) statistic, whose average value should ideally converge to 2 when there are two parameters to estimate. Both the log determinant term and the chi-squared term are monitored in real time during the training process. Substituting the values (7.3), (7.4), (7.6) from the GCNN model into this equation yields the precise loss function implemented:

$$\begin{aligned} Loss = & 2(y_2 + y_3) + (y_0 - y_{0,\text{true}})^2 e^{-2y_2} + (y_1 - y_{1,\text{true}})^2 e^{-2y_3} \\ & + (y_0 - y_{0,\text{true}})^2 y_4^2 e^{-2(y_2+y_3)} - 2(y_1 - y_{1,\text{true}})(y_0 - y_{0,\text{true}}) y_4 e^{-(2y_3+y_2)} . \end{aligned} \quad (7.9)$$

This loss function is employed in training the GCNN models for parameter estimation.

An important aspect to note about this loss function is that it relies on the Gaussian assumption for the likelihood function. This means that the inference is based on the assumption of Gaussianity, and computations of likelihood are performed accordingly. If you wish to perform inference beyond this Gaussian assumption, you can explore likelihood-free-inference (LFI) methods [38], such as those available in libraries like `pydelfi` [4], or consider using normalizing flow-based models. However, such advanced techniques are outside the scope of this thesis.

7.3 Training of GCNN Models

Once the model architecture is established, selecting the right hyperparameters becomes a critical factor in achieving 'optimal' prediction accuracy⁶. Given the characteristics of our chosen loss function and model, training a single model can take approximately 1 to 2 days. Given this time constraint, advanced hyperparameter tuning methods like Optuna[3] are not feasible for this Master's thesis. However, I have developed a basic training strategy that I would like to introduce in this section, along with a visualization of the loss curve.

7.3.1 Training Strategy

To begin the training process, the model must first be compiled. This entails specifying the optimizer, loss function, and evaluation metrics. In this thesis, the widely used Adam optimizer [72] is employed, which is often the popular choice for regression problems in CNNs. Nevertheless, the optimal optimizer may vary depending on the specific problem, and as such, future research could explore alternative options.

Another crucial of defining the optimizer is determining the learning rate. The learning rate controls the step size at which the model's weights are updated during training. Generally, if the learning rate is too large, the training process may overlook the global minimum of the loss function, while if it's too small, training may become time-consuming. This characteristic becomes especially evident when using the Negative-log Gaussian Likelihood Loss compared to Mean Squared Error. If the learning rate is excessively large, the model tends to overfit quickly, leading to suboptimal results. To mitigate this, a very small learning rate and a large number of training epochs (e.g., 3000 epochs) are necessary when employing this loss function.

As previously mentioned, training for 3000 epochs can take approximately 1-2 days, depending on the available GPU resources, making common hyperparameter tuning techniques impractical. Additionally, there's no need to set a small learning rate from the outset; it can be gradually decreased until the loss function reaches a certain minimal threshold value. For these reasons, a decaying learning rate is employed. Specifically, the `tf.keras.optimizers.schedules.PolynomialDecayschedule` is used with the following settings:

- (i) **initial_learning_rate**: This parameter specifies the initial learning rate at the beginning of training. In the final inference pipeline, it is set to 0.0001.
- (ii) **decay_steps**: This parameter determines the number of epochs required for the learning rate to decay from the initial value to the end value. In the final inference pipeline, it is set to 3000.

⁶It is interesting to consider how we define optimal accuracy, given various computational factors such as resource constraints or data quality. However, it is also necessary to establish a definition, as otherwise, we may never complete a single project.

- (iii) **end_learning_rate**: This is the target learning rate to be reached after `decay_steps` epochs. In the final inference pipeline, it is set to 0.00001.
- (iv) **power**: This parameter controls the learning rate's rate of decrease over time. In the final inference pipeline, it is set to 1.0, resulting in linear decay.

As previously mentioned, the chosen loss function in this thesis is the Negative-log Gaussian Likelihood Loss. In addition to this loss function, several metrics are selected to gauge the model's performance during training and evaluation. These metrics include:

- (i) **Loss Function**: The primary loss function itself, which is the Negative-log Gaussian Likelihood Loss. It plays a central role in guiding the training process, ensuring that the model converges towards accurate parameter estimation.
- (ii) **detlog(Σ)**: This metric corresponds to the first term of equation (7.8), and it serves as a key indicator of the quality of the covariance estimate. Monitoring this metric provides insights into the model's ability to estimate the parameter covariance effectively.
- (iii) χ^2 : The χ^2 metric represents the second term of equation (7.8). It measures the agreement between the predicted parameters and the true values. A value of 2 is expected for χ^2 since two parameters are being estimated. Deviations from this value can indicate not only potential issues with the model's accuracy but also problems with the covariance.
- (iv) **MSE**: In addition to the specialized loss function, Mean Squared Error is employed as an auxiliary metric. MSE is a commonly used metric in regression tasks and provides an overall assessment of the model's performance. It serves as a reference point based on previous experiences with tested MSE metrics.

By utilizing this set of metrics, the training and evaluation process can assess various aspects of the model's performance.

The selection of batch size is an important decision in the training process, and larger batch sizes, in particular, can offer some significant advantages. They often lead to more stable gradient estimates, which in turn promote smoother convergence during training and help the model avoid getting trapped in local minima. This aspect becomes even more crucial given the specific characteristics of our chosen model architecture and hyperparameters, which tend to produce unstable gradient estimates. Consequently, despite using identical hyperparameters and input data, I've observed varying prediction accuracy due to these fluctuations. In an ideal scenario, I would prefer to increase the batch size to take advantage of these benefits. However, there's a trade-off, as larger batch sizes also come with increased memory consumption. Currently, I have set the batch size to the maximum feasible value of 16 due to memory constraints. Beyond this limit, memory overflows become a concern, restricting further increases in batch size.

In the final inference pipeline, I employed a dataset consisting of 10,000 convergence maps for training purposes. These maps were generated through a combination of 10

random realizations and 1,000 different cosmological parameter sets. Additionally, for validation and testing, I utilized a dataset of 2,500 convergence maps, formed in a similar fashion with 10 random realizations and 250 different cosmological parameter configurations. It's important to note that both the training and validation datasets were obtained using LHS sampling separately but from the same prior range.

Two distinct models were developed in this context: the Gaussian model and the lognormal model⁷. Both models shared identical architecture and hyperparameters. Furthermore, to assess the error introduced by the GCNN model itself, I created two additional models. These models were identical in terms of whether they employed the lognormal or Gaussian framework but used different input data, generated with different random number seeds for the creation of convergence maps. Additional details regarding the errors of the GCNN models can be found in Appendix B.

The training process involved 3,000 epochs and employed `tf.distribute.MirroredStrategy`, allowing for parallel training across multiple GPUs. Here, I had the choice between using a single NVIDIA A100 GPU or three NVIDIA A40 GPUs. While the NVIDIA A100 is a powerful GPU and performs better individually, a time analysis revealed that utilizing three NVIDIA A40 GPUs allowed for faster training, taking only 33 seconds compared to the 54 seconds required by a single NVIDIA A40 GPU and 44 seconds by a single NVIDIA A100. Therefore, I opted for the strategy of employing three NVIDIA A40 GPUs for training.

7.3.2 Interpreting Loss and Metrics

One approach to validate machine learning methods is by scrutinizing the loss and metrics during the training process. By examining these plots, we can gain insights into whether our model is overfitting or underfitting. In Figure 7.5, we present the loss function alongside the metrics that we monitor during training: Figure 7.6, Figure 7.7, and Figure 7.8.

Let's begin by analyzing the loss function, as depicted in Figure 7.5. We observe a gradual decrease in the loss as the number of epochs increases, indicating that the model is still in the process of generalization, and further training epochs could lead to improved prediction accuracy. However, we also notice a peculiar step-function-like shape in the loss curve. This intriguing pattern might be attributed to several factors.

It could be attributed to the nature of our loss function, which comprises two terms. This complexity may pose a challenge for the optimization algorithm in locating the global minimum. In support of this notion, we observe that the model has never exhibited this peculiar shape when we used MSE as the loss function.

Secondly, the irregular 'step-function' shape might be influenced by the decaying learning rate strategy. Initially, without the learning rate decay, the models seem to learn up to a certain threshold, after which the loss no longer decreases, and overfitting occurs. The use of a decaying learning rate could introduce variability as the machine learns different

⁷In fairness, such comparisons may not be entirely equitable, as different input datasets could inherently require distinct optimal hyperparameters and model architectures.

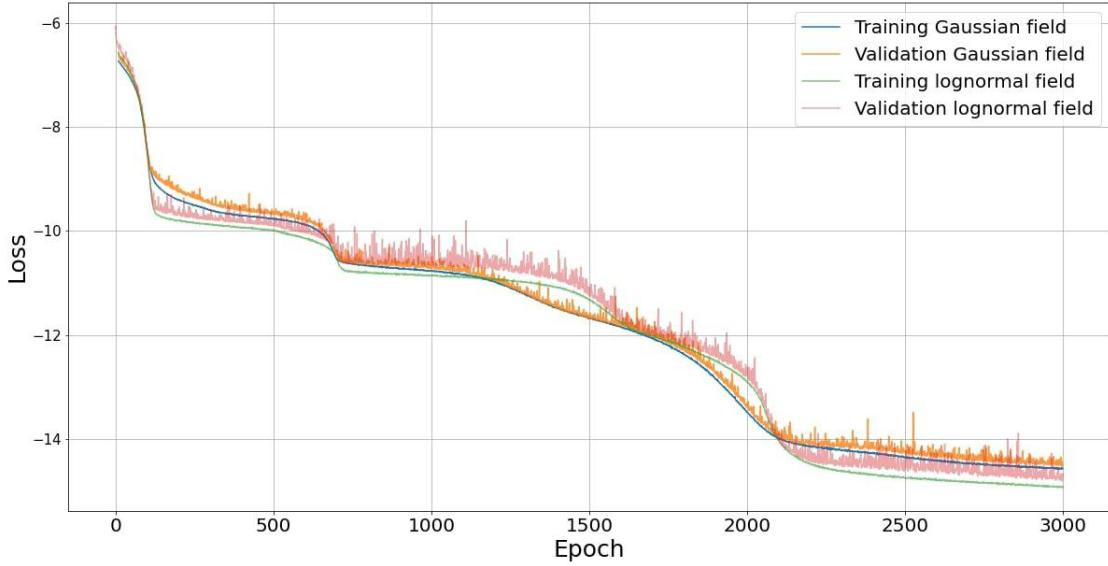


Figure 7.5: This figure shows the evolution of the overall loss in relation to the number of epochs.

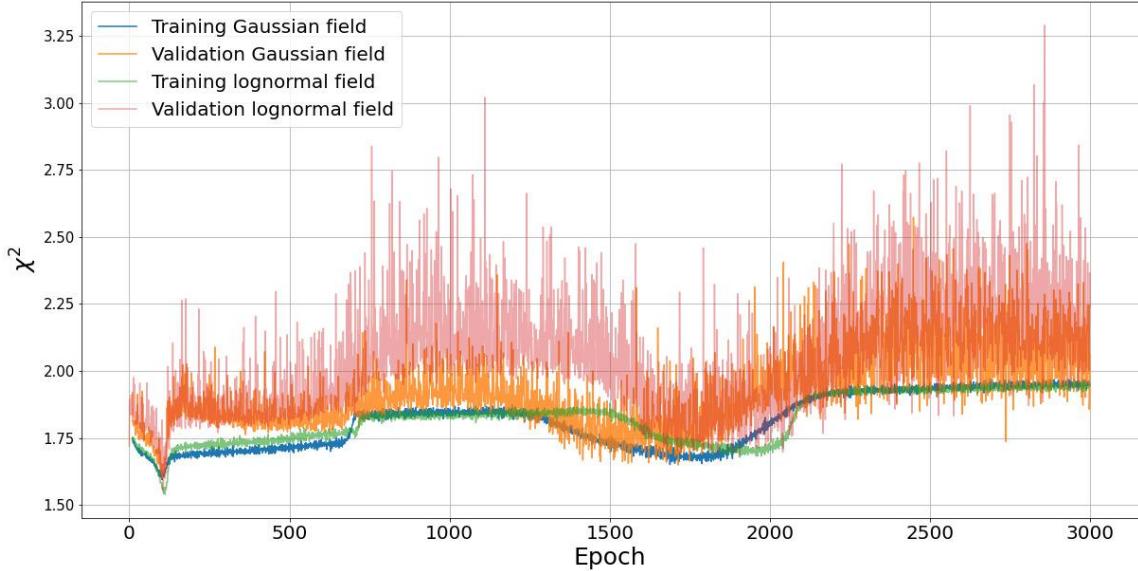


Figure 7.6: This figure shows the evolution of the χ^2 metrics in relation to the number of epochs.

features at various stages of the training process.

Lastly, the inadequate batch size might contribute to the observed instability. Notably, both the lognormal and Gaussian models exhibit distinct loss curve shapes, despite having similar input data characteristics, as shown in Figure 7.2. This discrepancy could be linked to training instability, but a more comprehensive investigation and testing are required to

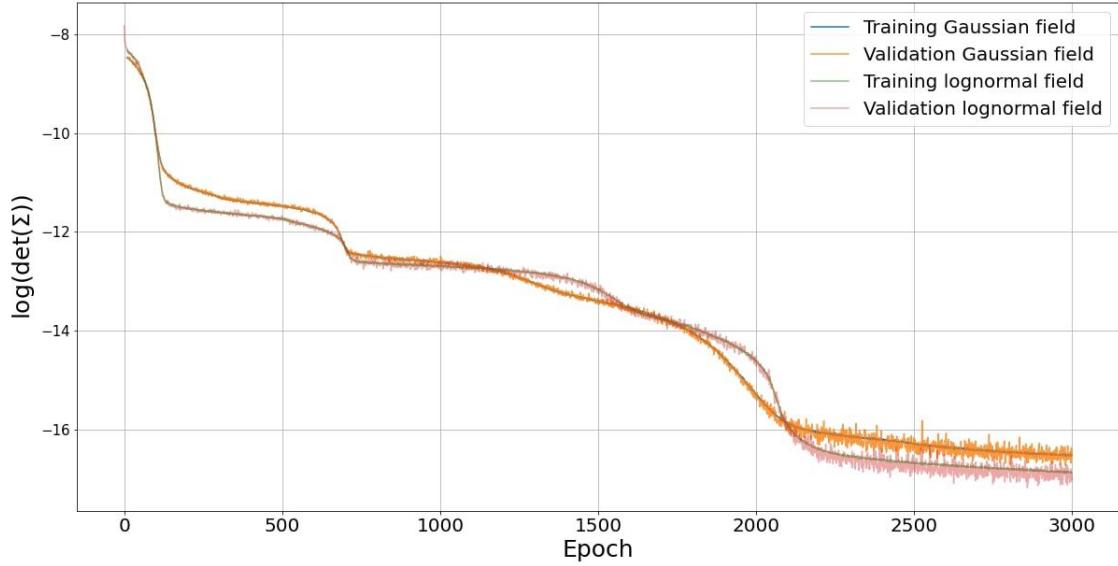


Figure 7.7: This figure shows the evolution of the $\log(\det(\Sigma))$ in relation to the number of epochs.

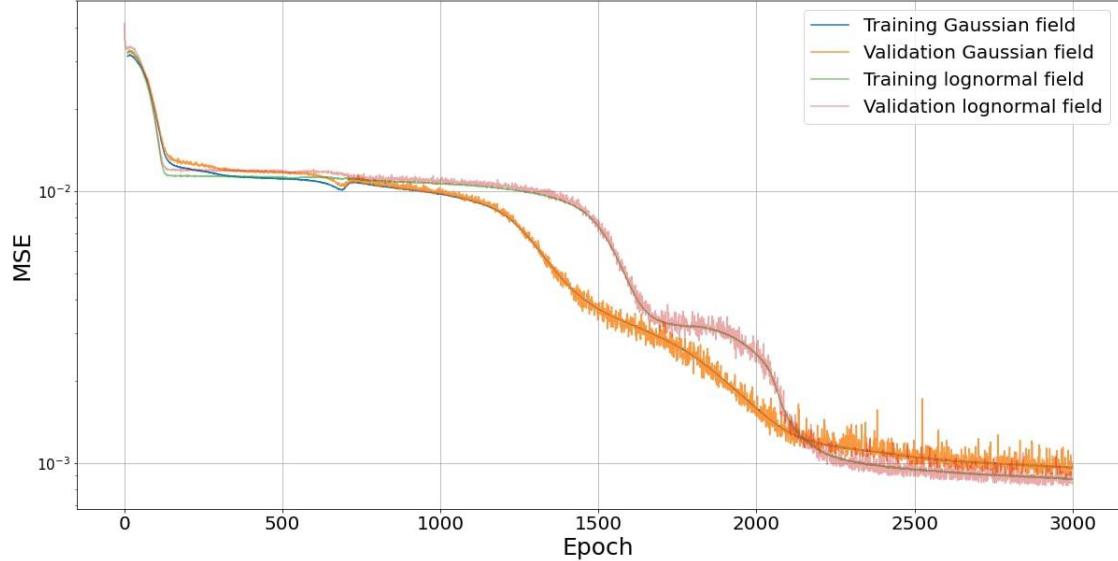


Figure 7.8: This figure shows the evolution of the MSE in relation to the number of epochs.

confirm those hypotheses.

Now, turning to Figure 7.6 and Figure 7.7, we can observe the dynamics of the primary loss components. Notably, the $\log(\det(\Sigma))$ term emerges as the dominant contributor to the loss, and both components gradually converge within a similar number of epochs. Encouragingly, χ^2 approaches a value of 2, signifying a desirable outcome. However, it's worth noting that there is significant fluctuation in χ^2 within the validation datasets.

Finally, we delve into the Mean Squared Error (MSE) metrics depicted in Figure 7.8. Surprisingly, the MSE evolution aligns with that of the overall loss function in Figure 7.5. The MSE values remain within the desired range, typically below 10^{-3} . When the MSE falls below this threshold, we can make the following empirical observation: after training GCNN with maps of a single cosmology and multiple realizations, the testing results with the same type of maps (e.g., training with Gaussian maps and testing with Gaussian maps) exhibit ‘favorable’ outcomes, such as a clustered cloud of points centered around the fiducial cosmology. However, it’s important to note that biases may still be present when testing with a different type of field than what was used for training, as discussed in more detail in Appendix B.

7.4 Validation Test of Inference Pipeline

This Master’s thesis employs another validation approach for evaluating the GCNN. It involves comparing the predicted values generated by the neural networks with the true values in the labels using validation datasets. This method provides insights into the potential biases introduced by the machine learning model.

In Figure 7.9, both Gaussian and lognormal models exhibit a similar level of bias. Notably, a significant bias appears in the upper-right corners of the plot. This bias occurs because machine learning models tend to perform worse at the edges of their training data distribution, as they are primarily trained within the data range provided. To mitigate this effect, it is advisable to have a relatively broad prior that encompasses the region where the test datasets’ fiducial cosmologies are situated. Fortunately, this does not pose a problem in practice, as our fiducial cosmology typically falls within the central region of this plot. Moving on to Figure 7.10 and 7.11, we observe differences in prediction accuracy between the two parameters. The error bars for Ω_m are relatively larger than those for σ_8 , and the accuracy of predictions for Ω_m is notably worse, especially at the edges. There is also some fluctuation in the Ω_m plot within the range of 0.35 to 0.45. To address this imbalance in training both parameters, one possible method is standardizing the labels. This involves subtracting the mean and dividing it by the standard deviation. However, in this thesis, this method is not employed as Ω_m and σ_8 have reasonably similar expected ranges. It is challenging to determine whether standardizing the labels would significantly improve this situation, as there are still areas for potential improvement in the inference pipeline. For instance, the machine is still in the process of generalization, and further increasing the number of epochs or adjusting the learning rates might enhance performance. Additionally, batch normalization could potentially alleviate this issue. Therefore, future research will explore these avenues in greater detail.

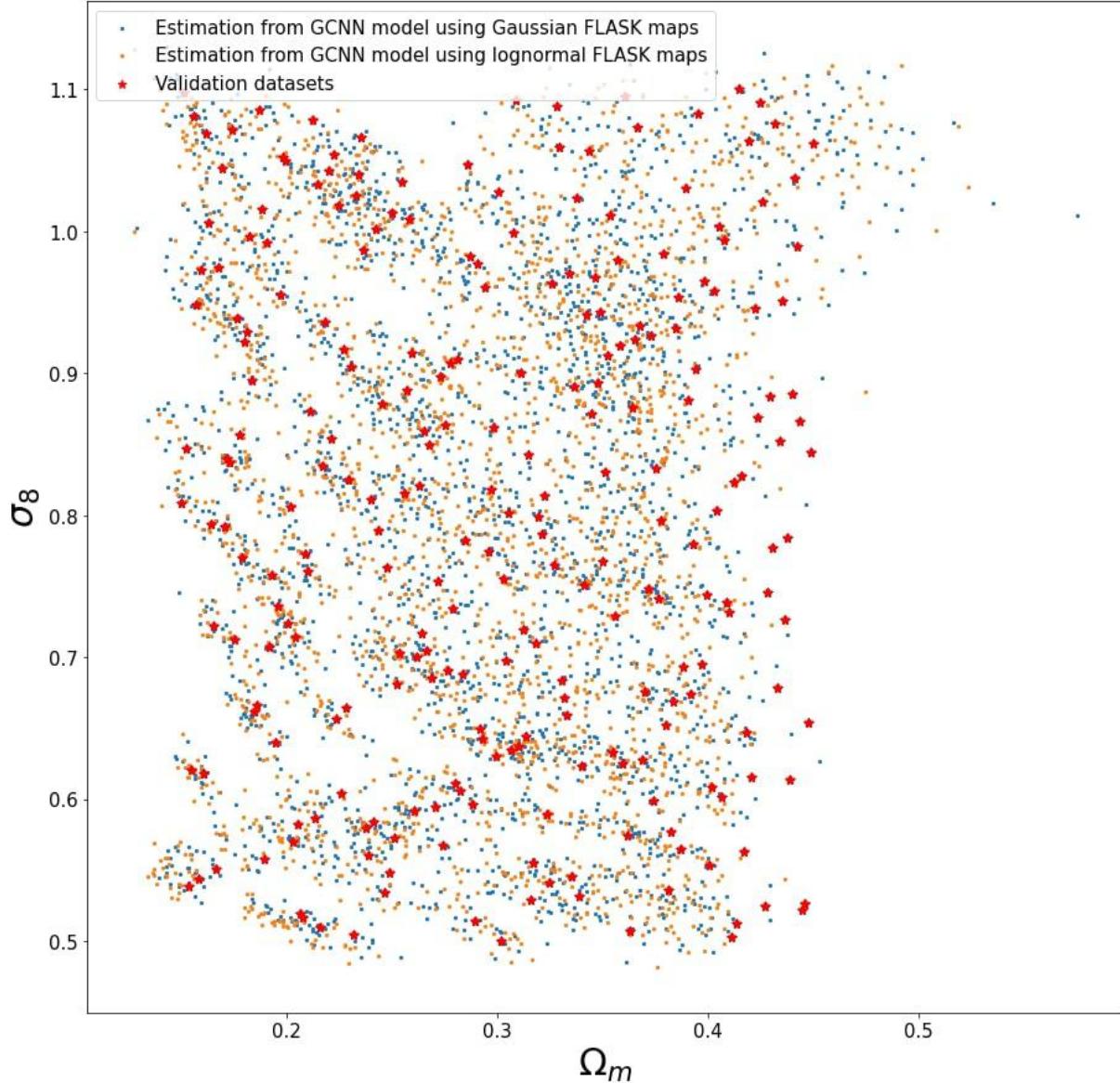


Figure 7.9: This figure showcases a comparison between the actual validation dataset points and the predictions in the 2D Ω_m - σ_8 plane. The red star points represent the validation datasets, with lognormal fields employed for lognormal models and Gaussian fields for Gaussian models. Notably, both datasets are sampled from the same set of cosmologies. The orange points denote predictions generated by lognormal models using lognormal validation datasets, while the blue points correspond to predictions made by Gaussian models.

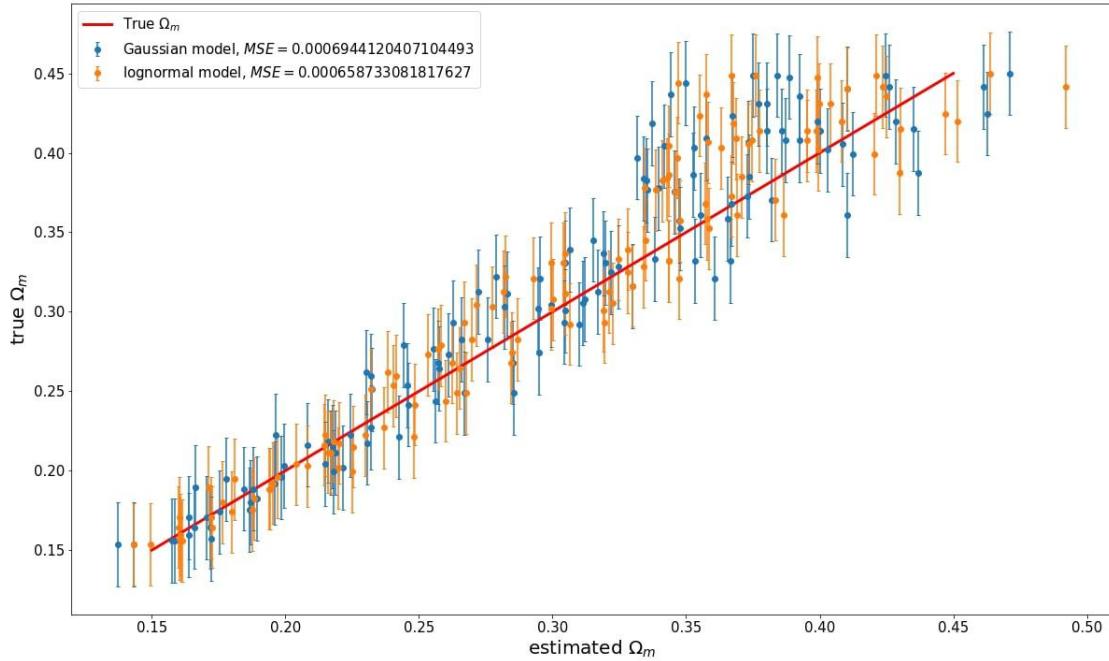


Figure 7.10: In this plot, we compare the predicted values of Ω_m using the validation datasets and true values of Ω_m . The red curves represent the unbiased prediction line, where $y = x$. To reduce the density of data points for better visualization, a step size of 5 was applied. These validation datasets are the same as those used in Figure 7.9. The errors are quantified using the MSE.

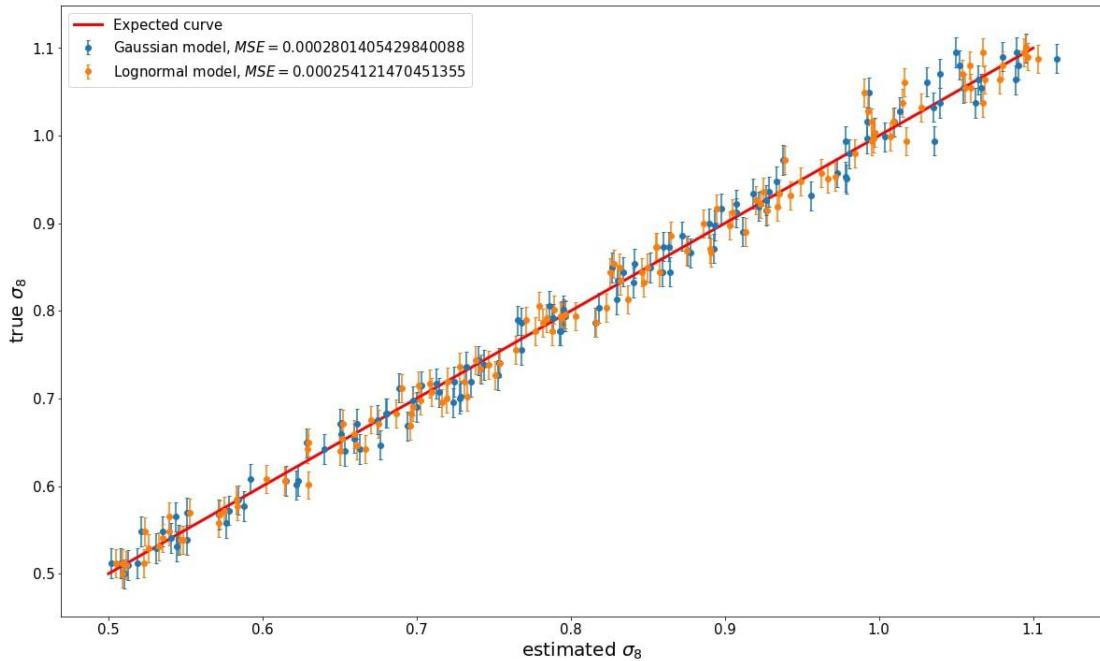


Figure 7.11: Analogous to the caption in Figure 7.10.

Chapter 8

Results and Discussions

The primary objective of this thesis centers around the exploration of GCNN models and their potential to enhance our understanding of the universe by employing them as tools in the sense of better constraining cosmological parameters compared to standard statistical tools such as power spectrum or 2-point correlation functions. As previously discussed in Chapter 2, observational cosmologists are primarily concerned with the posterior distribution of cosmological parameters, and we update their knowledge of the universe in a Bayesian framework¹. However, a question arises: How can we leverage field-based methods, often regarded as black-box approaches, to enhance our understanding of the universe?

To address this question, this thesis introduces two distinct models: the Gaussian model and the Lognormal model, as elaborated in Chapter 7. Additionally, three distinct test datasets are prepared for evaluation: Gaussian fields, lognormal fields, and convergence fields generated from T17 simulations [114]. The lognormal and Gaussian test datasets consist of 300 random realizations, all corresponding to a single WMAP 9-year best-fit cosmology. This setup enables an examination of the constraining capabilities of both models and offers insights into any biases introduced due to inconsistent approximations. The T17 simulations, encompassing 108 random realizations, are based on the same cosmological parameters as the other test datasets, representing a more realistic and challenging scenario.

The robustness analysis of the GCNN models is presented in detail in Appendix B. Throughout this chapter, a comparative analysis is conducted to assess the constraining power and potential biases associated with each combination of models and test datasets.

¹Notably, there are several valuable frequentist statistical methods applicable in cosmology, such as the χ^2 -Test and the Fisher matrix analysis [125]. These techniques also play a significant role in analyzing and interpreting cosmological data.

8.1 Clouds of Points

Before visualizing the posterior distribution, we execute the analysis using point clouds. In this section, our primary focus is on the point clouds generated by two models and across three distinct test datasets. We commence this analysis by quantifying both bias and variance to facilitate a comparison of our model’s accuracy. Subsequently, we proceed to construct the posterior distribution through kernel density estimation.

8.1.1 Cluster of Points

Firstly, we can directly visualize the first two outputs of our GCNN models, which correspond to estimations of cosmological parameters. This visualization can be observed in Figure 8.1 and Figure 8.2. In the Gaussian model, when using the Gaussian test dataset, and in the lognormal model, when using the lognormal test dataset, we observe clusters of points centered around the fiducial cosmology. This clustering indicates an ‘unbiased’ estimator of fiducial parameters. However, when examining the points from the T17 test dataset and the lognormal test dataset in the Gaussian model, we notice a different pattern compared to the lognormal model. In the lognormal model, the T17 test datasets and Gaussian test datasets show less bias than in the Gaussian model, which is contrary to our initial expectations. It’s worth noting that these results may vary depending on the intensity of model training. Further details are discussed in Appendix B.

8.1.2 Bias and Variance

In a manner analogous to Frequentist χ^2 methodology and maximum-likelihood analysis, we proceed to calculate the bias and variance associated with our estimators. Given our assumption of a Gaussian likelihood, the mean value, denoted as $\langle \hat{\theta} \rangle$, corresponds to the maximum likelihood estimate (MLE) and is also the minimum of the χ^2 statistic. The bias, a metric for assessing the quality of an estimator, is defined as the difference between the fiducial value θ_{fid} and the predicted mean $\langle \hat{\theta} \rangle$ in one dimension:

$$\text{Bias} = \theta_{\text{fid}} - \langle \hat{\theta} \rangle . \quad (8.1)$$

The bias quantifies how closely our estimator aligns with the true value. To quantify the spread of our estimator’s predictions, we calculate the variance:

$$\sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (\hat{\theta}_i - \langle \hat{\theta} \rangle)^2 . \quad (8.2)$$

Here, N represents the number of data points. The variance provides a measure of how dispersed the estimator’s values are around the mean estimate. By utilizing the variance, we can construct confidence intervals to characterize the uncertainty associated with our estimations.

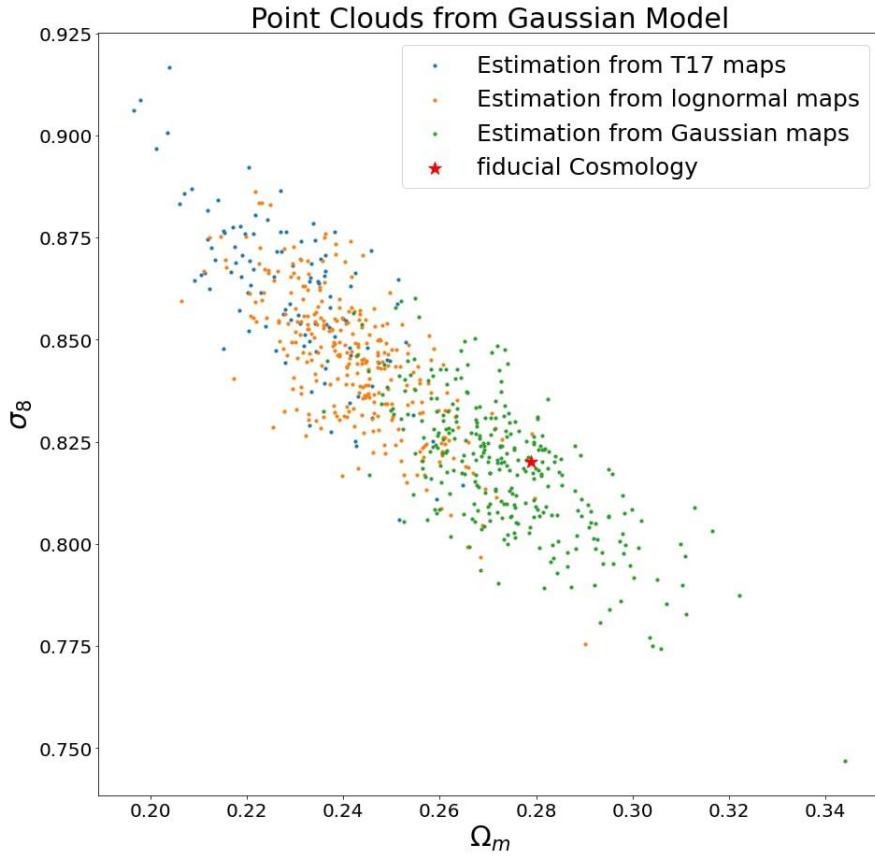


Figure 8.1: This visualization illustrates the predictions generated by the Gaussian model. The red star point corresponds to the fiducial cosmology, which serves as the true value of the label. The blue points represent predictions from 108 random realizations of T17 simulations. The green points represent predictions from 300 random realizations of Gaussian test datasets, while the orange points depict predictions from 300 random realizations of lognormal test datasets.

In Figures 8.3 and 8.4, we examine the coverage of the fiducial cosmology within the $1-\sigma$ confidence interval. Let's begin with Figure 8.3. Notably, most of the combinations of models and test datasets recover the fiducial value within the $1-\sigma$ confidence interval, with the exception of the lognormal test datasets and T17 datasets when using the Gaussian model. Remarkably, the Lognormal model, when tested with both lognormal and Gaussian datasets, and the Gaussian model when tested with Gaussian datasets, exhibit nearly unbiased the σ_8 .

Moving on to Figure 8.4, it's interesting to note that the variance shows relatively consistent behavior between σ_8 and Ω_m . This behavior is in contrast to what we observed in the MSE errorbars in Figures 7.10 and 7.11, where we observed different MSE patterns for these parameters. This discrepancy can be explained by the dominant contribution of bias being in the Ω_m direction, resulting in varying MSE values.

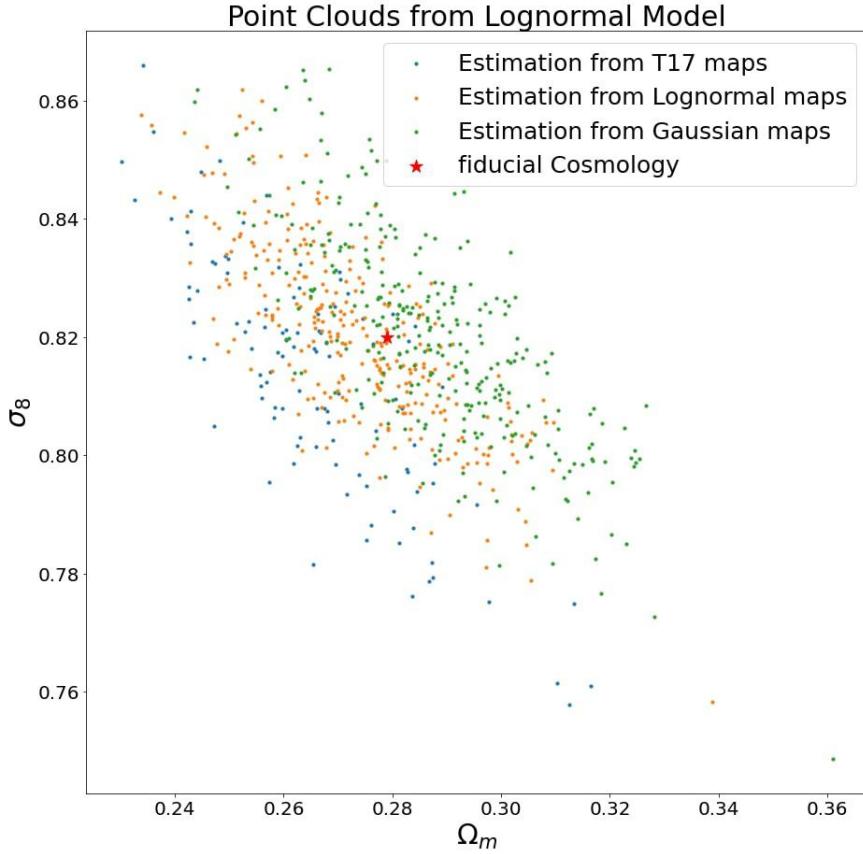


Figure 8.2: This visualization showcases the predictions generated by the lognormal model. The descriptions for each set of points are analogous to those in Figure 8.1.

Furthermore, I have assessed the quality of these estimators by calculating the Euclidean distance between the predicted mean and the fiducial cosmology. This metric is defined as:

$$dis_{\text{Euc}} = \sqrt{(\theta_{\text{fid},\Omega_m} - \langle \hat{\theta}_{\Omega_m} \rangle)^2 + (\theta_{\text{fid},\sigma_8} - \langle \hat{\theta}_{\sigma_8} \rangle)^2}. \quad (8.3)$$

This evaluation is depicted in Figure 8.5. Initially, it's worth noting that both Gaussian and lognormal models yield similar values for the same datasets, ensuring a fair comparison. In the case of the Gaussian model, the T17 dataset exhibits the largest Euclidean distance, followed by the lognormal field dataset. Conversely, the lognormal model proves to be the best estimator for the lognormal field dataset, followed by Gaussian field datasets and convergence maps from the T17 simulation. Interestingly, the T17 dataset yields a higher value when processed with lognormal models compared to Gaussian fields. This discrepancy may be attributed to the inclusion of higher-order moments in our training datasets, introducing complexity. To validate the accuracy of the lognormal approximation independently, further tests could involve generating the lognormal test dataset using measured C_l and λ from the measured values of the T17 simulation. It is noteworthy that even

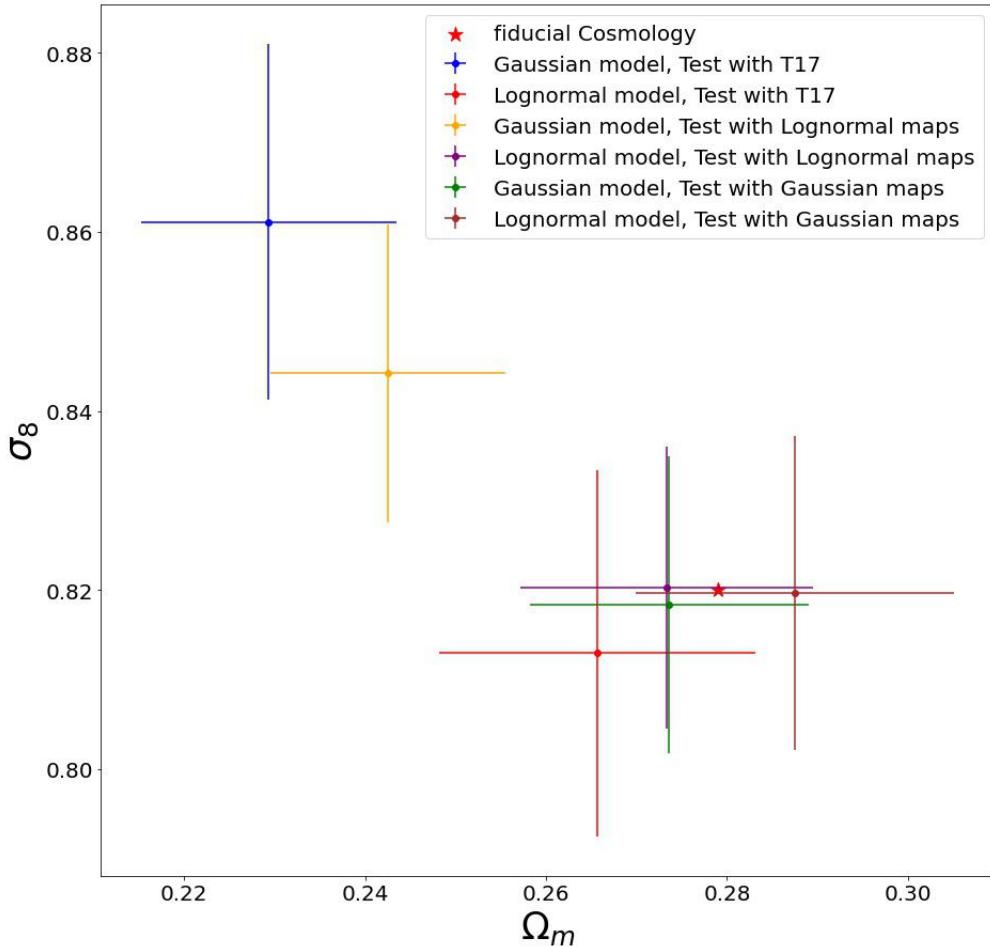


Figure 8.3: This figure visualizes the coverage of the fiducial cosmology using predicted mean and variance of the data points in 2D σ_8 - Ω_m plane. The red starred points correspond to the fiducial cosmology which corresponds to the truth value of labels. The error shows 1- σ confidence interval.

with the same inputs and hyperparameters, GCNN models may produce varying results, especially when applied to different types of datasets for training and testing. For more in-depth insights, please refer to Appendix B.

8.1.3 Sample Covariance

In Bayesian data analysis, our primary focus is on the posterior distribution, which is constructed by incorporating the prior information. One straightforward approach for deriving the posterior distribution is through the use of sample covariance of point clouds. In this method, we estimate the covariance structure of our data based on the density distribution of the observed data points. The resulting covariance matrix $\hat{C}_{i,j}$ provides a

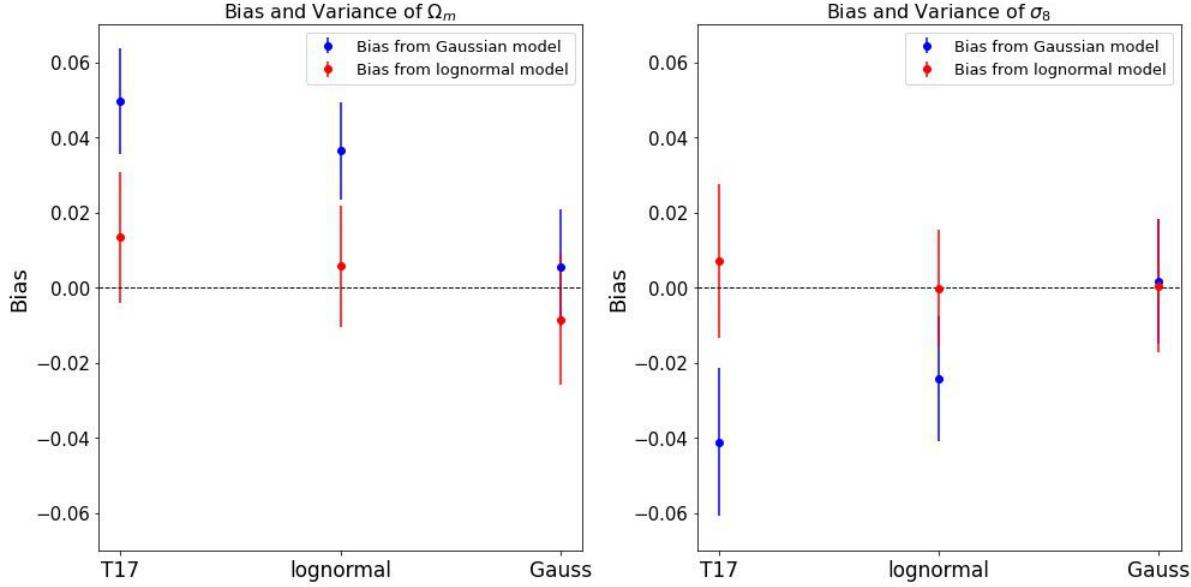


Figure 8.4: This figure also illustrates the representation of the fiducial cosmology's coverage using Bias and Variance for both Ω_m and σ_8 parameters. The blue points represent estimations from the Gaussian model, while the red points represent estimations from the lognormal model. The x-axis indicates the specific datasets employed as test datasets. The error bars depict the $1-\sigma$ confidence intervals, representing the standard deviation of each dataset.

measure of how data points are distributed, as expressed by the formula:

$$\hat{C}_{i,j} = \frac{1}{N-1} \sum_{i=1}^N (\hat{\theta}_i - \langle \hat{\theta} \rangle)^T (\hat{\theta}_j - \langle \hat{\theta} \rangle) . \quad (8.4)$$

Figure 8.6 visually represents the covariance matrix calculated using the above formula.

In continuation of the analysis discussed in Section 8.1.2, we now extend our investigation by incorporating prior information. A flat prior is employed, spanning the same range as the training dataset samples ($\Omega_m \in [0.15, 0.45]$ and $\sigma_8 \in [0.5, 1.1]$). We continue to assume a Gaussian likelihood, consistent with our employed loss function, and utilize the parameter covariance derived from the sample covariance. This allows us to construct the posterior distribution centered around the prediction mean. Figure 8.7 and Figure 8.8 provide visualizations of the posterior distribution, obtained through Markov Chain Monte Carlo (MCMC) sampling, with the `getdist` package for both sampling and visualization.

The outcomes of our posterior distribution closely align with the results observed in Figure 8.3. In the Gaussian model, the posterior distribution derived from Gaussian fields as test datasets successfully captures the fiducial cosmology within a $1-\sigma$ credible interval. Conversely, for the T17 simulation and lognormal field datasets, this does not hold true. For the lognormal model, both lognormal and Gaussian test datasets yield estimations consistent with the true cosmology. However, the T17 cosmology does not cover the fiducial

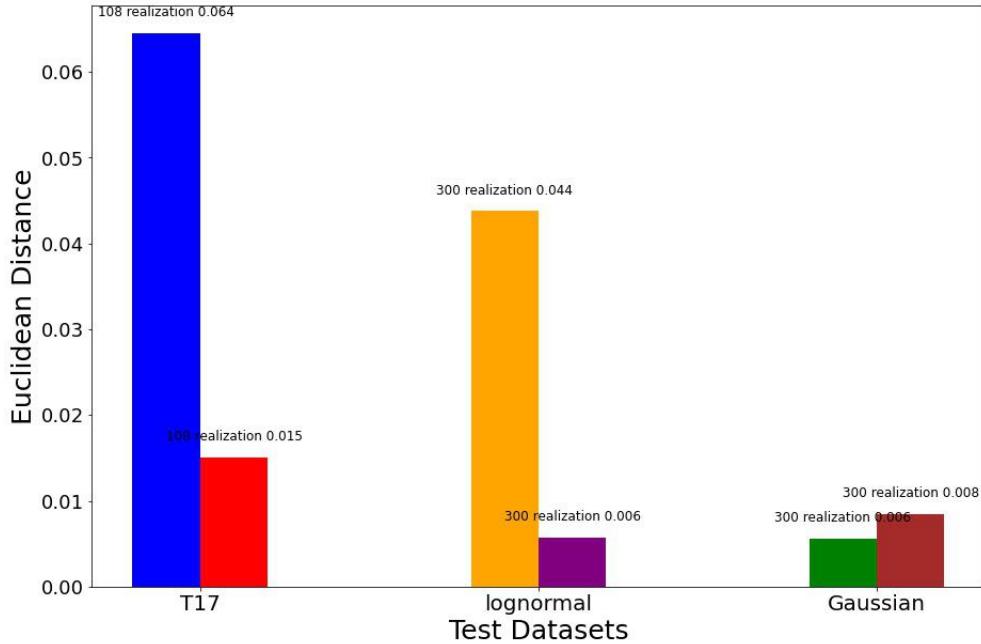


Figure 8.5: This bar plot illustrates the quality of estimators, as quantified by Equation (8.3). The x -axis represents the test datasets utilized, while the y -axis corresponds to the values of the Euclidean distance. The color scheme matches that of Figure 8.3.

value in the $1 - \sigma$ credible interval. This disparity is a distinct feature, distinguishing this observation from that presented in Figure 8.3.

Sample covariance represents a non-parametric approach to estimating the probability density function of a continuous random variable. However, it's essential to note that the precision of sample covariance-based probability density estimates can be comparatively lower than those derived from individual data realizations. In parallel, our GCNN architecture performs predictions for the covariance matrices of individual realizations under the assumption of a Gaussian likelihood function. In the following section, we leverage these estimated parameter covariances to derive the posterior distribution. Those results of clouds of points yielded surprising outcomes contrary to our initial expectations. Due to computational constraints, we limited our analysis to an angular resolution of $NSIDE = 128$, it can result in the smoothing out of non-Gaussianities. Initially, we expected either a similar distribution of point clouds or a uniform level of bias across all test datasets. However, the results were biased when using different distribution types as test datasets in contrast to the consistent combinations of training and test datasets that successfully recovered the fiducial cosmology in all discussed plots.

From these results, our primary conclusion is that bias can arise from the inconsistency in the assumptions regarding the density field, even at very low resolutions. As outlined in Appendix B, our findings were consistent when varying the random seed for generating random fields. However, the direction of bias was not consistent when using different training

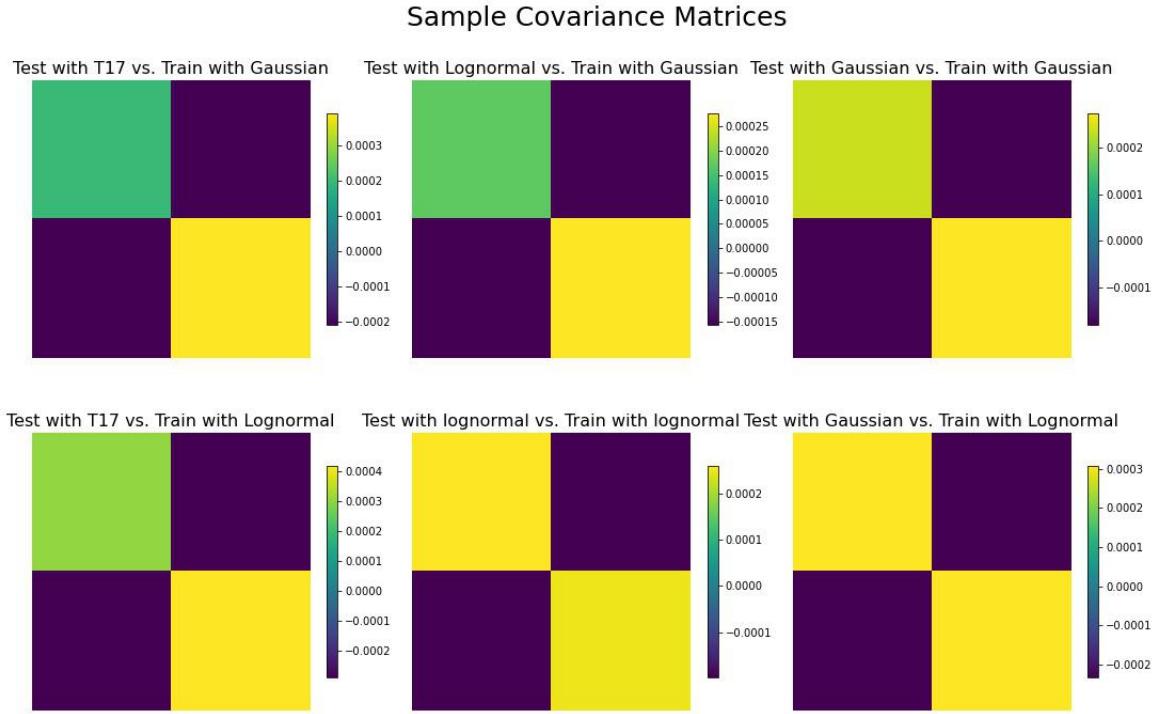


Figure 8.6: This figure illustrates parameter covariances estimated using kernel density estimation. The first row and column in this matrix correspond to Ω_m , while the second row and column correspond to σ_8 . Off-diagonal elements indicate correlations, positive or negative, while diagonal elements represent parameter variances.

datasets with varying random seeds for random realizations. This inconsistency can be attributed to unoptimized hyperparameters and training procedures. We speculate that this bias is correlated with the final loss value and may change with the level of generalization and overfitting of the machine learning model. Further exploration, particularly focusing on analyzing filters and understanding the higher-order information used to minimize the loss function, may provide additional insights into this bias effect.

8.2 Posterior Distributions from the Estimation using GCNN

In this section, first, we compute the parameter covariance matrix by utilizing the last three inputs of the GCNN model. Then, through a combination of likelihoods and the prior information, we calculate the posterior distribution. Finally, we provide an interpretation of the contour plots and conduct an evaluation of our GCNN-based methods.

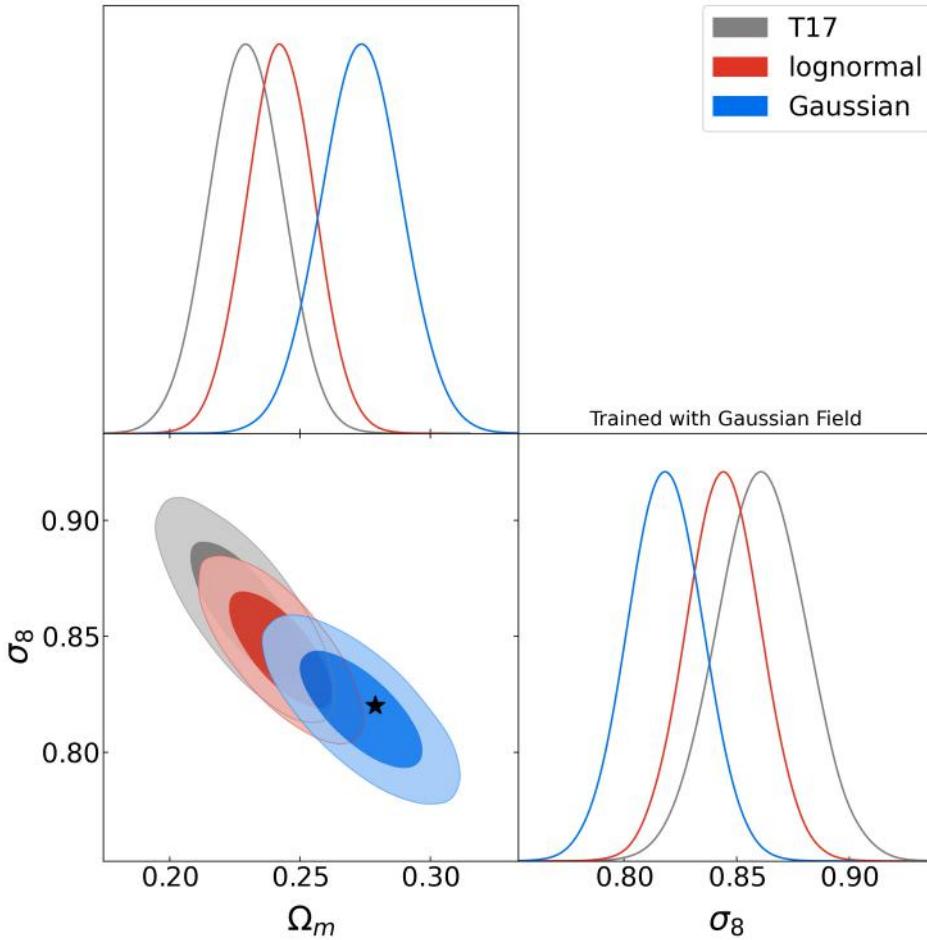


Figure 8.7: These contour plots visualize the posterior distribution in the σ_8 - Ω_m plane using the Gaussian model and parameter covariance derived from sample covariance. The colored areas represent credible intervals: the thin colored area corresponds to the 2σ credible interval, and the thick colored area corresponds to the 1σ credible interval. The gray area shows the credible interval using convergence maps from the T17 simulation as test datasets, the red area represents Gaussian fields, and the blue area represents lognormal fields.

8.2.1 Parameter Covariances

In this project, we employ the negative-log Gaussian likelihood to derive the covariance matrix directly from all individual data realizations. Each realization serves as a sample from the underlying distribution, and by computing the covariance matrix from all such samples, we capture the statistical relationships between variables based on actual data points without any smoothing involved. The parameter covariance is calculated from the

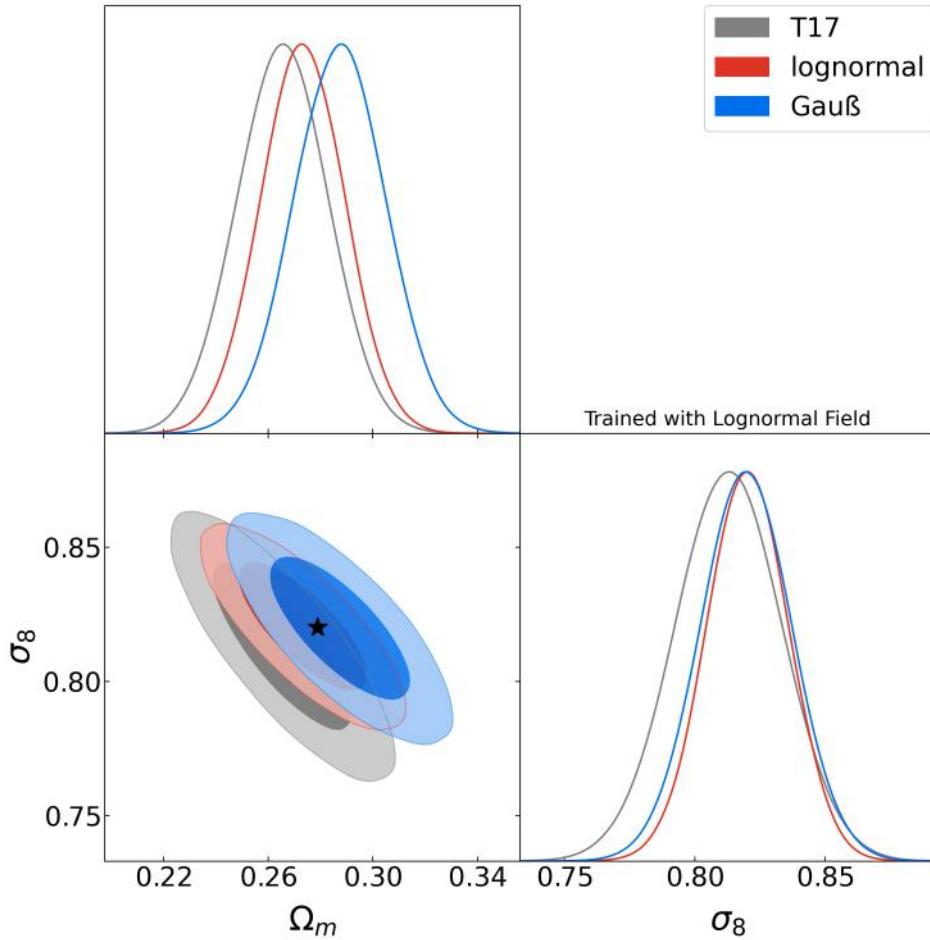


Figure 8.8: These contour plots visualize the posterior distribution in the σ_8 - Ω_m plane using the lognormal model and parameter covariance derived from sample covariance. The interpretation of the colored areas is analogous to that in Figure 8.7.

neural network outputs using equations (7.5) and (7.6), resulting in the following form:

$$\Sigma = \begin{pmatrix} e^{2y_2} & y_4 e^{y_2} \\ y_4 e^{y_2} & y_4^2 + e^{2y_3} \end{pmatrix}. \quad (8.5)$$

Figure 8.9 provides a visual representation of the mean parameter covariance matrix obtained through Equation (8.5). We then proceed to compare this with the sample covariance, as depicted in Figure 8.6. Notably, the diagonal elements of the covariance matrix signify variances, revealing that, in general, there is more error in the Ω_m direction, as evident in Figure 8.6. However, Figure 8.6 highlights that the difference in variance between σ_8 and Ω_m is generally small. The off-diagonal elements of the covariance matrix represent correlation coefficients, and in this case, both matrices exhibit similar values.

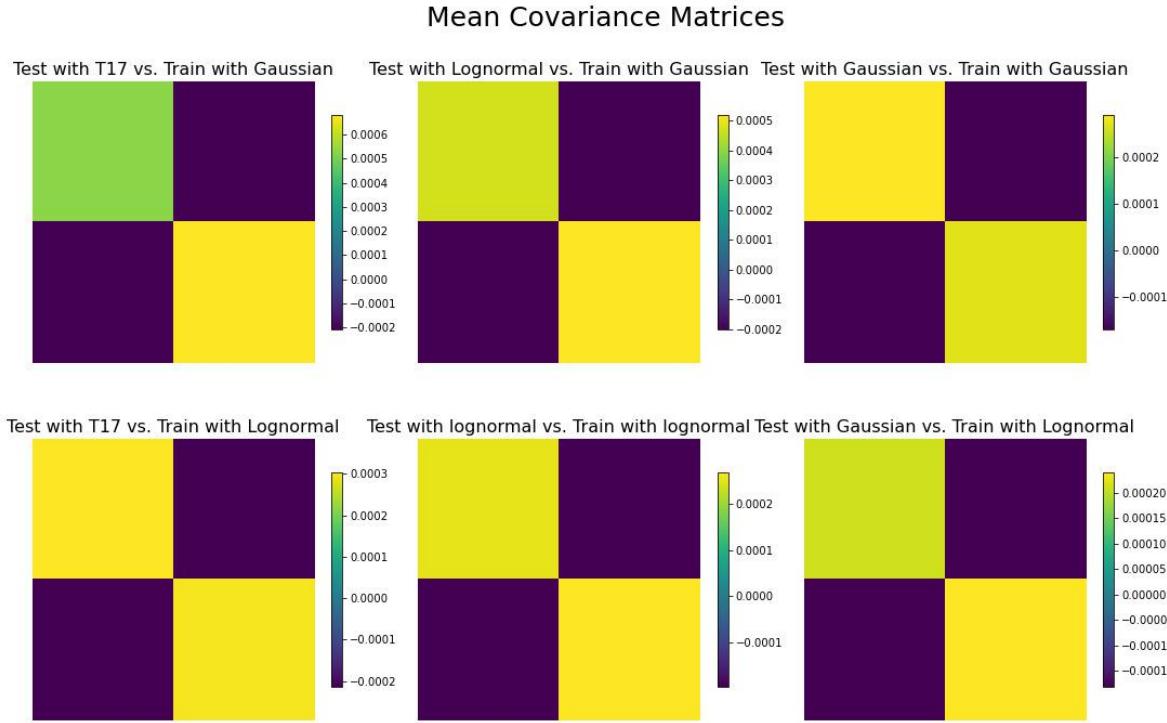


Figure 8.9: This figure presents the mean parameter covariance matrices estimated from the last three outputs of the GCNN model across all random realizations. The description is analogous to the figure 8.6.

8.2.2 Contours from Single Cosmology

In this GCNN framework, we have the capability to derive posterior distributions even from single test datasets. Therefore, before combining all the random realizations, we can create contour plots for individual test datasets. By employing Equation (8.5), We calculate the parameter covariances for a single realization, subsequently allowing for likelihood computation. By multiplying this likelihood with the prior and employing MCMC techniques, we obtain the posterior distribution for each realization. In Figure 8.10, we present the posterior distributions generated using the lognormal model for the first 6 random realizations of T17 simulations. It's important to observe that, although the size of the contours may differ across realizations, the general direction of correlation and shape consistently holds.

8.2.3 Posterior Distributions

We now turn our attention to comparing the combined posterior distributions across all realizations for two models and three types of test datasets. We employ two distinct averaging methods: one averages the outputs of GCNNs across all random realizations, while the other averages covariance matrices over all realizations. These methods are not

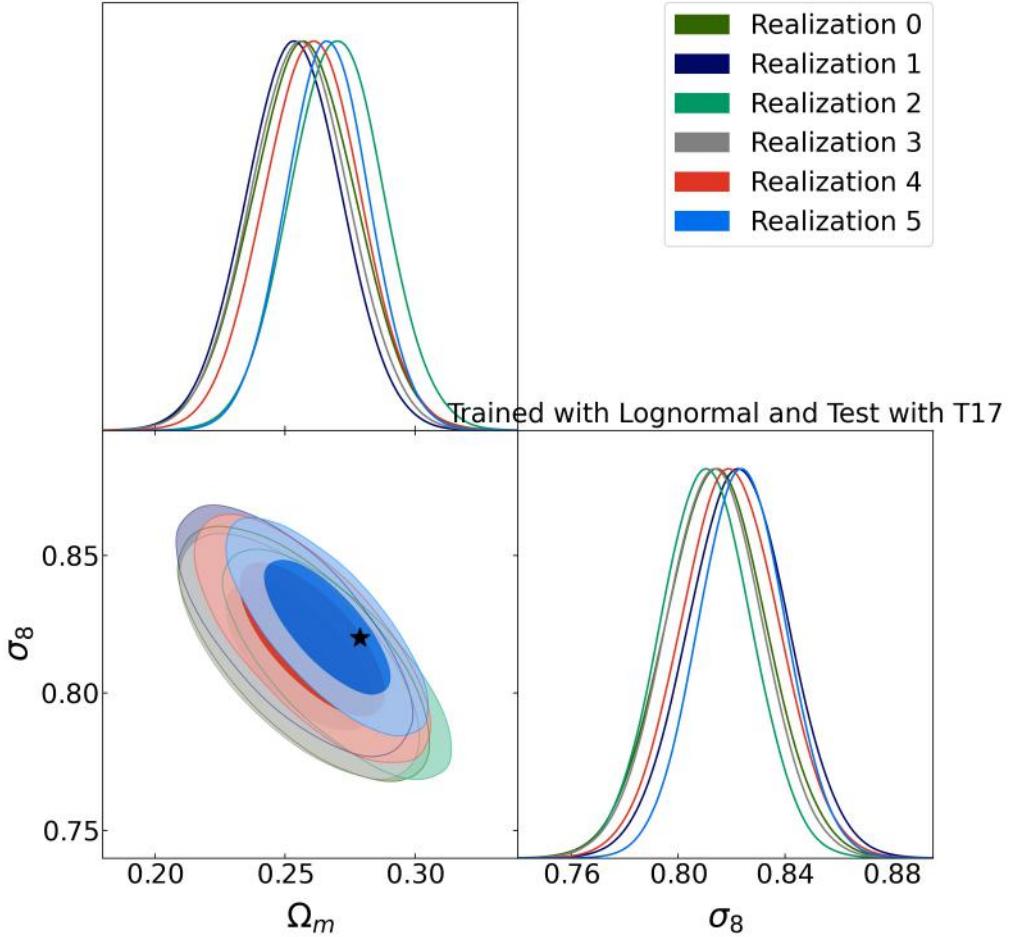


Figure 8.10: This contour plot illustrates the posterior distribution derived from various individual realizations of the T17 simulation [114] using the lognormal model. To enhance visibility, we have selected the first 6 realizations for the plot from a total of 108 realizations.

necessarily identical due to the presence of an exponential function in the conversion of GCNN outputs to covariance matrices. Figure 8.11 provides a visual representation of the posterior distribution centered around the prediction mean for these two averaging methods. Remarkably, the differences between them are almost negligible, and similar trends are observed in other models and test datasets as well.

It's important to note that directly averaging covariance matrices may not ensure that the resulting matrix remains positive-definite, a critical property for valid covariance matrices. It's crucial to emphasize that the two averaging methods are in agreement, ensuring the resulting covariance matrix remains positive semi-definite. This alignment effectively addresses any concerns related to the positive definiteness of the averaged covariance matrix.

Another challenge emerges with the precision matrix, which is the inverse of the covariance matrix. The precision matrix plays a pivotal role in computing the likelihood

function. However, matrix inversion is a non-linear process and may introduce additional biases into the estimation of the covariance matrix. While this concern is noteworthy, it falls outside the scope of this Master's thesis. We proceed with the assumption that we have a sufficient number of simulations to mitigate this issue.

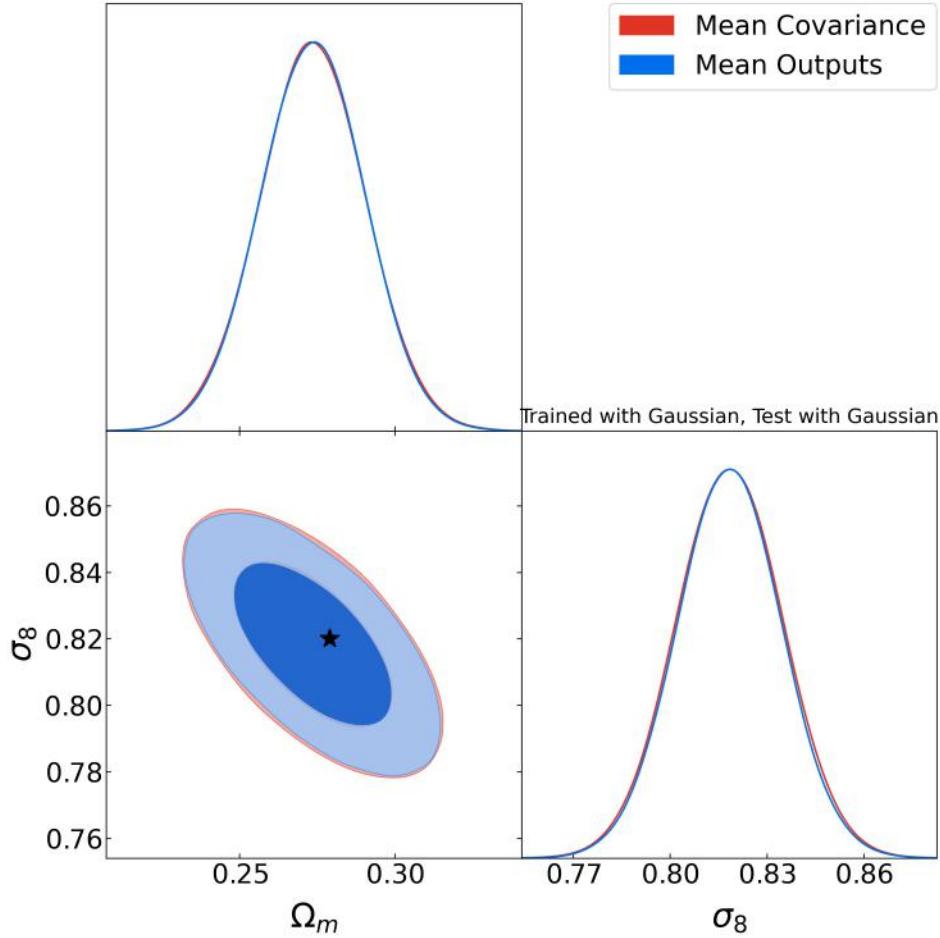


Figure 8.11: This figure presents a comparison of two posterior distributions. One is derived from averaging the covariance matrix, while the other is obtained from averaging the GCNN outputs. Both of these approaches are tested using Gaussian fields within the Gaussian model.

Using the method of covariance averaging, our study aims to explore various combinations of models and test datasets. The ensuing Figure 8.12 provides a visualization of the posterior distributions when employing Gaussian models with various test datasets. Specifically, we undertake a comparative analysis of posterior outcomes obtained through two distinct covariance estimation techniques: sample covariance and an averaging approach. In the context of sample covariance estimation, our observations reveal the emergence of three contour regions of approximately equal size. In contrast, when the averaged covariance method is applied, the most prominent contour corresponds to the T17 dataset,

followed by the lognormal and Gaussian datasets. It is noteworthy that within the T17 convergence and lognormal datasets, the sample covariance method exhibits a marginally higher degree of constraint compared to its Gaussian dataset counterpart. While the contours exhibit similar shapes, it is discernible that the employment of the averaged covariance method generally results in broader contour widths. Transitioning to the lognormal model, as depicted in Figure 8.13, and utilizing T17 convergence maps, we note that the posterior distributions for σ_8 are nearly indistinguishable when applying both covariance estimation methods. However, in the context of Ω_m , the utilization of the averaged covariance method yields a more pronounced constraint. For the Gaussian field, the adoption of the averaged covariance method imparts stronger constraints in both dimensions. Conversely, in the case of the lognormal field, both covariance estimation methods yield highly concordant results. .

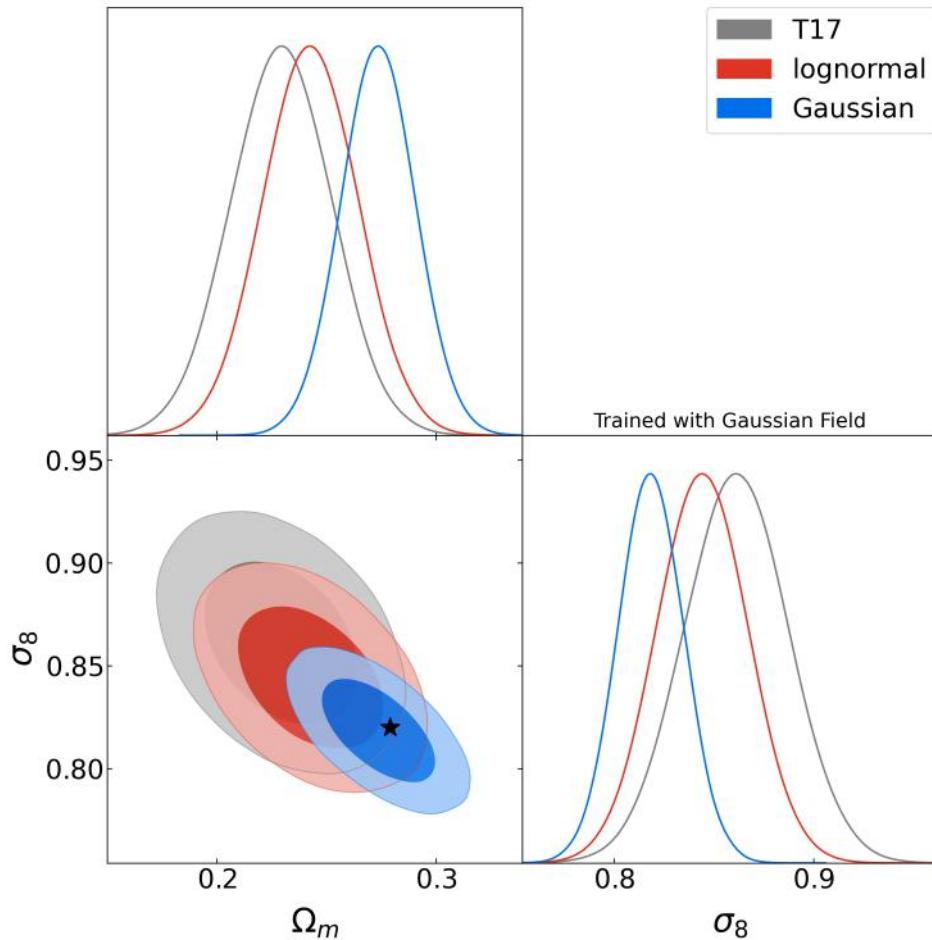


Figure 8.12: This figure showcases a comparison of three test datasets employing the Gaussian Model. The posterior distributions are plotted based on the averaged covariance and centered around the predicted mean.

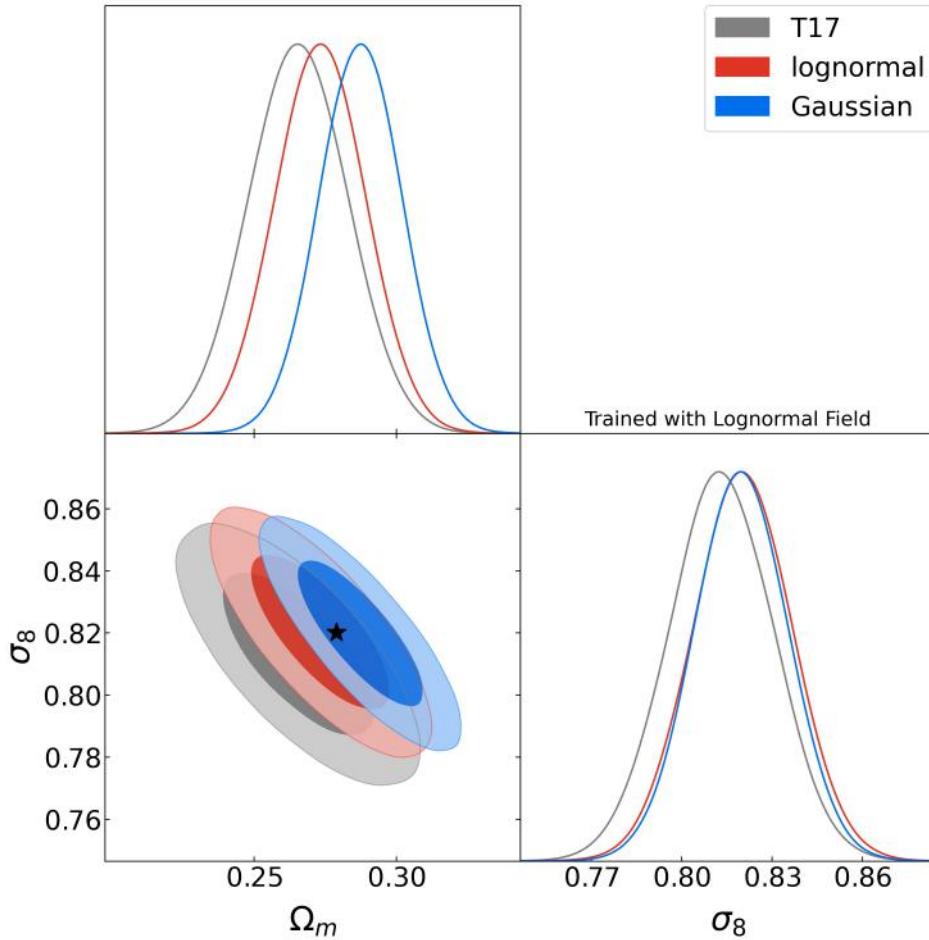


Figure 8.13: This figure showcases a comparison of three test datasets employing the Lognormal Model.

8.2.4 Figure of Merits

To compare the effectiveness of constraining power across different scenarios, we utilize a parameter evaluation metric known as the "figure of merit." This metric assesses the quality of the parameter covariance matrix, specifically for a pair of parameters, and is defined as follows:

$$\text{FoM} = \frac{1}{\sqrt{\det(\Sigma)}} = \frac{1}{\sqrt{\lambda_1 \lambda_2}} , \quad (8.6)$$

In this equation, λ_1 and λ_2 represent the eigenvalues of the parameter covariance matrix Σ . The figure of merit serves an intuitive purpose by quantifying the area enclosed by the parameter contour. A higher figure of merit signifies a posterior distribution that is more informative, resulting in narrower contours and offering more reliable estimates for the parameters in question. Following the preceding discussion, Figure 8.15 and Figure 8.14

present a visual representation of the figure of merit for σ_8 and Ω_m in various scenarios. These figures visualize the performance and effectiveness of different parameter covariance estimation methods.

First, we examine the figure of merit for the averaged covariance. In the case of the Gaussian model, the results align with our expectations. The scenario in which the model is tested with Gaussian fields exhibits the highest figure of merit. This outcome is consistent with our training methodology, as the model is optimized to provide the most constrained contours when evaluated with Gaussian fields. Subsequently, the figure of merit decreases for lognormal and T17 datasets. Interestingly, the lognormal model produces unexpected results, as the Gaussian fields perform better in constraining the parameters compared to the lognormal fields. On the whole, the lognormal model demonstrates superior performance over Gaussian models, which is consistent with our training results, where the lognormal models achieved lower loss values. Conversely, when employing the sample covariance estimation, a surprising reversal in results is observed. This surprising result calls for additional investigation and analysis in the future.

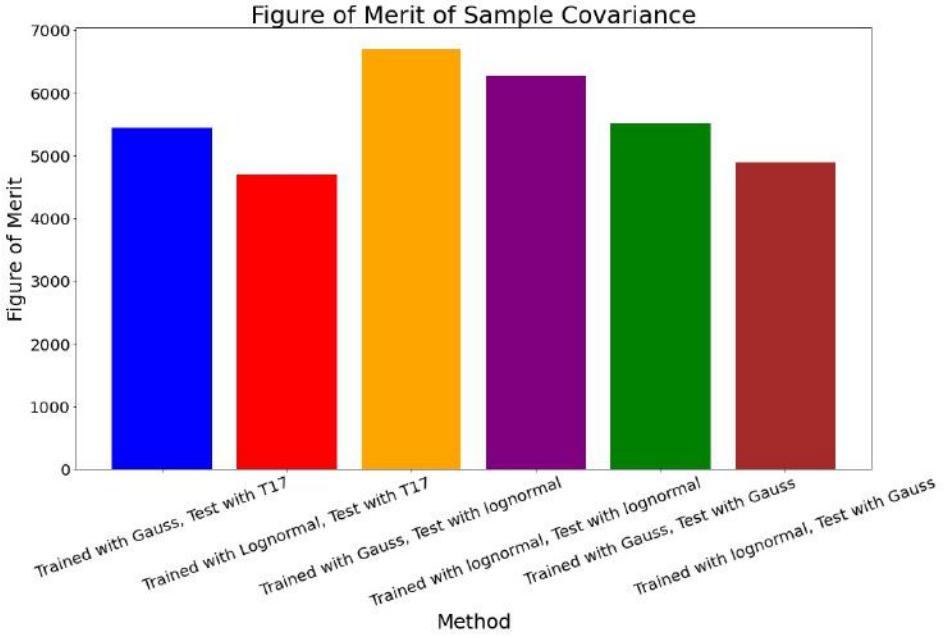


Figure 8.14: This bar plot illustrates the figure of merits of Sample covariance matrix of σ_8 and Ω_m .

8.3 Comparison with Fisher Forecast

One primary focus of this project lies in the development of more powerful statistical methods, which leverage the inherent non-Gaussianity present in density fields. Consequently, it is imperative to establish a comparison with traditional 2-point statistics as a fundamental

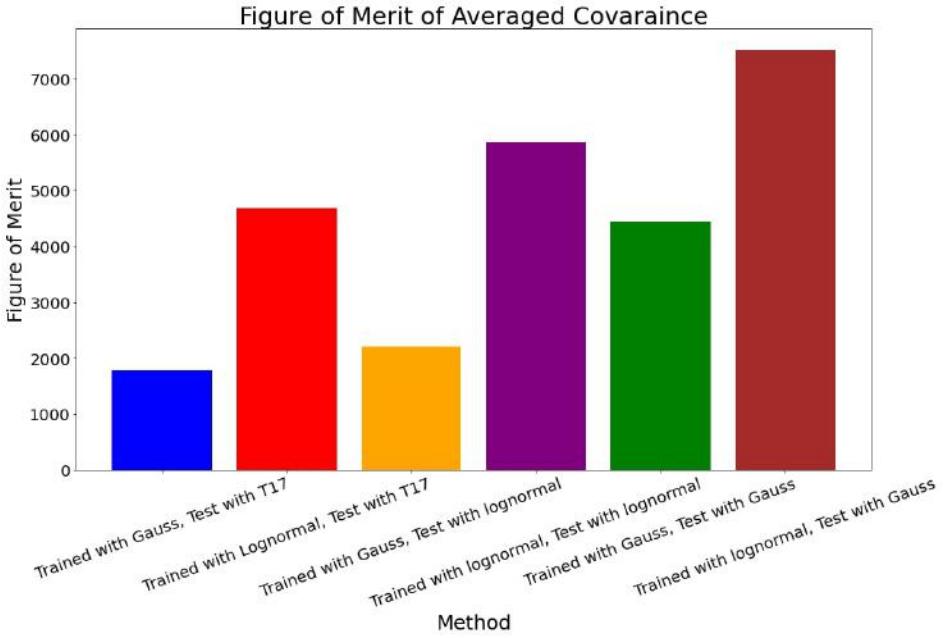


Figure 8.15: This bar plot illustrates the figure of merits of averaged parameter covariance matrix of σ_8 and Ω_m . The color scheme matches that of Figure 8.3.

aspect of this thesis. Furthermore, this comparative analysis with 2-point statistics serves as a robustness check for GCNN model employed in this research.

Specifically, for Gaussian fields, 2-point statistics serve as sufficient statistics, rendering them particularly apt for comparison with the field-based contour trained and tested using Gaussian fields. In this regard, we utilize the Fisher forecast from the power spectrum to compare with the Gaussian-Gaussian contour. Fisher forecasting, a methodology employed to predict the anticipated precision and covariance of parameter estimates within a given experimental or observational configuration, offers the advantage of yielding the most tightly constrained contours under the assumption of Gaussian likelihood. This is significant as the inverse of the Fisher information matrix establishes the Cramér-Rao lower bounds, representing the theoretical minimum variance achievable by any unbiased estimator².

Furthermore, it is essential to note that Fisher forecasting assumes homoscedastic error conditions³ in line with its application. This comparative analysis provides insights into the enhanced statistical capabilities of our approach compared to traditional 2-point statistics, contributing to an evaluation of the GCNN model's efficacy.

²The Cramér-Rao lower bounds provide the theoretical minimum value of variance. More detail in [125]

³Homoscedasticity refers to a situation where the variance of the residuals or the spread of data points remains roughly constant across all levels or values of an independent variable.[125]

8.3.1 Derivation of Fisher Matrix

To begin, we derive the Fisher matrix for the convergence power spectrum, parametrized by $\boldsymbol{\theta} = [\Omega_m, \sigma_8]$, drawing inspiration from the methodology outlined in [59]. Our analysis assumes that the likelihood function \mathcal{L} exhibits peaks around the true parameter values $\boldsymbol{\theta}_0 = [\Omega_{m,\text{true}}, \sigma_{8,\text{true}}]$. Under this assumption, we model the likelihood function as a multivariate Gaussian distribution (2.15) and employ a first-order Taylor expansion for the logarithm of the likelihood, yielding:

$$\ln\mathcal{L}(\boldsymbol{\theta}) \approx \ln\mathcal{L}(\boldsymbol{\theta}_0) + \frac{1}{2} \sum_{i,j}^2 \left. \frac{\partial\mathcal{L}}{\partial\theta_i\partial\theta_j} \right|_{\boldsymbol{\theta}_0} (\theta_i - \theta_{i,\text{true}})(\theta_j - \theta_{j,\text{true}}) . \quad (8.7)$$

This expansion leads to:

$$\mathcal{L}(\boldsymbol{\theta}) \approx \mathcal{L}(\boldsymbol{\theta}_0) \exp\left(-\frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^T \mathbf{H}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)\right) \approx \mathcal{L}(\boldsymbol{\theta}_0) \exp\left(-\frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^T \mathbf{C}^{-1}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)\right) , \quad (8.8)$$

where \mathbf{C}^{-1} represents the inverse of the parameter matrix, and \mathbf{H} is the Hessian matrix:

$$\mathbf{H}_{ij} = -\frac{\partial^2 \ln\mathcal{L}}{\partial\theta_i\partial\theta_j} . \quad (8.9)$$

From equation (8.8), we establish the relation:

$$\mathbf{C} = \mathbf{H}^{-1} . \quad (8.10)$$

Now, let's explore the expectation value of the parameter covariance. The expectation value of the Hessian matrix is termed the Fisher matrix \mathbf{F} . For the convergence power spectrum C_l , parametrized by Ω_m and σ_8 , it can be expressed as:

$$\mathbf{F}_{ij} \equiv \langle \mathbf{H}_{ij} \rangle = \left(\frac{dC_l}{d\theta_i} \right)^T \hat{C}_{\text{data}}^{-1} \left(\frac{dC_l}{d\theta_j} \right) , \quad (8.11)$$

where \hat{C}_{data} represents the data covariance matrix. Given that this matrix pertains to D observed data points, it possesses dimensions of $D \times D$. Consequently, the inverse of the Fisher matrix provides us with the expectation value of the parameter covariance matrix:

$$\mathbf{C} = \mathbf{F}^{-1} . \quad (8.12)$$

8.3.2 Data Covariance Matrixes

In Equation (8.11), it's clear that calculating the Fisher matrix requires access to the data covariance of C_l . In our experimental setup, we solely assume the presence of cosmic variance. This means that we do not apply any masks to the density map, and we do

not consider the influence of instrumental noises. Cosmic variance is an error term that doesn't depend on specific model parameters, and thus, it adheres to the assumption of homoscedasticity, where the error variance remains constant across different data points. However, in the process of simulating the density field, we apply a smoothing procedure that can introduce additional errors. Therefore, in this thesis, we employ two distinct approaches to determine this data covariance.

The first approach involves the derivation of an analytical covariance utilizing the Wick theorem⁴. Specifically, when considering cosmic variance alone, we can estimate the angular power spectrum \hat{C}_l as follows [40]:

$$\hat{C}_l = \frac{1}{2l+1} \sum_{m=-l}^l |a_{lm}|^2 . \quad (8.13)$$

Subsequently, the covariance of these angular power spectrum estimates, denoted as $\hat{\mathbf{C}}_{\text{data}}(\hat{C}_{l_1}, \hat{C}_{l_2})$, can be straightforwardly defined as:

$$\hat{\mathbf{C}}_{\text{data}}(\hat{C}_{l_1}, \hat{C}_{l_2}) = \langle \hat{C}_{l_1} \hat{C}_{l_2} \rangle - \langle \hat{C}_{l_1} \rangle \langle \hat{C}_{l_2} \rangle , \quad (8.14)$$

where the first term is expressed as:

$$\langle \hat{C}_{l_1} \hat{C}_{l_2} \rangle = \frac{1}{(2l_1+1)(2l_2+1)} \sum_{\substack{m_1=-l_1 \\ m_2=-l_2}}^{l_1, l_2} \langle a_{l_1, m_1} a_{l_1, m_1}^* a_{l_2, m_2} a_{l_2, m_2}^* \rangle . \quad (8.15)$$

To compute the term $\langle a_{l_1, m_1} a_{l_1, m_1}^* a_{l_2, m_2} a_{l_2, m_2}^* \rangle$, we rely on the assumption that the spherical harmonic coefficient a_{lm} follows a Gaussian distribution, enabling us to apply Wick's theorem, which states [113]:

$$\langle ABCD \rangle = \langle AB \rangle \langle CD \rangle + \langle AC \rangle \langle BD \rangle + \langle AD \rangle \langle BC \rangle . \quad (8.16)$$

It is valid $a_{l,m} = a_{l,-m}^*$ since the density field is real quantity. Then we can write the term $\langle a_{l_1, m_1} a_{l_2, m_2} \rangle = \delta_{l_1, l_2} \delta_{m_1, -m_2} C_l$ where δ_{l_1, l_2} and $\delta_{m_1, -m_2}$ are Kronecker's delta. It yields, that

$$\begin{aligned} \langle a_{l_1, m_1} a_{l_1, m_1}^* a_{l_2, m_2} a_{l_2, m_2}^* \rangle &= \langle a_{l_1, m_1} a_{l_1, m_1}^* \rangle \langle a_{l_2, m_2} a_{l_2, m_2} \rangle + \langle a_{l_1, m_1} a_{l_2, m_2} \rangle \langle a_{l_1, m_1}^* a_{l_2, m_2}^* \rangle \\ &\quad + \langle a_{l_1, m_1} a_{l_2, m_2}^* \rangle \langle a_{l_1, m_1}^* a_{l_2, m_2} \rangle \\ &= C_{l_1} C_{l_2} + \delta_{l_1, l_2} \delta_{m_1, -m_2} C_{l_1}^2 + \delta_{l_1, l_2} \delta_{m_1, m_2} C_l^2 l_1 . \end{aligned} \quad (8.17)$$

Taking into account the real nature of the density field ($a_{l,m} = a_{l,-m}^*$), we can simplify the term $\langle a_{l_1, m_1} a_{l_2, m_2} \rangle$ to $\delta_{l_1, l_2} \delta_{m_1, -m_2} C_l$. Subsequently, we derive the covariance matrix for the estimation of the angular power spectrum as:

$$\hat{\mathbf{C}}_{\text{data}}(\hat{C}_{l_1}, \hat{C}_{l_2}) = \frac{2\delta_{l_1, l_2}}{(2l_1+1)} C_{l_1}^2 . \quad (8.18)$$

⁴It is the same as Isserlis's theorem.

In addition to the analytical method, we have also adopted a simulated covariance approach, which is implemented using the `Flask` framework. This method involves simulating the density field and subsequently measuring the two-point function C_l . The next step entails computing the four-point function $\hat{\mathbf{C}}_{\text{data}}$ as follows:

$$\hat{\mathbf{C}}_{\text{data}} = \frac{1}{N-1} \sum_{i=1}^N (\hat{C}_{l,i} - \langle \hat{C}_l \rangle)(\hat{C}_{l,i} - \langle \hat{C}_l \rangle)^T. \quad (8.19)$$

Here, N signifies the number of independent realizations. This approach furnishes us with an estimation of the data covariance matrix, a critical component of our analysis. However, it's important to note that while we now possess an estimation of $\hat{\mathbf{C}}_{\text{data}}$, we actually require $\hat{\mathbf{C}}_{\text{data}}^{-1}$ for our calculations. The simulated covariance (8.19) serves as an unbiased estimator, yet the straightforward application of matrix inversion results in a non-linear operation. Consequently, the relationship $\langle \hat{\mathbf{C}}_{\text{data}}^{-1} \rangle = \mathbf{C}_{\text{data}}[\boldsymbol{\theta}_0]^{-1}$ does not hold. To address this, we introduce a corrective factor known as the Kaufmann-Hartlap factor [58]:

$$\langle \hat{\mathbf{C}}^{-1} \rangle = \frac{N_{\text{sim}} - 1}{N_{\text{sim}} - N_{\text{data}} - 2} \mathbf{C}[\boldsymbol{\theta}]^{-1}, \quad (8.20)$$

where N_{sim} represents the number of simulations used to estimate the covariance, and N_{data} . This factor ensures the validity of the relation, allowing us to proceed with our analysis.

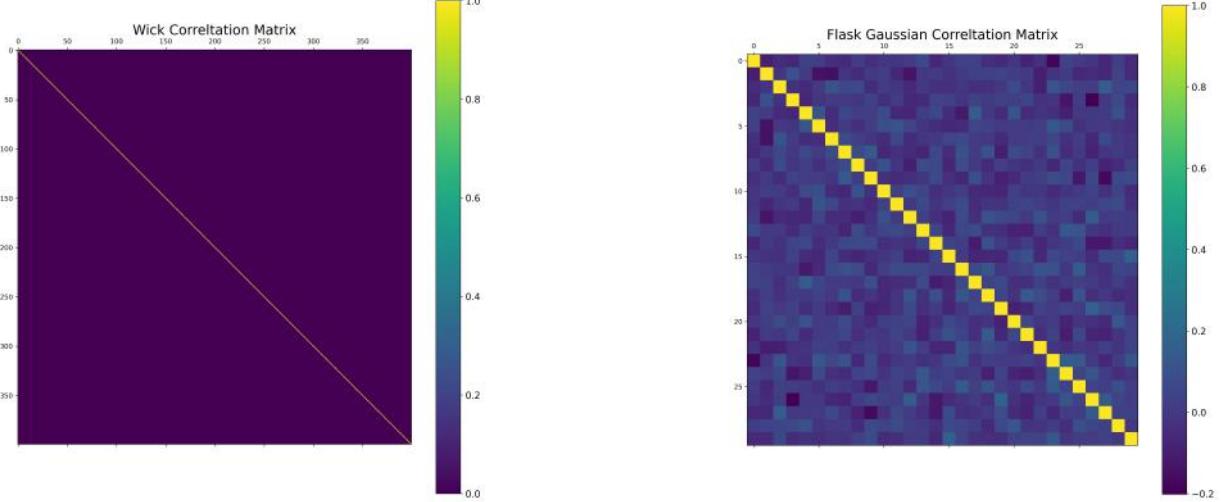


Figure 8.16: This figures visualize Correlation matrices for angular power spectrum C_l .

Figure 8.16 presents two data correlation matrices⁵, each offering a distinct perspective on the statistical properties of the data.

⁵In the visualization presented here, a correlation matrix is plotted for enhanced clarity in illustrating off-diagonal terms. However, it's important to note that in the actual computation of the Fisher matrix and related analyses, the covariance matrix is employed.

On the left-hand side, we have the correlation matrix from analytical covariance matrix, as derived in equation (8.18). This analytical covariance matrix is constructed using angular power spectrum measurements (C_l) for a range of l values, spanning from 1 to 400, roughly corresponding to an $NSIDE$ of 128 in the `healpy` function. Notably, the analytical covariance matrix exclusively features diagonal elements, consistent with its definition.

Conversely, on the right-hand side, we encounter the correlation matrix from the covariance matrix generated using the 300 `Flask` Gaussian maps, as outlined in equation (8.19). This matrix accounts for the Kaufmann-Hartlap factor (8.20) and employs a logarithmic binning scheme with 30 bins, averaging the angular power spectrum data. Unlike the analytical counterpart, the `Flask` covariance matrix exhibits off-diagonal elements. This emergence of off-diagonal terms can be primarily attributed to the smoothing effect applied by the `healpix` function during the simulation process.

8.3.3 Comparison with GCNN Model

Once the data covariance has been computed, our next step is to determine the derivative term required to calculate the Fisher matrix. To achieve this, we generate the angular power spectrum using the `Class` code, coupled with the Limber integral method, which exhibits a controlled deviation of 5% from the reference values⁶. With these perturbed power spectra, we calculate the derivative term of (8.11) using central differences. Subsequently, we employ this derivative term to compute the Fisher covariance matrix as outlined in equation (8.11). Finally, we calculate the parameter covariance using equation (8.12).

Figure 8.17 offers a comparative analysis of the posterior distributions obtained through Fisher analysis of C_l and our Gaussian GCNN model. The Fisher contour, centered around the fiducial cosmology, is depicted in red, corresponding to the Fisher contour derived from Gaussian `Flask` data covariance. In contrast, the blue region represents the Fisher contour derived from analytical Gaussian data covariance. The gray area portrays the posterior distribution derived through Sample using Gaussian fields, while the green area represents the posterior distribution obtained from the averaged parameter covariance. Our comparison begins with an examination of the two Fisher contours. It becomes apparent that the Fisher matrix derived from analytical covariance yields slightly tighter constraints compared to the `Flask` contour. Moving on to the comparison between the GCNN contours and the contours resulting from Fisher analysis, a notable discrepancy emerges. The Fisher analysis contours exhibit significantly tighter constraints and reveal an exceptionally high degeneracy between Ω_m and σ_8 . In the direction of σ_8 , all contours exhibit similar constraints, while in the Ω_m direction, the Fisher contours notably outperform the GCNN contours.

In the case of Gaussian fields, 2-point statistics serve as sufficient statistics⁷, and ideally, they should align closely with the contours generated by the GCNN model when the

⁶It's noteworthy that we conducted sensitivity tests with deviations of 1% and 10%, and the results remained largely consistent across these variations.

⁷Sufficient statistics refer to summary statistics that capture the full information necessary for parameter estimation.

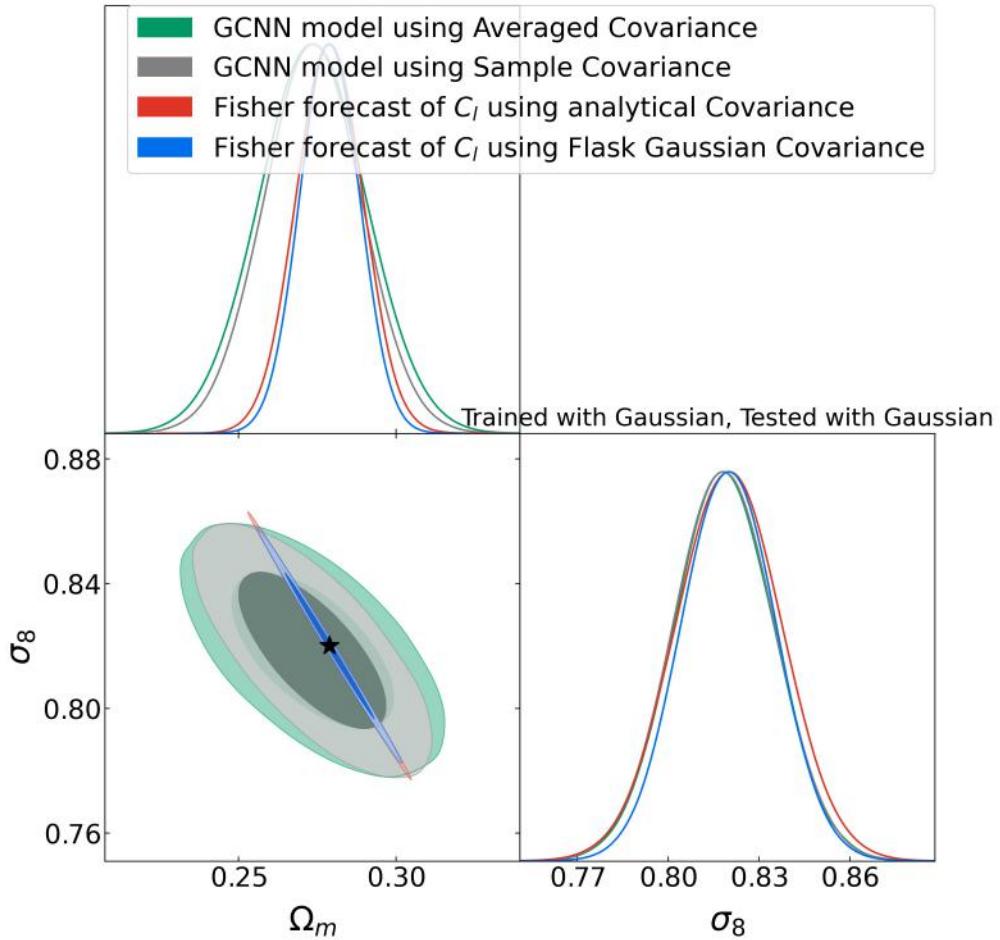


Figure 8.17: This plot illustrates contours generated by the Gaussian GCNN using Gaussian test datasets alongside contours obtained from Fisher analysis of C_l .

GCNN has captured all the information contained in the Gaussian field. However, our results indicate that the Fisher analysis produces tighter constraints compared to the GCNN model. This suggests that our GCNN model hasn't fully captured all the information present in the Gaussian fields. Two main difficulties contribute to this discrepancy. First, our experimental setup is overly idealized as it assumes only cosmic variance, whereas in real observations, additional sources of uncertainty such as masking and instrumental noise introduce unconstraints in parameter estimation. Consequently, Fisher analysis achieves the tightest contours, and replicating this with the GCNN is a challenging task. The second challenge lies in the training process. The loss during GCNN training indicates that our machine-learning model is still in the generalization phase. To improve the model's accuracy, I would need either more training time or better hyperparameters. Training a single model already takes more than a day, so finding a way to enhance model accuracy efficiently is essential. One proposed method is to modify the loss function. In our current calculation, we use an exponential function to enforce positive semidefiniteness of diagonal

elements of the covariance matrix. However, the diagonal elements of the parameter covariance derived from the Fisher Forecast are very small. To achieve this, we would require exceptionally large negative outputs from neural networks. Thus, I propose modifying the loss function to use parameter covariances from the Fisher Forecast. For example, a possible parametrization of the parameter covariance in terms of GCNN model outputs could be:

$$\Sigma = \begin{pmatrix} \sigma_1^{2y_2} & y_4\sigma_1^{y_2} \\ y_4\sigma_1^{y_2} & y_4^2 + \sigma_2^{2y_3} \end{pmatrix}. \quad (8.21)$$

Here, σ_1 and σ_2 are diagonal elements of the covariance matrix derived from the Fisher Forecast of Gaussian fields. The comparison of 2-point methods like the Fisher Forecast provides insight into the robustness of our GCNN model. Hence, following the completion of this thesis, our central objective will shift towards conducting thorough comparisons with established robust statistics for cosmic density fields. This approach aims to enhance our understanding of what the machine is truly learning and to address the fundamental question of whether field-based inference through machine learning can indeed advance our knowledge about the universe.

Chapter 9

Conclusion

Non-linear structure formation leads to non-Gaussianity in cosmic density fields, rendering traditional 2-point summary statistics suboptimal for estimating cosmological parameters due to their insensitivity to non-Gaussian features. In this Master’s thesis, we have developed a field-based inference pipeline capable of capturing the non-Gaussian information inherent in cosmic density fields. Our approach is built upon the lognormal model, a computationally efficient and powerful model for describing late-time cosmic density fields. However, the precise accuracy of this approximation in estimating posterior distributions through field-based inference remains an open question.

In this thesis, we have established a mock simulation pipeline employing the lognormal model to generate both training and test datasets. This pipeline encompasses several steps:

- (i) Generating power spectra for given cosmological parameters using `CLASS` [76] and calculating log-normal shift parameters with `CosMomentum` [41].
- (ii) Integrating these power spectra along the line of sight using the source galaxy redshift distribution from the Dark Energy Survey (DES) year-3 data [45] to obtain the convergence angular power spectrum.
- (iii) Generating full-sky convergence maps based on these input power spectra and log-normal shift parameters using `Flask` [124].

We generated these maps for a range of different input cosmologies to create a training set for cosmological parameter estimation. Additionally, we generated Gaussian fields to facilitate comparisons and robustness checks with 2-point statistics. Using 10,000 lognormal maps, 10,000 Gaussian maps, and cosmological parameters (σ_8 and Ω_m) as labels, we trained two distinct GCNN models to estimate posterior distributions. Our GCNN architecture leveraged `DeepSphere`, a graph convolutional neural network designed for spherical data representation, such as `HEALPix` maps, which respects rotational invariance—unlike conventional convolutional neural networks. We utilized the negative log Gaussian likelihood as our loss function, enabling not only the estimation of cosmological parameters but also the estimation of the parameter covariance matrix.

We tested the trained network using three types of test datasets: 300 lognormal fields, 300 Gaussian fields, and 108 convergence fields from T17 simulations. These tests aimed to investigate how different approximations impact the resulting posterior distributions in field-based inference. Our primary findings can be summarized as follows:

- (i) Both of our inference pipelines yield unbiased parameter estimations when using the same types of training and test datasets. However, different combinations of training and test datasets can introduce significant biases.
- (ii) Different combinations of test and training datasets can lead to the overestimation or underestimation of posterior distributions compared to the same combination of training and test datasets.
- (iii) The estimates of cosmological parameters obtained from our GCNN models roughly agree with the contour range from the Fisher analysis. However, our GCNN model does not capture the strong degeneracy exhibited in the Fisher analysis.

In summary, further analysis is needed in order to understand our results, particularly within the inference pipeline, to obtain robust predictions that align with 2-point statistics when using Gaussian field training datasets and that surpass 2-point statistics when using lognormal fields. One challenge in achieving this is tuning hyperparameters or finding the correct model architecture. In this Master’s thesis, our GCNN architecture faced limitations due to computational constraints. Running a single model took over a day, and memory requirements imposed limitations on tunable hyperparameters. Therefore, it is necessary to adapt our GCNN architecture to be more memory-efficient.

Our analysis was constrained to $NSIDE = 128$ due to computational costs, resulting in the smoothing out of significant non-Gaussian information on small scales. Additional computational resources and investigations into smaller scales will yield more substantial results in the future.

Our analysis pipeline was also executed under highly idealized conditions compared to real observations. We marginalized the parameters of Λ CDM except σ_8 and Ω_m . It is crucial for a full and realistic analysis to include all model parameters (e.g., H_0) and assess their correlations. Additionally, we need to account for instrumental effects and the limitations of survey geometry. In real convergence fields observed through weak gravitational lensing, there are masks and noise that need to be considered in our analysis.

Another significant challenge in our inference pipeline is the interpretability of machine learning processes. Achieving robust estimations requires addressing this challenge, potentially through filter-based interpretation or by comparing results from machine learning with those from 2-point and higher-order robust statistics.

Ultimately, our goal is to apply this technique to DES Y3 data to enhance existing constraints on cosmological parameters. We aim to update our understanding of the universe in a Bayesian manner.

Chapter 10

Futural Research

In this Master’s thesis, our primary objective was to tackle several substantial open questions at the intersection of cosmology and artificial intelligence. This chapter serves as an introduction to our research interests and presents our proposed ideas for addressing these open inquiries, which I intend to implement in the future.

10.1 Towards Trustworthy Estimations in Field-Based Cosmology with Machine Learning

How can we establish trust in machine learning estimations derived from field-based methods? Field-based methods have the potential to capture the complete information embedded in cosmic density fields, including intricate non-Gaussianity details at small scales, provided that our simulations accurately model these features. Do these estimations lead to the same posterior distribution as those obtained from Cosmic Microwave Background (CMB) measurements? How does the information propagate to higher-order cumulants or smaller scales? If discrepancies arise, what factors contribute to them? These questions are pivotal for advancing our understanding of the universe in a Bayesian manner. Unfortunately, our current inference pipeline often operates as a black box, severely limiting interpretability.

To address this challenge, I propose ideas aimed at enhancing the interpretability of machine learning-based inferences. The first idea involves leveraging filter observations. Saliency maps, a widely adopted tool for interpreting convolutional neural network layers [81], can be integrated into our pipeline. Saliency maps help us see which areas of an input image have a strong impact on a network’s decision, providing insight into the model’s actions and making it more interpretable. While generating these saliency maps within our pipeline is feasible, the true challenge lies in interpreting them and establishing quantitative and qualitative criteria to assess the relevance of specific features during training. One suggestion is to measure quantities such as the two-point function, skewness, and kurtosis from these maps. Quantifying these moments is crucial for understanding which information is utilized or dominant in estimating the parameters or covariance matrix.

The second idea involves designing an interpretable neural network structure. For example, in the paper by Miles et al. (2021) [85], CNNs were constructed to distinguish between two snapshots of many-body states in quantum gas spectroscopy, a task that cannot be accomplished using two-point correlation functions alone. Notably, the architecture in this paper is interpretable, allowing us to determine the significance of different correlation function orders for classifying the data. However, applying this approach to regression problems and GCNNs presents significant challenges.

Another method I propose is to interpret the posterior distribution derived from neural networks by comparing it to analytical higher-order statistics. This approach would enable us to estimate the realistic range of improvement in the posterior distribution compared to standard two-point statistics. However, modeling higher-order statistics is a complex task, and performing a comprehensive analysis of higher-order statistics requires computationally efficient models.

Understanding the behavior of machine learning models is crucial for the robustness of field-based methods, and this can be achieved through comparisons with analytical higher-order statistics. This is a task that must be undertaken in the future, as it has the potential to resolve existing tensions in the Λ CDM model, such as those related to S_8 or H_0 .

10.2 Addressing Real-World Complexities in Field-Based Cosmological Inference

In this Master’s thesis, we have simplified our analysis for the sake of clarity and tractability. In actual data obtained from lensing surveys, complexities such as instrumental noise and survey area limitations are bound to be present. The ultimate objective of this project is to extend the field-based inference methodology to real datasets, such as those collected by the Dark Energy Survey (DES) or the European Space Agency’s EUCLID mission. In this section, we will explore the future directions for this endeavor.

First, we address the issue of instrumental limitations. Real-world data from galaxy surveys is often compromised by instrumental noise and the inherent constraints of the survey area. Incorporating instrumental noise is a feasible task; we can introduce Gaussian noise into both the training and test datasets to analyze its impact on the derived posterior distribution across various noise levels. However, incorporating survey masks presents a more complex challenge. To tackle this, we would need to modify our GCNN pipeline since its current architecture is designed for full-sky maps. While this modification may be daunting, it offers the advantage of reducing the pixel count, which can alleviate memory requirements during training.

Second, we consider the expansion of our analysis to encompass additional parameters. Our current analysis focuses on two parameters from the Λ CDM cosmological model. However, the full Λ CDM model includes six parameters, and capturing the correlations between these parameters is not straightforward. Furthermore, when considering extensions of the Λ CDM model or including astronomical parameters, the parameter space grows substan-

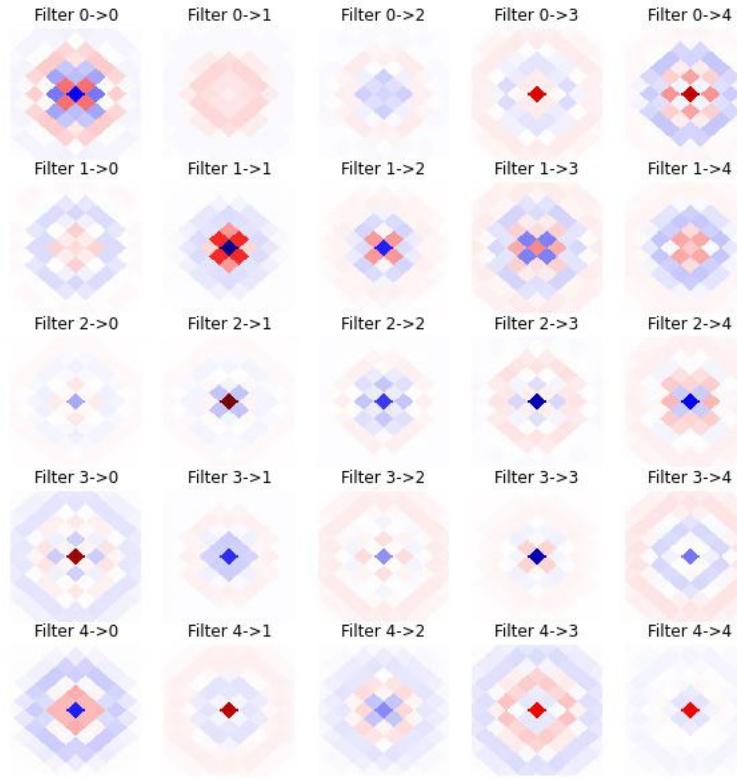


Figure 10.1: These figures depict the Gnomonic projection of filters from a sample of GCNN models. The labels indicate the input layer’s number followed by the output layer’s number. Unfortunately, we did not discover a practical application for enhancing the interpretability of our GCNN pipeline using this visualization.

tially. This expansion necessitates a significantly larger training dataset. However, it is conceivable that our simulation pipeline may become a bottleneck due to its processing speed. One approach to mitigate this challenge is to optimize code for GPU processing, as this has previously sped up the integration component significantly. Nevertheless, memory transfer from CPU to GPU remains a limiting factor, making it challenging to write the entire codebase¹, including libraries such as **Class** [76] or **Flask** [124], for GPU². Alternatively, standard emulation techniques, such as using emulators like **COSMOPOWER** [112] for the Boltzmann solver or implementing additional neural networks for predicting maps based on cosmological parameters, could be explored. It’s worth noting that our simulation pipeline relies on **CosMomentum** [41], which utilizes Eisenstein-Hu transfer functions designed for

¹Another limitation is that GPU computation is less accurate with respect to CPU computation due to the limitations of floating-point numbers.

²I have rewritten our simulation pipeline to utilize GPU processing, with the exception of **Flask**, **Class** and **CosMomentum**. The Limber integration step showed a notable increase in speed. However, the overall execution time is longer compared to the CPU-based version due to the data transfer between CPU and GPU.

Λ CDM cosmology. To extend our analysis to non-standard cosmologies, adjustments to predict lognormal shift parameters λ will be necessary.

Thirdly, I suggest analyzing higher-resolution maps. Non-Gaussianity resulting from nonlinear structure formation is more pronounced at smaller scales, potentially leading to more significant improvements in estimation compared to standard two-point statistics. However, this approach introduces computational challenges. For instance, transitioning from $NSIDE = 128$ to $NSIDE = 256$ quadruples pixel size, increasing memory requirements and computational time. Moreover, higher-resolution data may demand more complex GCNN models³. Ensuring the robustness of our model on smaller scales presents another concern. Lognormal approximation has limitations at small scales, requiring training with N-body simulations that provide sufficient data, such as `CosmoGrid` [68]. Additionally, addressing small-scale phenomena necessitates precise baryonic models. Consequently, understanding the connection between hydro simulations, which model baryonic matter, and N-body simulations, which model dark matter, becomes imperative when incorporating smaller scales into our analysis.

Lastly, I would like to address our assumption of a Gaussian likelihood. While we have assumed a Gaussian likelihood for our loss function and posterior distribution generation in this thesis, we have not empirically verified its validity. One promising approach to move beyond this assumption is to incorporate methods from likelihood-free inference (LFI) into our pipeline. Notably, recent work by Fluri et al. (2021) [38] discussed the application of various LFI methods, such as Gaussian Process or density estimation likelihood-free inference (Delfi), in the context of field-based inference.”

10.3 Cosmology beyond Λ CDM

We find ourselves at the crossroads of two distinct eras – the twilight of the Λ CDM paradigm and the dawn of new physics [7]. The execution of field-based inference is limited by the modeling of random fields based on theory. Therefore, if we have a new physical model that we can simulate, we can then execute field-based inference. The first and foremost requirement is the development of such theoretical models. However, to truly transcend the boundaries of Λ CDM and validate the accuracy of these models, we propose the exploration of novel observables, and personally, two methods are particularly exciting for us.

Firstly, the observation of parity violation within the distribution of large-scale structure captivates our curiosity. Standard particle physics theories predict parity violation in weak interactions [73], as confirmed by collider experiments [74] [2]. Intriguingly, measurements such as the four-point function of galaxies [17] or cosmic birefringence in the CMB [86] in cosmology also reveal signs of parity violation within the large-scale structure. This anomaly suggests the presence of something beyond the Λ CDM framework, possibly arising

³I am currently considering rewriting my entire code in PyTorch or JAX as they offer improved speed and performance compared to TensorFlow.

from exotic phenomena in the early universe, such as the coupling of the axion-like field⁴ with photons. However, for the robustness of these observations, additional sources of evidence for parity violation within the large-scale structure are needed. We propose the use of gravitational lensing as one such source of evidence. Employing an approach akin to cosmic birefringence, we advocate for the examination of observables like the angular power spectrum of EE , EB , and BB modes, in lieu of the CMB power spectrum. Traditional cosmic shear analyses lean on the Born approximation, which, without systematic errors, yields a vanishing B -mode in the convergence field. Hence, it may be imperative to delve into theories that extend beyond the Born approximation, akin to CMB lensing. A notable concern here is the limitation of information along the line of sight and the presence of systematics, primarily stemming from the limitations of photometric surveys. Nevertheless, I would like to explore this interesting topics in the near future.

The second avenue of our interest is the detection of primordial gravitational waves. This pursuit offers valuable constraints on various inflationary models through the tensor-to-scalar ratio. Direct measurements pose a formidable challenge, prompting researchers to explore alternative approaches. One potentially indirect method involves scrutinizing the B-mode spectrum of the CMB. The Japan Aerospace Exploration Agency's LiteBird mission [82] aims to detect the footprints of primordial gravitational waves in the CMB through a distinctive polarization pattern known as B-mode. Nevertheless, this endeavor confronts challenges such as instrumental noise and modeling the contamination from foreground sources, including synchrotron radiation, Sunyaev-Zeldovich effects, gravitational lensing, and galactic dust emissions, all of which contribute to the survey's noise. Another intriguing method leverages the large-scale structure, as proposed in the paper by Friedrich et al. (2018) [41], which outlines a strategy for measuring primordial non-Gaussianity using probability density function models of cosmic density fields.

In our quest to bridge the gap between these two cosmic eras, we are almost prepared to enter the golden age of cosmology.

⁴It is considered to exhibit axion-like behavior(axion-like particles(ALPs)), but it is not an axion itself.

Appendix A

Graph Spectral Theory

In this appendix, we delve into the fundamental concepts of defining essential transformations like convolution, Fourier transform, and Wavelet transform on a graph, which are essential technique for implementing Graph Convolutional Neural Networks (GCNNs). This section draws heavily from the insights provided by the review by Hammond et al. (2009) [57].

A.1 Graph Theory

Before delving into the definition of graph convolution, it's essential to establish the concept of the Laplace operator on a graph. Let's start by understanding what a graph is. In graph theory, a weighted undirected graph with non-negative edge weights is represented as a mathematical structure denoted as $\mathcal{G} = (\mathcal{E}, \mathcal{V}, w)$. Here, \mathcal{E} is the set of edges, \mathcal{V} is the set of vertices, and $w : E \rightarrow \mathbb{R}$ is the weight function [57]. For the scope of this master's thesis, I focus on dealing with HealPix maps, which can be depicted as an undirected graph with non-negative edge weights, leading us to consider this specific scenario. Now, let's introduce the three pivotal matrices associated with the graph.

- (1) **Adjacency Matrix:** This square matrix encapsulates the connections between vertices in a graph along with their respective weights. For a weighted graph \mathcal{G} , an element of the adjacency matrix \mathbf{A} can be defined as follows [57]:

$$\mathbf{A}_{ij} = \begin{cases} w(e) & \text{if } e \in E \text{ connects vertices } i \text{ and } j \\ 0 & \text{otherwise} \end{cases} . \quad (\text{A.1})$$

In the context of an undirected graph, this matrix is symmetric ($\mathcal{A}_{n,m} = \mathbf{A}_{m,n}$)..

- (2) **Degree Matrix:** This diagonal matrix encodes the degree of each vertex within the graph. The element \mathbf{D}_{ii} is computed as the sum of weights of all connected vertices:

$$\mathbf{D}_{i,j} = \sum_{j=1}^n a_{ij} . \quad (\text{A.2})$$

Here, n represents the number of connected vertices.

- (3) **Non-normalized Laplacian:** This matrix, denoted as \mathcal{L} , is defined as the difference between the degree matrix \mathbf{D} and the adjacency matrix \mathbf{A} :

$$\mathcal{L} = \mathbf{D} - \mathbf{A} . \quad (\text{A.3})$$

This non-normalized graph Laplacian \mathcal{L} is symmetric and positive semi-definite, suitable for an undirected graph with non-negative edge weights. An intriguing property of this matrix can be derived from a completion of the square complement. For any $f \in \mathbb{R}^N$, \mathcal{L} satisfies:

$$f^T \mathcal{L} f = \frac{1}{2} \sum_{i,j=1}^n a_{i,j} (f_i - f_j)^2 . \quad (\text{A.4})$$

This property allows us to draw an analogy with the standard Laplace-Beltrami operator in continuous variables. By considering a signal that attenuates based on the square of the distance $d(x_i, x_j)$ between two grid points x_i and x_j , the adjacency matrix can be expressed as $a_{ij} = \frac{1}{d(x_i, x_j)}$. Equation (A.4) takes the form:

$$f^T \mathcal{L} f = \frac{1}{2} \sum_{i,j=1}^n \left(\frac{f_i - f_j}{d(x_i, x_j)} \right)^2 . \quad (\text{A.5})$$

This expression allows us to make an analogy with the standard Laplace-Beltrami operator of continuous variables, thereby interpreting the graph Laplacian as the discrete counterpart of the Laplace-Beltrami operator.

An alternative version of the graph Laplacian is the symmetric graph Laplacian, tailored for graphs featuring non-negative weights. It is expressed as follows¹:

$$\mathcal{L}^{\text{sym}} = \mathbf{D}^{-\frac{1}{2}} \mathcal{L} \mathbf{D}^{-\frac{1}{2}} . \quad (\text{A.6})$$

An advantageous attribute of the normalized Laplacian is that its eigenvalues are confined within the interval $[0, 2]$, rendering it a preferable choice over the unnormalized graph Laplacian for applications such as graph neural networks, due to the reduction in computational complexity [108]².

A.2 Graph Spectral Theory

Now, we aim to establish the concept of convolution on the graph. Given that the graph Laplacian is a real symmetric and positive semi-definite matrix, the Laplacian matrix for

¹In this equation, D^{-1} represents the Moore-Penrose inverse of matrix D .

²Notably, **DeepSphere** employs the normalized Laplacian for computing the Fourier basis of graphs [96].

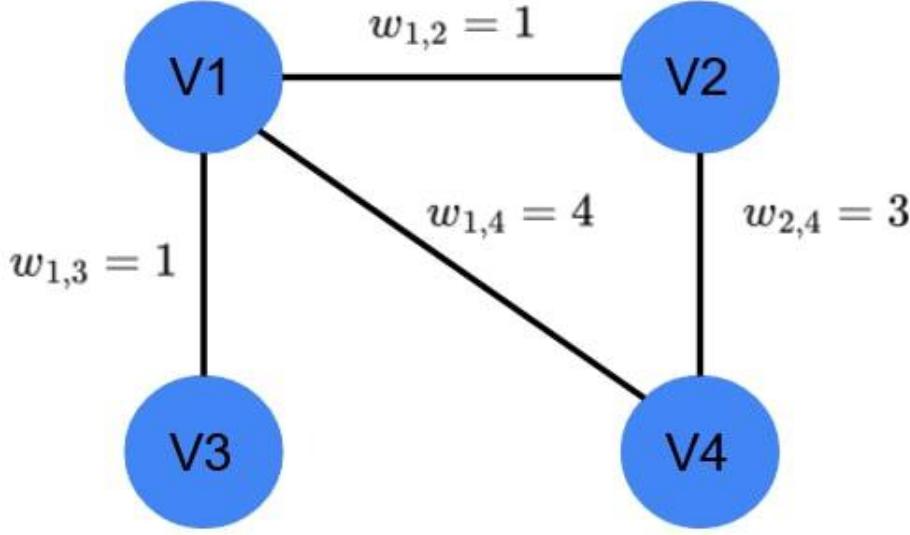


Figure A.1: The provided illustration serves as an exemplar of an undirected weighted graph. Here, the set V pertains to the vertices, while w signifies the weight associated with each connection. The numerical labels correspond to the numbering of both the columns and rows in the subsequent adjacency matrix below. Notably, in the context of an undirected graph, the principle of symmetry applies to the weights, resulting in $w_{i,j} = w_{j,i}$. This symmetry leads us to derive the adjacency matrix \mathbf{A} , the degree matrix \mathbf{D} , and the graph Laplacian \mathcal{L} , as elaborated in the following:

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 1 & 4 \\ 1 & 0 & 0 & 3 \\ 1 & 0 & 0 & 0 \\ 4 & 3 & 0 & 0 \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} 6 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 7 \end{pmatrix}, \quad \mathcal{L} = \begin{pmatrix} 6 & -1 & -1 & -4 \\ -1 & 4 & 0 & -3 \\ -1 & 0 & 1 & 0 \\ -4 & -3 & 0 & 7 \end{pmatrix}.$$

an undirected graph with non-negative weights and N vertices can be decomposed through eigendecomposition as follows [57]:

$$\mathcal{L}\mathbf{U} = \Lambda\mathbf{U}. \quad (\text{A.7})$$

In this equation, $\mathbf{U} = [u_1, \dots, u_N]$ represents an orthonormal matrix of eigenvectors, and Λ is a diagonal matrix containing the eigenvalues.

To elucidate the significance of this eigendecomposition, let's draw a parallel with an analogous scenario. The graph Laplacian can be perceived as the discrete counterpart of the Laplace-Beltrami operator in continuous 1-D space. Consider the continuous equation resulting from the eigendecomposition in (A.7):

$$\frac{d}{dx^2}u(x) = \lambda u(x). \quad (\text{A.8})$$

Here, λ is a real scalar value and u is a real function of the real value x . Solving this differential equation can be achieved through the separation of variables, yielding:

$$u(x) \propto \exp(-2\pi ikx) . \quad (\text{A.9})$$

Where i denotes the imaginary unit and k is the frequency. This $u(x)$ can be interpreted as an eigenfunction. Reflecting on the Fourier transform, it emerges as the product of the eigenfunction $u(x)$ and the signal function $f(x)$:

$$\hat{f}(k) = \mathcal{F}(f(x)) = \langle f(x)u(x) \rangle = \int f(x)\exp(-2\pi ikx)dx . \quad (\text{A.10})$$

Drawing from this analogy, we can define the graph Fourier transform as the product of a discrete signal f on the graph and the eigenvector U :

$$\hat{f}(l) = \mathcal{F}(f(n)) = \sum_{n=1}^N f(n)u_l^*(n) = U^T f(n) . \quad (\text{A.11})$$

In this equation, u_l signifies an eigenvector of the graph Laplacian. Correspondingly, we can define the inverse graph Fourier transform:

$$f(n) = \mathcal{F}^{-1}(\hat{f}(n)) = \sum_{l=0}^{N-1} \hat{f}(l)u_l(n) = U\hat{f}(l) . \quad (\text{A.12})$$

Similar to the Fourier transformation, the Parseval relation holds for the graph Fourier transformation as well [57], for any f and g belonging to \mathbb{R}^N [57]:

$$\langle f, g \rangle = \langle \hat{f}, \hat{g} \rangle . \quad (\text{A.13})$$

Now, our focus shifts towards defining the localized filter. In the Euclidean domain, the fast Fourier transform readily provides a localized variant of the Fourier transform. However, in non-Euclidean domains, this is not as straightforward. Wavelets, on the other hand, offer a distinct advantage over the Fourier transform by concurrently localizing signals in both the frequency and time domains.

For a given signal f , the wavelet coefficient at scale s and position a is captured by the wavelet transform equation [57]:

$$W_f(s, a) = \int_{-\infty}^{\infty} \frac{1}{s} \psi^* \left(\frac{x-a}{s} \right) f(x)dx . \quad (\text{A.14})$$

Here, ψ represents a chosen mother wavelet.

To extend this notion to the realm of graphs, Hammond et al. (2009)[57] derived the wavelet transform on the graph. The spectral graph wavelet transform is expressed using the wavelet kernel g , which is a function of the graph Laplacian's eigenvalue λ_l :

$$W_f(t, n) = \sum_{l=0}^{N-1} g(t\lambda_l) \hat{f}(l) u_l(n) . \quad (\text{A.15})$$

In this equation, $\hat{f}(l)$ denotes the l -th element of the graph Fourier transform of signal f , and $u_l(n)$ represents the l -th eigenvector component associated with vertex n . This formulation enables us to extend the concept of wavelet transforms to graph-structured data.

Appendix B

Robustness of Inference Pipeline

The neural network model involves a highly complicated and complex statistical framework, and as such, the results presented in Section 8 may exhibit fluctuations even under identical settings¹. These fluctuations can introduce additional biases or widen the variance. Estimating the error of a neural network is a challenging task, as there is currently no universally accepted method for its determination. To address this challenge, we embark on a robustness assessment of our methodology. First, we investigate the stability of our approach by conducting experiments with 300 different random realizations of Gaussian and lognormal fields as test datasets. This extensive set of realizations allows us to gauge whether our results remain consistent. Furthermore, we perform a robustness check on our training pipeline. We generate new training datasets from the same simulation setup explained in Chapter 5, merely altering the random number seed. We subsequently repeat the same analysis to determine whether consistent contours are obtained across different training datasets.

B.1 Robustness of GCNN Model

In our quest to assess the robustness of our GCNN models, which generated the results detailed in Chapter 8, we embarked on a rigorous examination. To address this inquiry, we constructed test datasets using the same simulation settings, with the only variation being the random seeds employed during the generation of random fields. Subsequently, we replicated the same analysis procedures employed in Chapter 8, and in this section, we present several plots that serve to evaluate the model's robustness. For clarity, it is essential to distinguish between the test datasets used in Chapter 8, referred to as 'Test Datasets 1', and the newly generated datasets, denoted as 'Test Datasets 2'.

Figure B.1 provides a visual representation of the predicted mean and variance for both the new and old test datasets. Notably, the data points exhibit slight upward shifts. To quantify these differences in position for the predicted mean and variance, we refer to Figure B.2. Regarding the predicted mean positions, we observed fractional differences ranging

¹Batch size...

from 0.3% to 0.05%. In terms of variance, the maximum observed discrepancy is 5%.

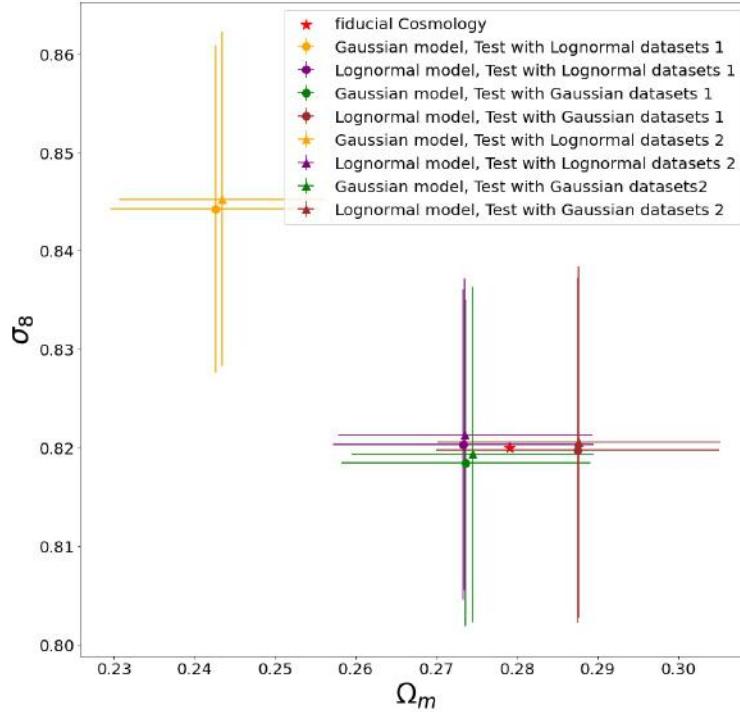


Figure B.1: Analogous to the plot in Figure 8.3, the colors in this plot correspond to those in Figure 8.3. The upper arrow points represent new test datasets, each of which is generated by changing the seed of the random number for all realizations.

Now, let's examine how the other outputs that contribute to the estimation of parameter covariance differ between test data 1 and test data 2. To visualize these variations, we present Figure B.3, which utilizes the Gaussian model and Gaussian test datasets.

In this figure, we observe a shift in the contours due to the different predicted means between the two test datasets. However, the constraining power remains nearly consistent when considering the errors introduced during the training of the GCNN model, as discussed in the subsequent section. Importantly, similar results are also observed across various combinations of test datasets and GCNN models.

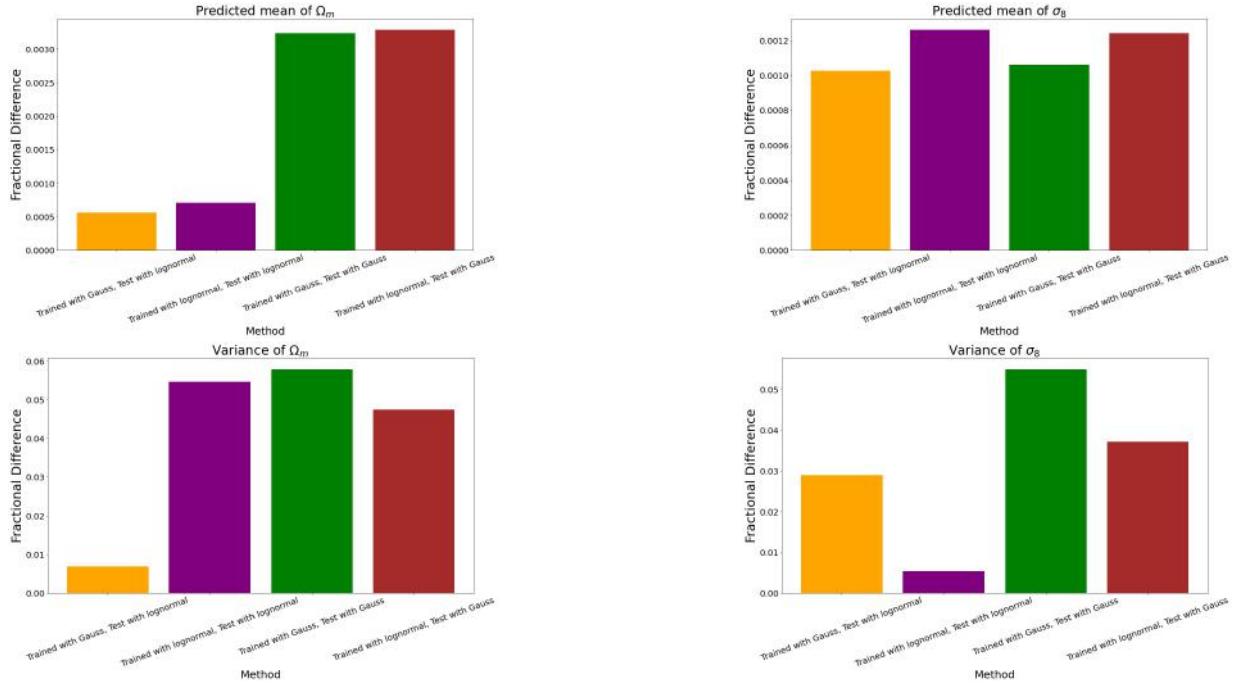


Figure B.2: These figures illustrate the fractional differences between test datasets 1 and test datasets 2. The upper figures display the positions of the predicted mean, while the lower figures represent the variances. The colors in these figures correspond to those in figure B.1.

B.2 Robustness of Training of GCNN Model

In this section, we aim to address the uncertainties arising from the training process. To accomplish this, we initiated the process by generating a new set of training data. These data were created using the same simulation pipeline described in Chapter 5, with the sole alteration being the random number seed used during data generation. Subsequently, we trained a GCNN model using this new dataset, adhering to the same experimental settings outlined in Chapter 7. Our objective is to replicate the analysis performed in Chapter 8 in a manner analogous to the original approach. To ensure clarity, we will refer to the model used in Chapter 8 as 'Model 1', while the newly trained model will be referred to as 'Model 2'.

To begin our investigation, we examine the loss functions associated with two distinct training datasets. In Figure B.4, we compare the loss functions of the GCNN model trained on training datasets 1 and training datasets 2. Surprisingly, the loss curves exhibit different evolution patterns throughout the training process, resulting in distinct final loss values. Notably, for both the lognormal and Gaussian models, model 1 consistently achieves lower final loss values when compared to model 2, even when trained on the same datasets. This observation holds true across identical training datasets. As a result, we conclude that our chosen hyperparameters and model architecture may not be optimized for this specific

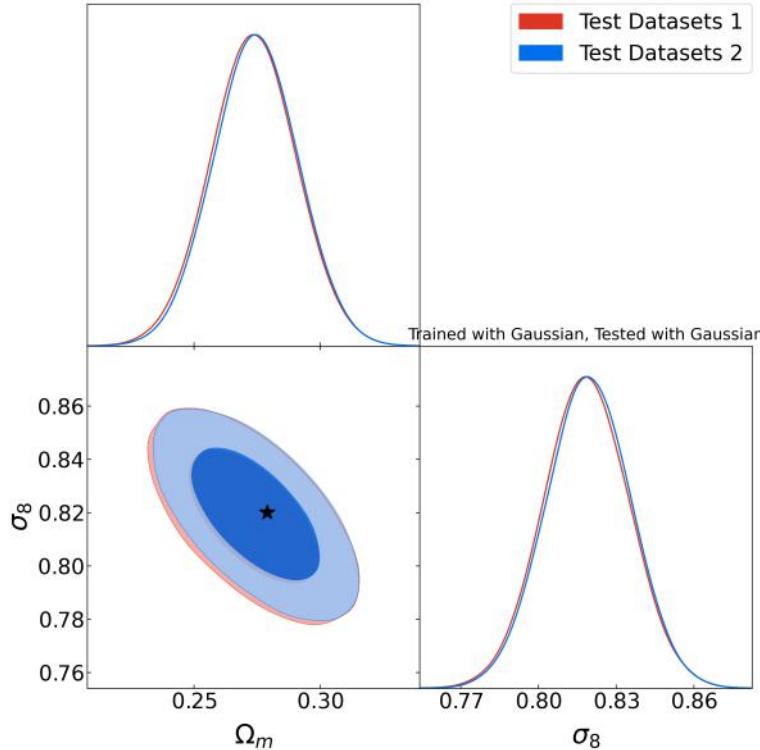


Figure B.3: This plot depicts contours created using the Gaussian test datasets 1 and Gaussian test datasets 2 when employing the Gaussian GCNN model and averaged covariance. The contours are centered around the predicted mean.

task. It's worth mentioning that employing a larger batch size has the potential to yield a more stable evolution of the loss function. However, it's important to note that this approach typically requires additional GPU memory. Next, we explore the relationship between the difference in loss and its impact on our estimation of cosmological parameters. To visualize this effect, we have created Figure B.5, following a similar approach as in Figure 7.9. Notably, the edge effect becomes more pronounced for models with higher loss values.

As evident in Figure B.4, our training process is inherently a generalization process. Consequently, as we decrease the loss function over successive epochs, the estimated model parameters will naturally converge to the true range of parameter values. We then proceed to evaluate the model predictions using the test datasets. Figure B.8 presents a visual representation of the predicted mean and variance, allowing for a comparison between two sets of models, following a similar methodology as shown in Figure B.1. Interestingly, unlike the scenario where the test datasets were changed, altering the training datasets has a notable impact. In the case of Model 1, all combinations, except for the Gaussian model tested with lognormal fields and the Gaussian model tested with T17 convergence fields, successfully recover the fiducial cosmology within the $1-\sigma$ confidence interval. However, for Model 2, the Gaussian model only covers the fiducial cosmology when tested with Gaussian

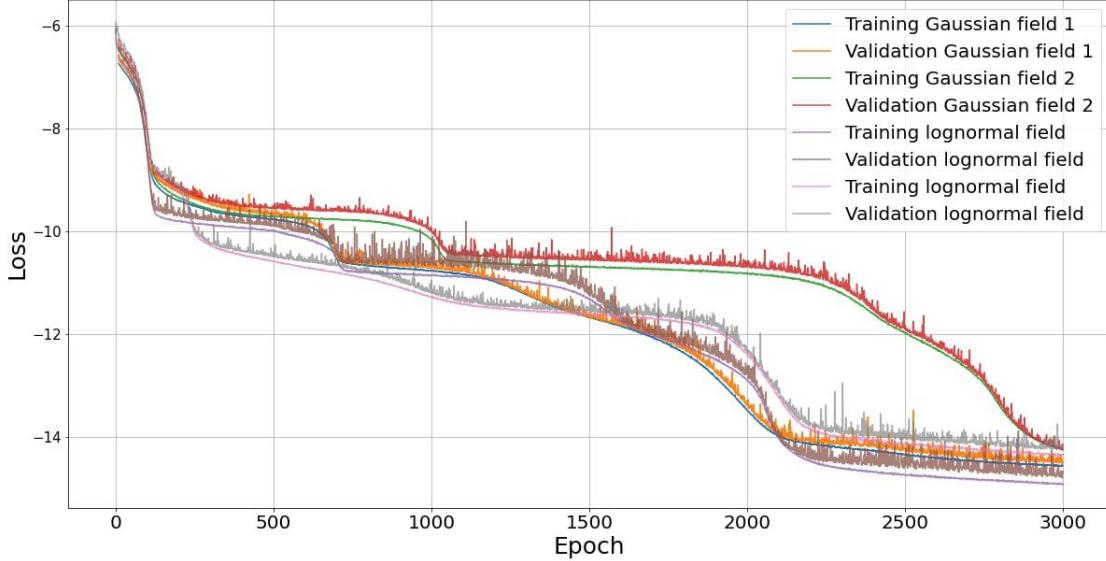


Figure B.4: This plot presents a comparison of the loss function between the model trained with training datasets 1 and the model trained with training datasets 2. It serves as an analogous representation to that shown in Figure 7.5

fields, and the lognormal model achieves this when tested with lognormal fields and T17 projected density fields. This outcome aligns with my initial expectations, although it remains challenging to interpret why a higher loss led to the desired results, given our current understanding.

Next, our aim is to estimate the bias in our results. Figure 8.5 illustrates the bias indicators using a methodology consistent with that presented in Figure 8.5. In the Gaussian model, we observe that Gaussian fields exhibit the lowest bias, followed by lognormal fields and T17 projected density fields. Conversely, in the lognormal model, lognormal fields display the least bias, followed by T17 projected density fields and Gaussian fields. Notably, it's interesting to observe that despite a higher loss in Gaussian model 2 compared to Gaussian model 1, the former yields lower bias. Conversely, Gaussian model 1 provides superior estimations in the lognormal model and T17 simulations. To gain a deeper understanding of these results, we plan to explore an interpretation of filters in our future work. Finally, our attention turns to the evaluation of the posterior distribution contours. Figures B.10 and B.11 depict these contours, utilizing the averaged covariance, mirroring the analysis conducted in Chapter 8.

In the case of the Gaussian model, it's noteworthy that the contour produced using T17 simulation test datasets extends beyond the predefined range, leading to its omission. Furthermore, the posterior distribution derived from lognormal test datasets only marginally intersects the edge of the prior distribution due to a considerable bias. Comparatively, the contour produced using Gaussian fields is slightly larger than those generated by the lognormal fields. Within the lognormal model, both the contours derived from lognormal fields and T17 simulations successfully encompass the fiducial cosmology. However,

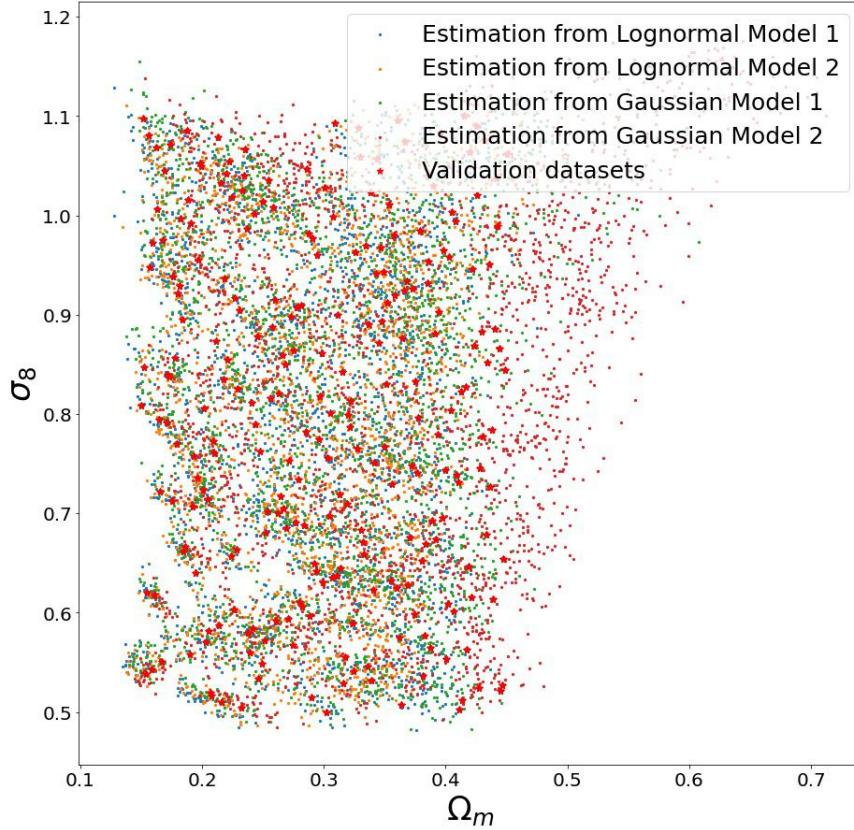


Figure B.5: This plot illustrates the predictions generated by the GCNN models, following an approach to that of Figure 7.9.

the posterior distribution obtained using Gaussian fields exhibits biases in both the σ_8 and Ω_m directions. Intriguingly, the contour produced with Gaussian fields provides more significant constraints when compared to the contours generated by lognormal fields and T17 simulations. In conclusion, our findings suggest that biased contours tend to provide more restrictive constraints than their unbiased counterparts. This observation may be attributed to the bias-variance trade-off, although further analysis is needed to substantiate this hypothesis.

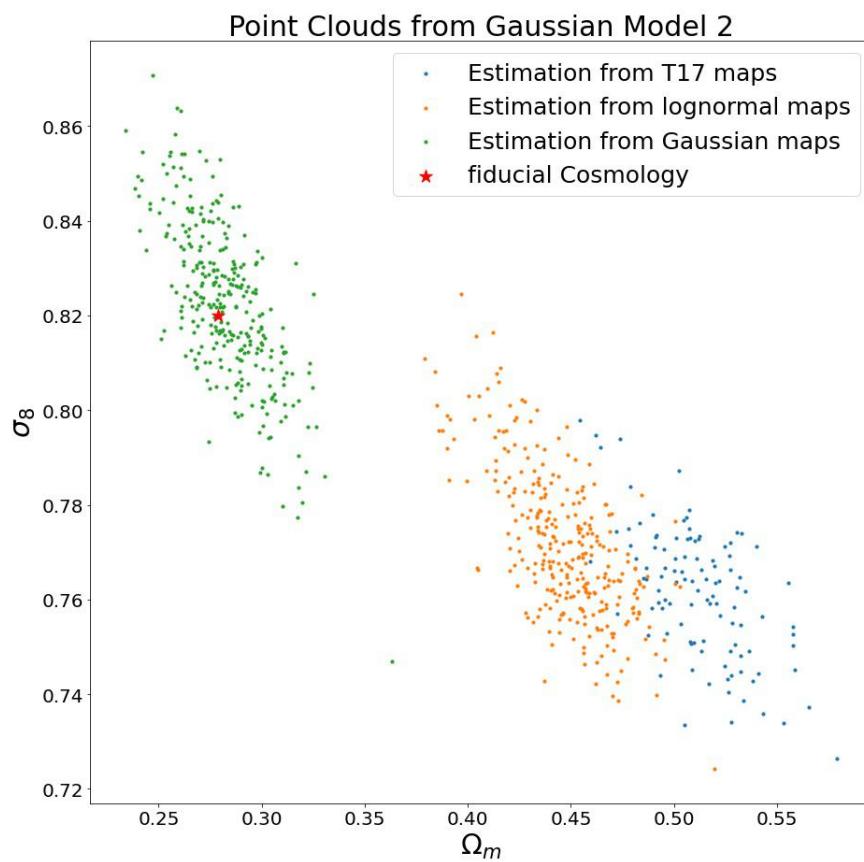


Figure B.6: This visualization showcases the predictions generated by the Gaussian model 2 with test datasets 1. The descriptions for each set of points are analogous to those in Figure 8.1.

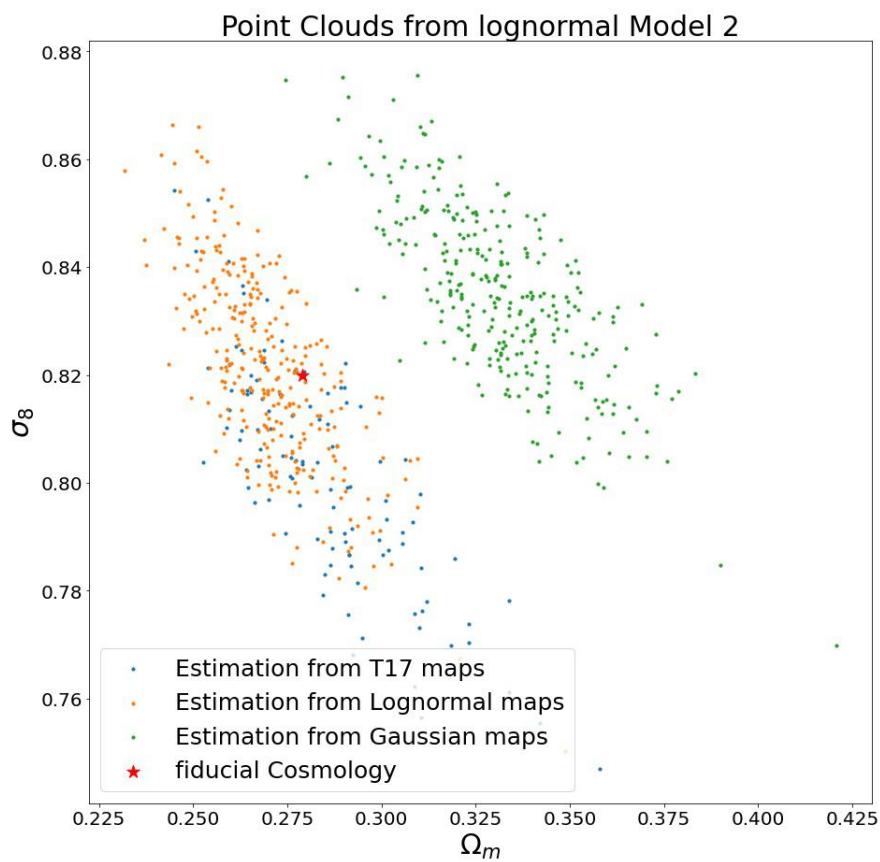


Figure B.7: This visualization showcases the predictions generated by the lognormal model 2 with test datasets 1. The descriptions for each set of points are analogous to those in Figure 8.1.

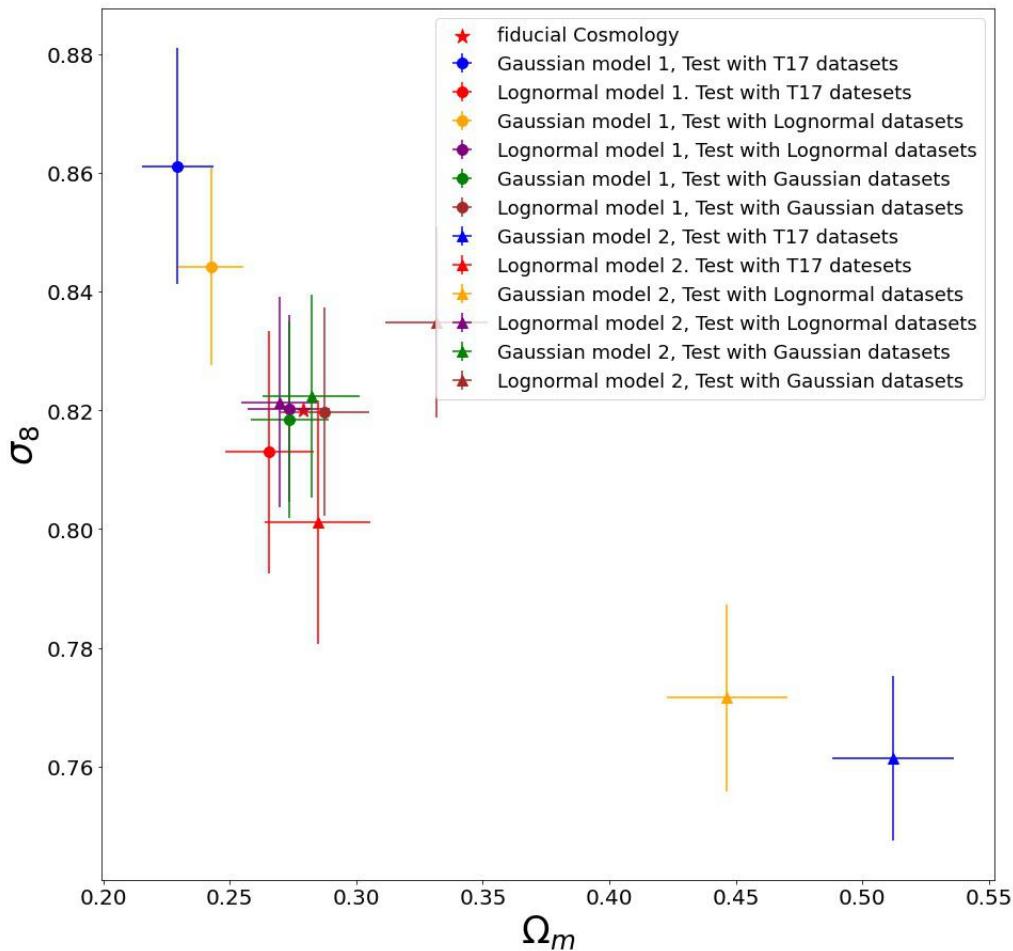


Figure B.8: The colors in this plot correspond to those in Figure 8.3. The upper arrow points represent model 2.

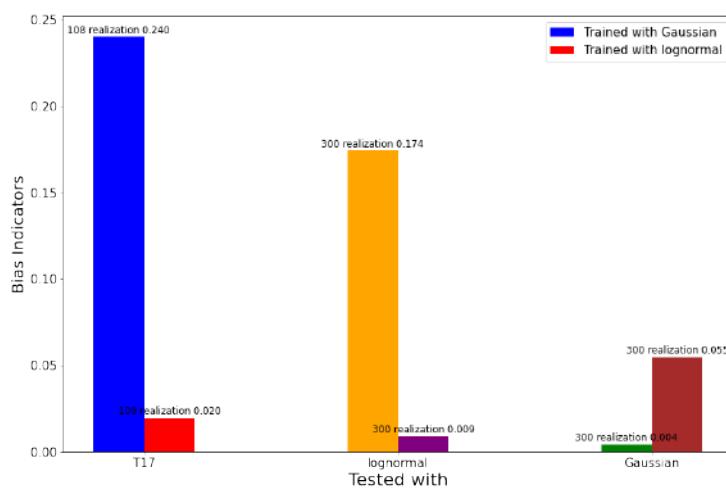


Figure B.9: The caption is analogous to 8.5.

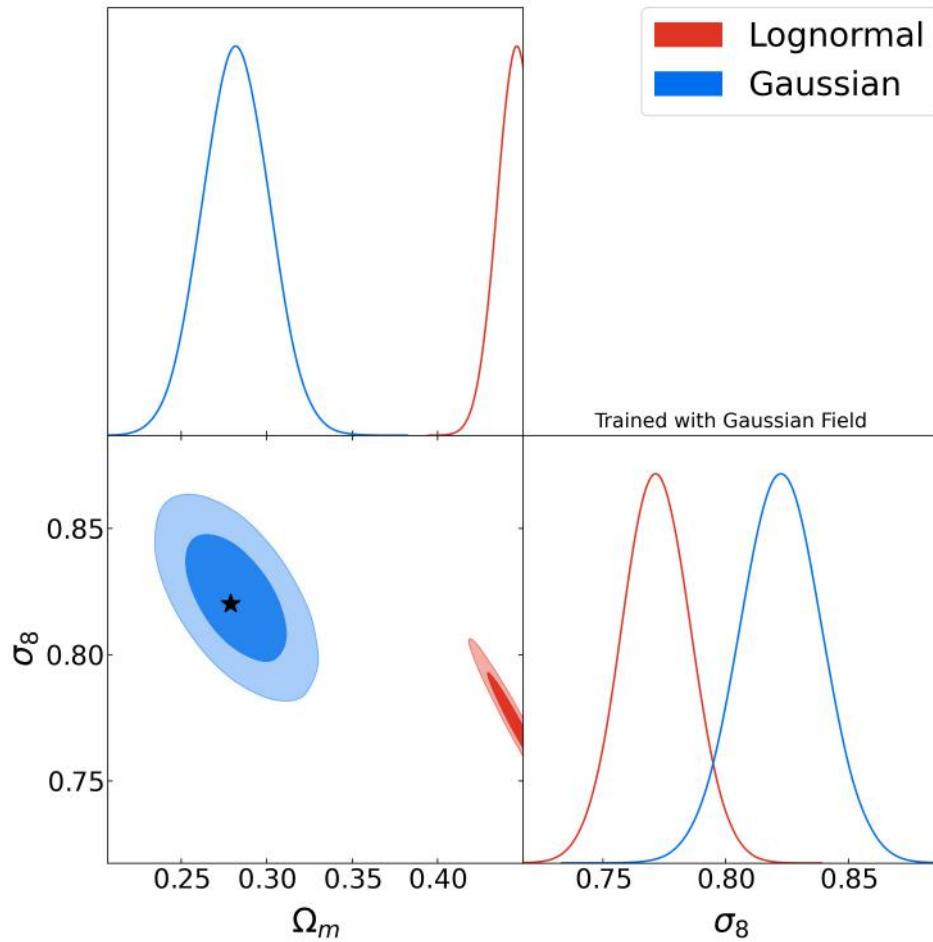


Figure B.10: A contour plot is generated using the averaged covariance and Gaussian model 2. However, in the case of testing with T17 simulations, the results exceeded the predefined range, leading to the omission of this contour.

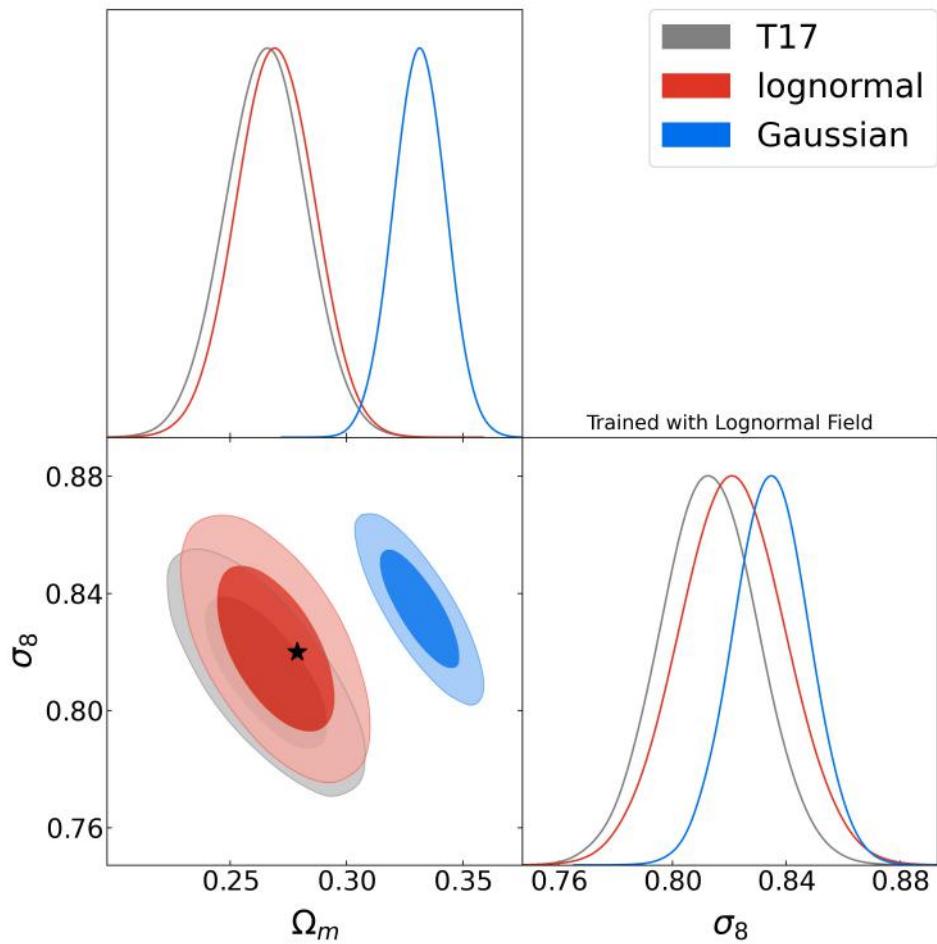


Figure B.11: A contour plot is generated using the averaged covariance and Lognormal model 2.

Appendix C

The contours of S_8 vs. Ω_m

In cosmology, S_8 serves as an alternative parameter for characterizing the amplitude of matter fluctuations in the universe. Opting for S_8 over the singular use of σ_8 is a strategic choice in certain cases. This preference arises from the valuable role played by S_8 in alleviating degeneracies that commonly manifest between two vital parameters: Ω_m (the present-day matter density parameter) and σ_8 itself, particularly within the framework of two-point statistics. S_8 embodies a more comprehensive approach, as it takes into account both the amplitude of matter fluctuations (σ_8) and the matter density of the universe (Ω_m). This combined parameter, S_8 , proves exceptionally robust in constraining the growth of cosmic structure. Furthermore, it exhibits reduced sensitivity to variations in other cosmological parameters, making it another choice in the realm of precision cosmology. Given these advantages, this appendix embarks on a reevaluation of the analysis, utilizing S_8 as the parameter of choice.

C.1 Conversion from σ_8 to S_8

In our analysis, we aim to convert the cosmological parameter σ_8 to the parameter S_8 , as predicted by the GCNN pipeline. This conversion is essential for our study, as σ_8 is the predicted quantity, while S_8 is the parameter of interest. The definition of S_8 is given by the following equation:

$$S_8 = \sigma_8 \sqrt{\frac{\Omega_m}{0.3}}. \quad (\text{C.1})$$

where Ω_m and σ_8 represent the first two outputs of the GCNN pipeline, making the conversion straightforward. To account for the parameter covariance matrix in this transformation, we utilize error propagation. The transformation of the parameter covariance matrix \mathbf{C} is expressed as:

$$\mathbf{C}(\Omega_m, S_8) = \mathbf{J}(\Omega_m, S_8)\mathbf{C}(\Omega_m, \sigma_8)\mathbf{J}^T(\Omega_m, S_8), \quad (\text{C.2})$$

where \mathbf{J} represents the Jacobian matrix, which is computed as follows:

$$J = \begin{pmatrix} \frac{\partial \Omega_m}{\partial \Omega_m} & \frac{\partial \Omega_m}{\partial S_8} \\ \frac{\partial S_8}{\partial \Omega_m} & \frac{\partial S_8}{\partial \sigma_8} \end{pmatrix} = \begin{pmatrix} 1 & -0.9S_8^2\sigma_8^{-3} \\ \frac{1}{2\sqrt{0.3}}\frac{\sigma_8}{\sqrt{\Omega_m}} & \sqrt{\frac{\Omega_m}{0.3}} \end{pmatrix}. \quad (\text{C.3})$$

C.2 Clouds of Points in $S_8 - \Omega_m$ Plane

In line with the preceding section, we begin our discussion with the analysis of point clouds. As depicted in Figure C.1, we observe a cluster of points plotted within the S_8 - Ω_m plane.

In the Gaussian model, there is a notable change in the direction of correlation compared to Figure C.1a. However, what makes for an intriguing observation is the behavior within the lognormal model, as illustrated in Figure C.1b. Visually, it was challenging to distinguish whether the cases of a Gaussian field and T17 convergence are biased or not. However, when viewed within the S_8 - Ω_m plane, this difference becomes significantly more pronounced. Notably, only the cluster corresponding to the unbiased scenario is evident when tested with a lognormal field.

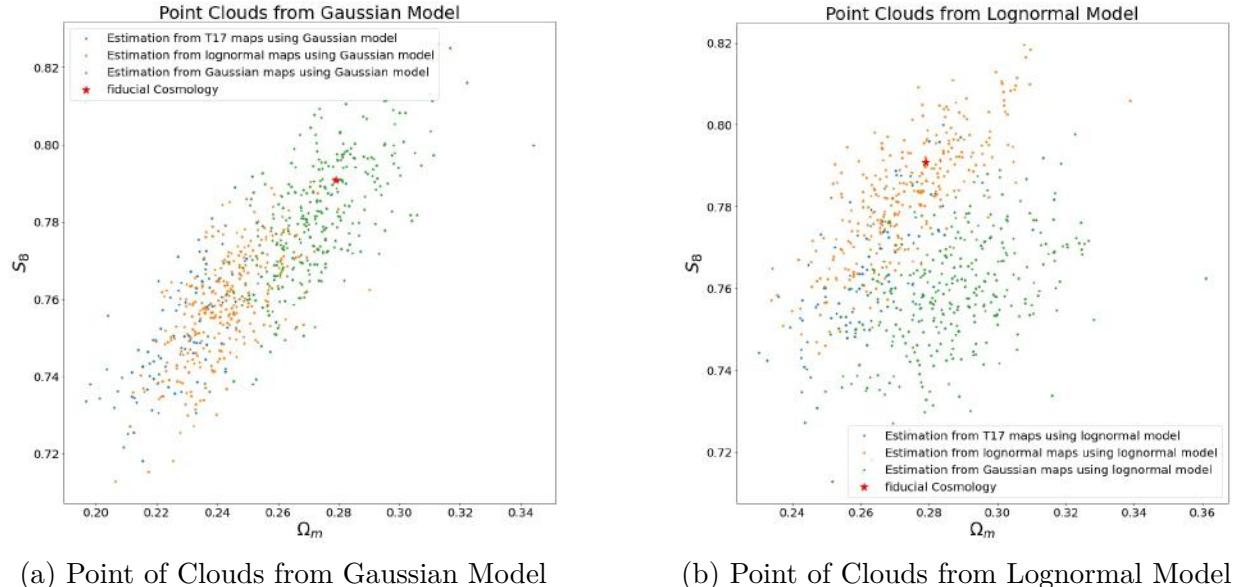


Figure C.1: Analogous to figure C.1a and C.1b, the point clouds are plotted in the S_8 - Ω_m plane.

We then proceed with the Frequentist analysis. In Figure C.2, the error bars are centered at the predicted mean, with their width estimated from the variance. Notably, unlike Figure 8.3, only the Gaussian-trained model, when tested with Gaussian fields, and the lognormal-trained model, when tested with lognormal fields, encompass the fiducial cosmology within the $1-\sigma$ confidence interval.

However, when considering T17 convergence fields and Gaussian fields within the lognormal model, both scenarios cover the fiducial cosmology in terms of Ω_m . In contrast,

in the direction of S_8 , they fall outside the $1-\sigma$ confidence interval. This particular case highlights a more pronounced bias in the S_8 parameter compared to σ_8 . We then pro-

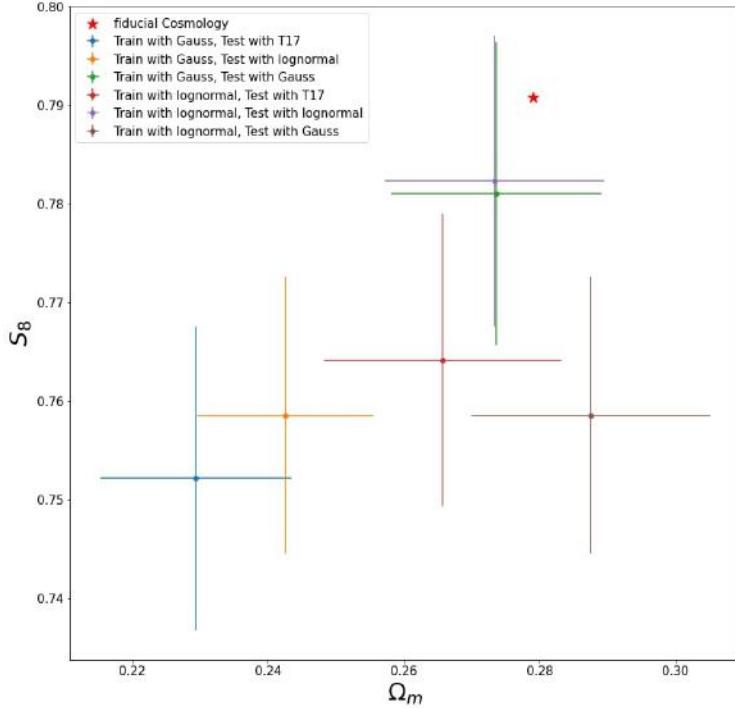


Figure C.2: This figure is plotted in an analogous fashion to Figure 8.3, within the S_8 - Ω_m plane.

ceed to assess bias, utilizing the parameter S_8 . In Figure C.3, we evaluate bias through the Euclidean distance between the predicted mean and the fiducial cosmology, mirroring the analysis presented in Figure 8.5. Interestingly, when employing S_8 as our bias metric, cases that exhibit bias are notably more pronounced compared to the traditional σ_8 metric. The overall bias indicator remains relatively stable for T17 convergence maps and lognormal fields as test datasets in the context of a Gaussian-trained GCNN model. However, for other scenarios, we observe a reduction in the bias indicator, particularly when employing T17 convergence maps and Gaussian fields as test datasets, both within the lognormal model. Notably, when employing σ_8 as the metric, the lognormal model using lognormal test datasets exhibits less bias compared to the Gaussian model using Gaussian test datasets. However, when utilizing S_8 as the bias metric, this relationship is reversed. Furthermore, we observe a significant reduction in the difference between the lognormal model using T17 convergence fields as test datasets and the lognormal model using Gaussian fields as test datasets.

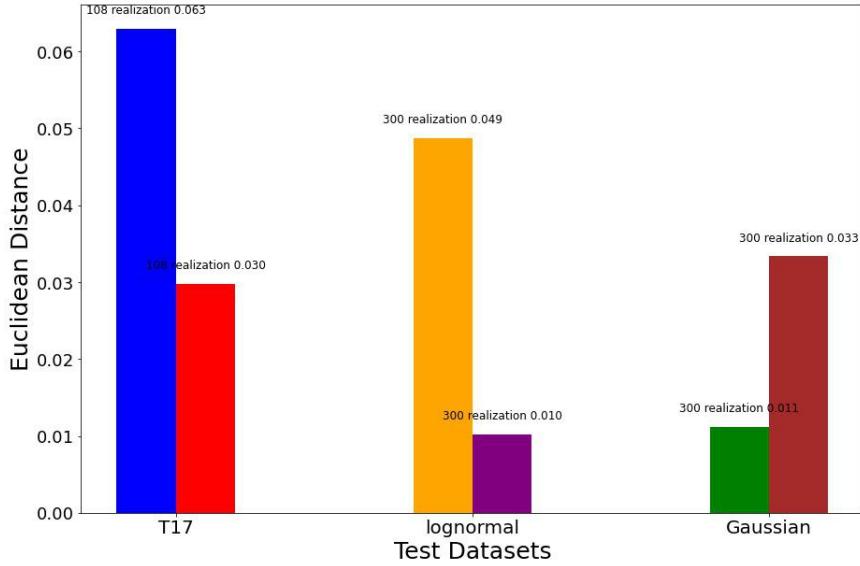


Figure C.3: The bias indicator is plotted using S_8 analogously to 8.4.

C.3 Posterior Distributions in S_8 - Ω_m Plane

In the final stage of our analysis, we aim to assess the posterior distribution within the S_8 - Ω_m plane. As in previous sections, we explore two different approaches to obtain parameter covariances. One method involves employing sample covariance method on the point clouds, while the other entails calculating covariance matrices for each realization using Equation (C.3), followed by averaging these matrices.

Figure C.4 presents contour plots generated using sample covariance matrices. In the Gaussian model, a noticeable shift in the orientation of contours can be observed when compared to Figure 8.7. However, the overall shape of the contours remains relatively consistent. In the lognormal model, the alignment of each contour undergoes changes, with the contour derived from tests conducted solely with lognormal fields exhibiting a more slender shape, while the contours derived from other scenarios appear more rounded. In both cases, only the contours derived from the best models—those obtained from lognormal fields in the case of the lognormal model and from Gaussian fields in the case of the Gaussian model—successfully encompass the fiducial parameter within the $1-\sigma$ credible interval.

Subsequently, we turn our attention to Figure C.5. Here, the contours exhibit a more rounded shape and demonstrate reduced correlation between the two parameters. Surprisingly, in the Gaussian model, the contours from all types of test datasets successfully encompass the correct cosmological parameters within the $1-\sigma$ credible region. In the lognormal model, the contours derived from lognormal and T17 scenarios appear visually larger, while the Gaussian contour, except for a slight inclination, exhibits a similar visual representation. Finally, we turn our attention to the evaluation of the figure of merit. In the realm of 2-point statistics, the figure of merit for the S_8 - Ω_m parameter pair is often enhanced by breaking the degeneracy between these two parameters. It raises the question

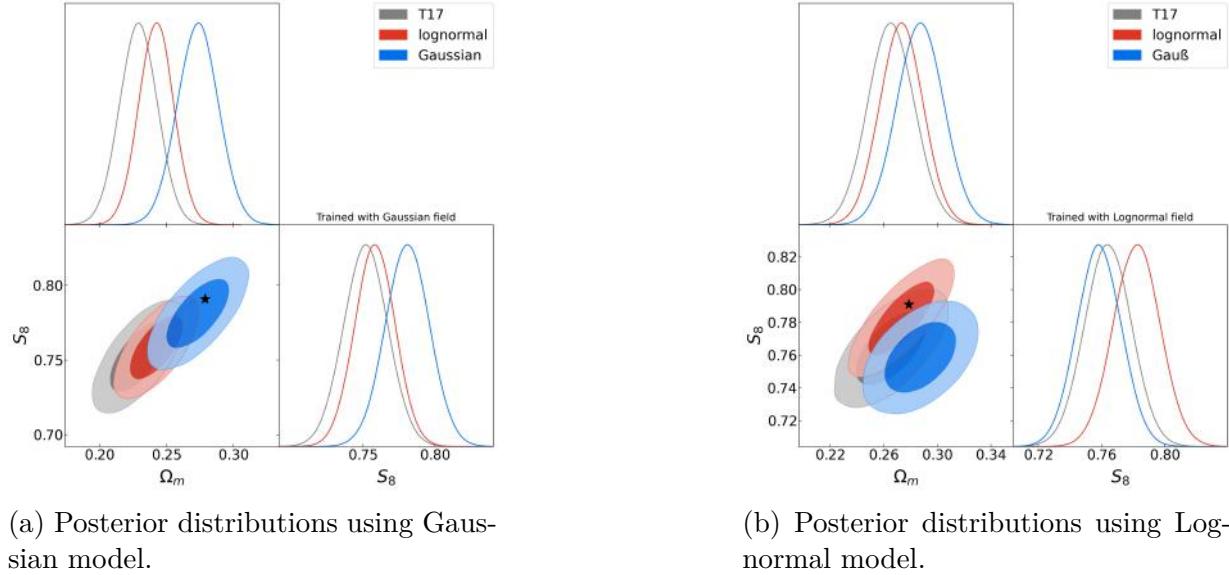


Figure C.4: This contour plot visualizes the credible interval of the posterior distribution within the S_8 - Ω_m plane, employing sample covariances.

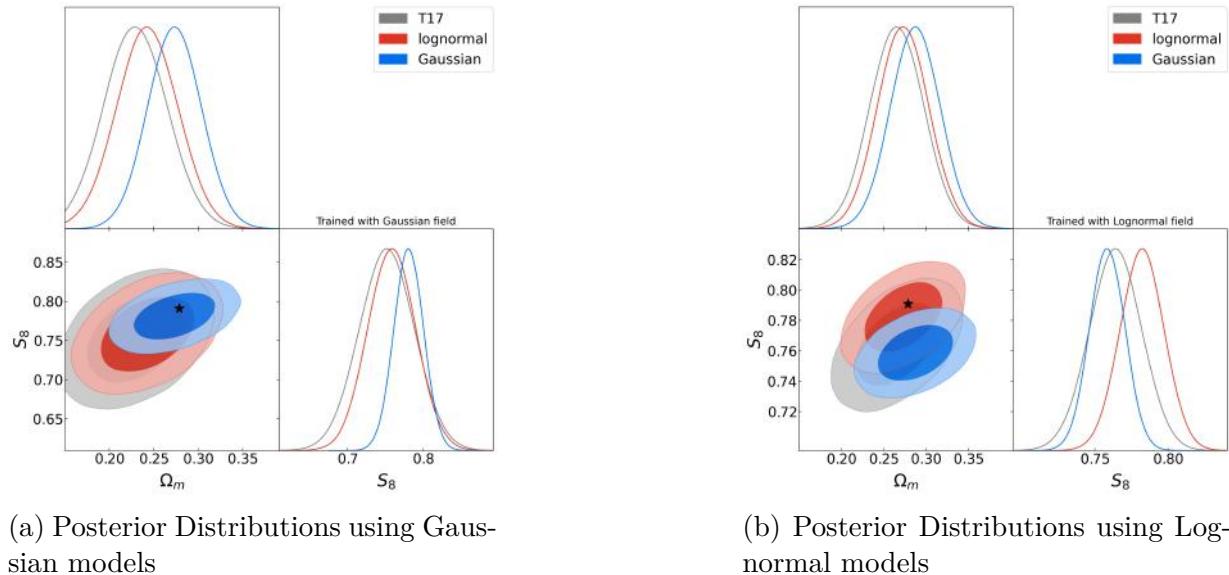


Figure C.5: This contour plot illustrates the credible interval of the posterior distribution within the S_8 - Ω_m plane, employing averaged covariances.

of how this principle translates into field-based inference.

Figure C.7 and Figure C.6 depict the figure of merit using bar plots, employing the same format as previously. When utilizing the averaged covariance matrix, we observe a significant reduction of nearly half in the figure of merit for each case when transitioning from σ_8 to S_8 as the parameter of interest. Conversely, when utilizing the sample covariance

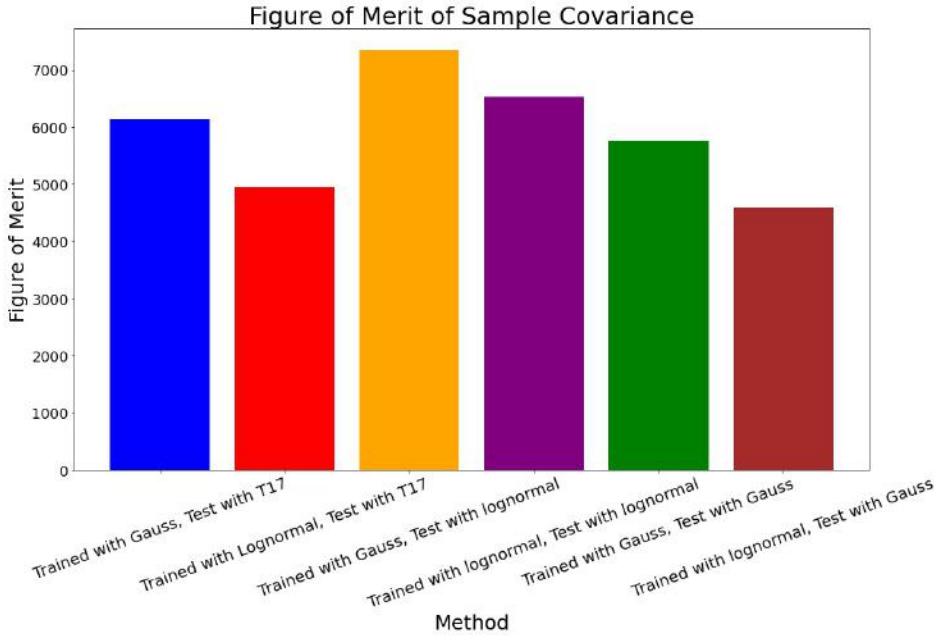


Figure C.6: This bar plot illustrates the figure of merits of sample covariance matrix of S_8 and Ω_m .

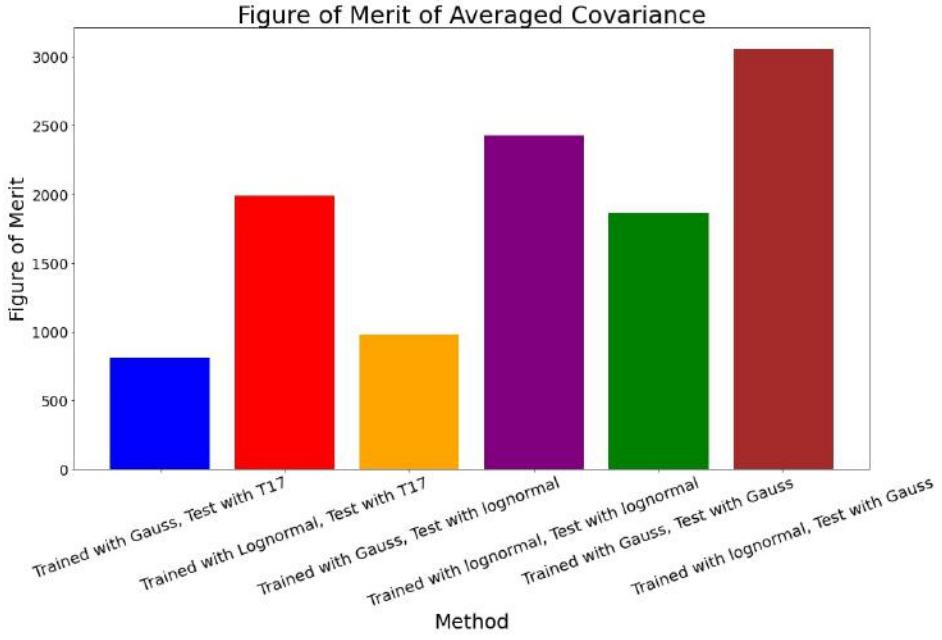


Figure C.7: This bar plot illustrates the figure of merits of averaged covariance matrix of S_8 and Ω_m .

matrix, the figure of merit experiences a slight increase, albeit with minimal observable differences in my specific case.

Bibliography

- [1] Abdalla E., et al. Cosmology intertwined: A review of the particle physics, astrophysics, and cosmology associated with the cosmological tensions and anomalies, 2022. <https://ui.adsabs.harvard.edu/abs/2022JHEAp..34...49A>.
- [2] Abe, K., et al., The Belle Collaboration. Observation of large cp violation in the neutral b meson system. *Physical Review Letters*, 87, 2001. <https://doi.org/10.1103%2Fphysrevlett.87.091802>.
- [3] Akiba T., Sano S., Yanase T., Ohta T., Koyama M. Optuna: A next-generation hyperparameter optimization framework, 2019. <https://arxiv.org/abs/1907.10902>.
- [4] Alsing J., Charnock T., Feeney S., Wandelt B. Fast likelihood-free cosmology with neural density estimators and active learning. *Monthly Notices of the Royal Astronomical Society*, 4488, 2019. <https://doi.org/10.1093%2Fmnras%2Fstz1960>.
- [5] Asgari M., et al. KiDS-1000 cosmology: Cosmic shear constraints and comparison between two point statistics. *Astronomy and Astrophysics*, 645, 2021. <https://doi.org/10.1051%2F0004-6361%2F202039070>.
- [6] Bartelmann M., Schneider P. Weak gravitational lensing. *Physics Reports*, 340, 2001. <https://doi.org/10.1016%2Fs0370-1573%2800%2900082-x>.
- [7] Barthelemy A. Modélisation théorique d'effets non-linéaires sur la statistique des champs faibles d'astigmatisme cosmique. 2021. <https://www.theses.fr/2021SORUS357>.
- [8] Barthelemy A., Codis S., Bernardeau F. Probability distribution function of the aperture mass field with large deviation theory. *Monthly Notices of the Royal Astronomical Society*, 503, 2021. <https://doi.org/10.1093%2Fmnras%2Fstab818>.
- [9] Battaglia P. The euclid wide survey. *Astronomy and Astrophysics*, 662, 2022.
- [10] Baumann D. Cosmology. 2022.
- [11] Bernardeau F., Colombi S., Gaztanaga E., Scoccimarro R. Large-scale structure of the universe and cosmological perturbation theory. *Physics Reports*, 367, 2002. <https://doi.org/10.1016%2Fs0370-1573%2802%2900135-7>.

- [12] Bird S., Viel M., Haehnelt M. G. Massive neutrinos and the non-linear matter power spectrum. *Monthly Notices of the Royal Astronomical Society*, 420, 2012. <https://doi.org/10.1111%2Fj.1365-2966.2011.20222.x>.
- [13] Boruah S. S., Rozo E., Fiedorowicz P. Map-based cosmology inference with lognormal cosmic shear maps, 2022. <https://arxiv.org/pdf/2204.13216.pdf>.
- [14] Boyle A., Uhlemann C., Friedrich O., Barthelemy A., Codis S., Bernardeau F., Giocoli C., Baldi M. Nuw CDM cosmology from the weak-lensing convergence PDF. *Monthly Notices of the Royal Astronomical Society*, 505, 2021. <https://doi.org/10.1093/mnras/stab1381>.
- [15] Bronstein I. N., Semendyaev K. A., Musiol G., Mühlig H. Taschenbuch der Mathematik. 2007.
- [16] Bruna J., Zaremba W., Szlam A., LeCun Y. Spectral networks and locally connected networks on graphs. *arXiv e-prints*, 2014. <https://arxiv.org/abs/1312.6203>.
- [17] Cahn R. N., Slepian Z., Hou J. A test for cosmological parity violation using the 3d distribution of galaxies, 2023. <https://ui.adsabs.harvard.edu/abs/2023PhRvL.130t1002C>.
- [18] Carlos G. U., Neil H. The netflix recommender system. *ACM Transactions on Management Information Systems*, 6, 2015.
- [19] Chollet F. Deep learning with python, second edition. <https://books.google.de/books?id=XHpKEAAQBAJ>.
- [20] Cimatti A., Frernali F., Nipoti C. Introduction to Galaxy Formation and Evolution: From Primordial Gas to Present-Day Galaxies. 2019. <https://ui.adsabs.harvard.edu/abs/2019igfe.book.....C>.
- [21] Cox R. T. Probability, Frequency and Reasonable Expectation. *American Journal of Physics*, 14, 1949. <https://ui.adsabs.harvard.edu/abs/1946AmJPh..14....1C>.
- [22] Croton D. J. Damn you, little h! (or, real-world applications of the hubble constant using observed and simulated data). *Publications of the Astronomical Society of Australia*, 30, 2013. <https://doi.org/10.1017%2Fpasa.2013.31>.
- [23] D'Amico G., Kleban M. Non-gaussianity after BICEP2. *Physical Review Letters*, 113, 2014. <https://doi.org/10.1103%2Fphysrevlett.113.081301>.
- [24] Davidson J., Liebald B., Liu J., Nandy P., Van Vleet T., Gargi U., Gupta S., He Y., Lambert M., Livingston B., Sampath D. The youtube video recommendation system. pages 293–296, 09 2010. <https://dl.acm.org/doi/10.1145/1864708.1864770>.

- [25] Dawson K. S., et al. THE BARYON OSCILLATION SPECTROSCOPIC SURVEY OF SDSS-III. *The Astronomical Journal*, 145, 2012. <https://doi.org/10.1088%2F0004-6256%2F145%2F1%2F10>.
- [26] Defferrard M., Bresson X., Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering, 2017. <https://arxiv.org/abs/1606.09375>.
- [27] Defferrard M., Milani M., Gusset F., Perrauidin N. Deepsphere: a graph-based spherical cnn, 2020. <https://arxiv.org/abs/2012.15000>.
- [28] DES Collaboration, et al. Dark Energy Survey Year 3 results: Cosmological constraints from galaxy clustering and weak lensing. *Physical Review D*, 105, 2022. <https://doi.org/10.1103%2Fphysrevd.105.023520>.
- [29] Dodelson S. Modern cosmology. 2003.
- [30] Duchi J., Hazan E., Singer Y. Adaptive subgradient methods for online learning and stochastic optimization. volume 12, 2010.
- [31] Ecke V. R., Cole S., Frenk C. S. Cluster evolution studies with the rosat-pspc: Scaling relations and the evolution of x-ray clusters. *Monthly Notices of the Royal Astronomical Society*, 282, 1996.
- [32] Eisenstein D. J., et al. Detection of the Baryon Acoustic Peak in the Large-Scale Correlation Function of SDSS Luminous Red Galaxies. *The Astrophysical Journal*, 633, 2005. <https://doi.org/10.1086%2F466512>.
- [33] Eisenstein D. J., Hu W. Baryonic features in the matter transfer function. *The Astrophysical Journal*, 496, 1998. <https://doi.org/10.1086%2F305424>.
- [34] Elemento O., Leslie C., Lundin J., Tourassi G. Artificial intelligence in cancer research, diagnosis and therapy. *Nature Reviews Cancer*, 21, 2021.
- [35] Feng S., Yan X., Sun H., Feng Y., Liu H. Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment. *Nature Communications*, 12, 2021.
- [36] Fluri J., Kacprzak T., Lucchi A., Refregier A., Amara A., Hofmann T., Schneider A. Cosmological constraints with deep learning from KiDS-450 weak lensing maps. *Physical Review D*, 100, 2019. <https://doi.org/10.1103%2Fphysrevd.100.063514>.
- [37] Fluri J., Kacprzak T., Refregier A., Amara A., Lucchi A., Hofmann T. Cosmological constraints from noisy convergence maps through deep learning. *Physical Review D*, 98, 2018. <https://doi.org/10.1103%2Fphysrevd.98.123518>.

- [38] Fluri J., Kacprzak T., Refregier A., Lucchi A., Hofmann T. Cosmological parameter estimation and inference using deep summaries. *Physical Review D*, 104, 2021. <https://doi.org/10.1103%2Fphysrevd.104.123526>.
- [39] Friedrich O. Statistical properties of the cosmic density field beyond 2-point statistics. 2018. <http://nbn-resolving.de/urn:nbn:de:bvb:19-234011>.
- [40] Friedrich O. The probability of the cosmos - classical, quantum, bayesian. https://www.usm.lmu.de/~oliverf/files/probability_of_the_cosmos.pdf, 2022.
- [41] Friedrich O., Gruen D., et al. Density split statistics: Joint model of counts and lensing in cells. *Physical Review D*, 98, 2018. <https://doi.org/10.1103%2Fphysrevd.98.023508>.
- [42] Friedrich O., Uhlemann C., Uhlemann, Villaescusa-Navarro F., Baldauf T., Manera M., Nishimichi T. Primordial non-Gaussianity without tails - how to measure f_{NL} with the bulk of the density PDF. *Monthly Notices of the Royal Astronomical Society*, 498, 2020. <https://doi.org/10.1093%2Fmnras%2Fstaa2160>.
- [43] Fryer, M. J. Lognormal Distributions: Theory and Applications. *Journal of the Royal Statistical Society Series C*, 38, 1989. <https://EconPapers.repec.org/RePEc:bla:jorssc:v:38:y:1989:i:2:p:381-382>.
- [44] Giesel E., Reischke R., Schäfer B. M., Chia D. Information geometry in cosmological inference problems. *Journal of Cosmology and Astroparticle Physics*, 2021, 2021. <https://doi.org/10.1088%2F1475-7516%2F2021%2F01%2F005>.
- [45] Gong Z., Halder A., Barreira A., Seitz S., Friedrich O. Cosmology from the integrated shear 3-point correlation function: simulated likelihood analyses with machine-learning emulators. *Journal of Cosmology and Astroparticle Physics*, 2023, 2023. <https://doi.org/10.1088%2F1475-7516%2F2023%2F07%2F040>.
- [46] Goodfellow I., Bengio Y., Courville A. Deep learning. 2016. <http://www.deeplearningbook.org>.
- [47] Gorski K. M., Hivton E., Bandy A. J., Wandelt B. D., Hansen F. K., Reinecke M., Bartelmann M. HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622, 2005. <https://doi.org/10.1086%2F427976>.
- [48] Gott III J. R., Juric M., Schlegel D., Hoyle F., Vogeley M., Tegmark M., Bahcall N., Brinkmann J. A map of the universe. *The Astrophysical Journal*, 624, 2005. <https://doi.org/10.1086%2F428890>.
- [49] Gruen D., Friedrich O., et al. Density split statistics: Cosmological constraints from counts and lensing in cells in DES y1 and SDSS data. *Physical Review D*, 98, 2018. <https://doi.org/10.1103%2Fphysrevd.98.023507>.

- [50] Géron A. Hands-on machine learning with scikit-learn, keras, and tensorflow: Concepts, tools, and techniques to build intelligent systems. 2019. <https://books.google.de/books?id=HHetDwAAQBAJ>.
- [51] Halder A., Barreira A. Response approach to the integrated shear 3-point correlation function: the impact of baryonic effects on small scales. *Monthly Notices of the Royal Astronomical Society*, 515, 2022. <https://doi.org/10.1017%2Fmnras.2013.31>.
- [52] Halder A., Friedrich O., Seitz S., Varga T. N. The integrated three-point correlation function of cosmic shear. *Monthly Notices of the Royal Astronomical Society*, 506, 2021. <https://doi.org/10.1093%2Fmnras%2Fstab1801>.
- [53] Halder A., Gong Z., Barreira A., Friedrich O., Seitz S., Gruen D. Beyond 3×2 -point cosmology: the integrated shear and galaxy 3-point correlation functions, 2023. <https://arxiv.org/abs/2305.17132>.
- [54] Hamana T., et al. Cosmological constraints from cosmic shear two-point correlation functions with HSC survey first-year data. *Publications of the Astronomical Society of Japan*, 72, 2020. <https://doi.org/10.1093%2Fpasj%2Fpsz138>.
- [55] Hamana T., Sakurai J., Koike M., Miller L. Cosmological constraints from subaru weak lensing cluster counts. *Publications of the Astronomical Society of Japan*, 67, 2015. <https://doi.org/10.1093%2Fpasj%2Fpsv034>.
- [56] Hammerman R., Russell A. L. Ada's legacy: Cultures of computing from the victorian to the digital age. 7, 2015.
- [57] Hammond D. K., Vandergheynst P., Gribonval R. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30, 2011. <https://www.sciencedirect.com/science/article/pii/S1063520310000552>.
- [58] Hartlap J., Simon P., Schneider P. Why your model parameter confidences might be too optimistic. unbiased estimation of the inverse covariance matrix. *Astronomy and Astrophysics*, 464, 2006. <https://doi.org/10.1051%2F0004-6361%3A20066170>.
- [59] Heavens A. Statistical techniques in cosmology. *arXiv e-prints*, 2010. <https://arxiv.org/abs/0906.0664>.
- [60] Hilbert S., Hartlap J., Schneider P. Cosmic shear covariance: the log-normal approximation. *Astronomy and Astrophysics*, 536, 2011. <https://doi.org/10.1051%2F0004-6361%2F201117294>.
- [61] Hinshaw G., et al. Nine-year wilkinson microwave anisotropy probe (wmap) observations: Cosmological parameter results. *The Astrophysical Journal Supplement Series*, 208, 2013. <https://dx.doi.org/10.1088/0067-0049/208/2/19>.

- [62] Hinton G. E., Osindero S., Teh Y. W. A fast learning algorithm for deep belief nets. *Neural Computation*, 18, 2006.
- [63] Hubble E. A Relation between Distance and Radial Velocity among Extra-Galactic Nebulae. *Proceedings of the National Academy of Science*, 15, 1929. <https://ui.adsabs.harvard.edu/abs/1929PNAS...15..168H>.
- [64] Ioffe S., Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv e-prints*, 2015. <https://arxiv.org/abs/1502.03167>.
- [65] Isserlis L. ON A FORMULA FOR THE PRODUCT-MOMENT COEFFICIENT OF ANY ORDER OF A NORMAL FREQUENCY DISTRIBUTION IN ANY NUMBER OF VARIABLES. *Biometrika*, 12, 1918. <https://doi.org/10.1093/biomet/12.1-2.134>.
- [66] Ivakhnenko A. G., Lapa V. G. Cybernetic predicting devices. 1973. <https://books.google.de/books?id=FhwVNQAAQAAJ>.
- [67] Jeffrey N., et al. Dark energy survey year 3 results: Curved-sky weak lensing mass map reconstruction. *Monthly Notices of the Royal Astronomical Society*, 505, 2021. <https://doi.org/10.1093/mnras/stab1495>.
- [68] Kacprzak T., Fluri J., Schneider A., Refregier A., Stadel J. Cosmogridv1: a simulated wcdm theory prediction for map-level cosmological inference. *Journal of Cosmology and Astroparticle Physics*, 2023, 2023. <https://doi.org/10.1088/1475-7516-2023-02050>.
- [69] Kaiser N. On the spatial correlations of Abell clusters. *The Astrophysical Journal Letters*, 284, 1984. <https://ui.adsabs.harvard.edu/abs/1984ApJ...284L...9K>.
- [70] Kaiser N., Squire G. Mapping the Dark Matter with Weak Gravitational Lensing. *The Astrophysical Journal*, 404, 1993. <https://ui.adsabs.harvard.edu/abs/1993ApJ...404..441K>.
- [71] Kilbinger M. Cosmology with cosmic shear observations: a review. *Reports on Progress in Physics*, 78, 2015. <https://doi.org/10.1088/0034-4885/78/2F086901>.
- [72] Kingma D. P., Ba J. Adam: A method for stochastic optimization, 2017. <https://arxiv.org/abs/1412.6980>.
- [73] Kobayashi M., Masukawa T. CP-Violation in the Renormalizable Theory of Weak Interaction. *Progress of Theoretical Physics*, 49, 1973. <https://academic.oup.com/ptp/article-pdf/49/2/652/5257692/49-2-652.pdf>.

- [74] KTeV Collaboration. Observation of direct cp violation in $k_{S,L} \rightarrow \pi\pi$ decays. *Physical Review Letters*, 83, 1999. <https://doi.org/10.1103/2Fphysrevlett.83.22>.
- [75] LeCun Y., Bengio Y., Hinton G. Deep learning. *Nature*, 521, 2015. <https://doi.org/10.1038/nature14539>.
- [76] Lesgourgues J. The cosmic linear anisotropy solving system (class) i: Overview. *arXiv e-prints*, 2011. <https://arxiv.org/pdf/1104.2932.pdf>.
- [77] Lewis A., Challinor A., Anthony L. Efficient computation of CMB anisotropies in closed FRW models. *The Astrophysical Journal*, 538, 2000. <https://arxiv.org/abs/astro-ph/9911177>.
- [78] Limber D. N. The Analysis of Counts of the Extragalactic Nebulae in Terms of a Fluctuating Density Field. *The Astrophysical Journal*, 117, 1953. <https://ui.adsabs.harvard.edu/abs/1953ApJ...117..134L>.
- [79] Lyons L. Bayes and Frequentism: a particle physicist's perspective. *Contemporary Physics*, 54, 2013. <https://doi.org/10.1080%2F00107514.2012.756312>.
- [80] Martin J., Ringeval C., Vennin V. Encyclopaedia inflationaris. *arXiv e-prints*, 2023. <https://arxiv.org/abs/1303.3787>.
- [81] Matilla J. M. Z., Sharma M., Hsu D., Haiman Z. Interpreting deep learning models for weak lensing. *Physical Review D*, 102, 2020. <https://doi.org/10.1103%2Fphysrevd.102.123506>.
- [82] Matsumura T., et al. Mission design of LiteBIRD. *Journal of Low Temperature Physics*, 176, 2014. <https://doi.org/10.1007%2Fs10909-013-0996-1>.
- [83] McCulloch W. S., Pitts W. A logical calculus of the ideas immanent in nervous activity. *Bulletin of mathematical biophysics*, 5, 1943.
- [84] Meschede D. Gerthsen physik. 2014.
- [85] Miles C., Bohrdt A., Wu R., Chiu C., Xu M., Ji G., Greiner M., Weinberger K. Q., Demler E., Kim E. A. Correlator convolutional neural networks as an interpretable architecture for image-like quantum matter data. *Nature Communications*, 12, 2021. <https://doi.org/10.1038%2Fs41467-021-23952-w>.
- [86] Minami Y., Komatsu E. New extraction of the cosmic birefringence from the planck 2018 polarization data. *Physical Review Letters*, 125, 2020. <https://link.aps.org/doi/10.1103/PhysRevLett.125.221301>.
- [87] Mitchell T. M. Machine learning. 1997. <https://books.google.de/books?id=EoYBngEACAAJ>.

- [88] Mukhanov V. Cmb-slow, or how to estimate cosmological parameters by hand. *International Journal of Theoretical Physics*, 43, 2004. <https://doi.org/10.1023%2Fb%3Aijtp.0000048168.90282.db>.
- [89] Mukhanov V. Physical foundations of cosmology. 2005.
- [90] Mukhanov V., Chibisov G. Quantum fluctuations and a nonsingular universe. *Soviet Journal of Experimental and Theoretical Physics Letters*, 33, 1981. <https://ui.adsabs.harvard.edu/abs/1981JETPL..33..532M>.
- [91] Padmanabhan T. Gravitation: Foundations and Frontiers. 2010. <https://ui.adsabs.harvard.edu/abs/2010grav.book.....P>.
- [92] Peacock J. A. Cosmological physics. 1999.
- [93] Peacock J. A., Dodds S. J. Reconstructing the linear power spectrum of cosmological mass fluctuations. *Monthly Notices of the Royal Astronomical Society*, 280, 1996.
- [94] Peebles P. J. E. Principles of physical cosmology. 1993.
- [95] Perlmutter S., et al. Measurements of Ω and Λ from 42 High-Redshift Supernovae. *The Astrophysical Journal*, 517, 1999. <https://doi.org/10.1086%2F307221>.
- [96] Perraудин N., Defferrard M., Kacprzak T., Sgier R. DeepSphere: Efficient spherical convolutional neural network with HEALPix sampling for cosmological applications. *Astronomy and Computing*, 27, 2019. <https://doi.org/10.1016%2Fj.ascom.2019.03.004>.
- [97] Planck Collaboration, et al. Planck 2018 results. VI. Cosmological parameters. *Astronomy and Astrophysics*, 641, 2020. <https://doi.org/10.1051%2F0004-6361%2F201833910>.
- [98] Planck Collaboration, et al. Planck 2018 results X. Constraints on inflation. *Astronomy and Astrophysics*, 641, 2020. <https://doi.org/10.1051%2F0004-6361%2F201833887>.
- [99] Porqueres N., Heavens A., Mortlock D., Lavaux G. Lifting weak lensing degeneracies with a field-based likelihood. *Monthly Notices of the Royal Astronomical Society*, 509, 2021. <https://arxiv.org/abs/2108.04825>.
- [100] Ramesh A., Dhariwal P., Nichol A., Chu C., Chen M. Hierarchical text-conditional image generation with clip latents, 2022. <https://arxiv.org/abs/2204.06125>.
- [101] Rifai S., Dauphin Y., Vincent P., Bengio Y., Muller X. The manifold tangent classifier. 2011. <https://api.semanticscholar.org/CorpusID:10210500>.
- [102] Rosenblatt F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, pages 65–386, 1958.

- [103] Samuel A. L. Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 44, 2000.
- [104] Sanchez A. G. The formation and evolution of cosmic structures. 2015. <https://www.mpe.mpg.de/~arielsan/>.
- [105] Sanchez A. G. Arguments against using $h^{-1}\text{Mpc}$ units in observational cosmology. *Physical Review D*, 102, 2020. <https://doi.org/10.1103%2Fphysrevd.102.123511>.
- [106] Schneider P., Ehlers J., Falco E. E. Gravitational Lenses. 1992. <https://ui.adsabs.harvard.edu/abs/1992grle.book.....S>.
- [107] Seitz C., Schneider P. Steps towards nonlinear cluster inversion through gravitational distortions. III. Including a redshift distribution of the sources. *Astronomy and Astrophysics*, 318, 1997. <https://ui.adsabs.harvard.edu/abs/1997A&A...318..687S>.
- [108] Shuman D. I., Narang S. K., Frossard P., Ortega A., Vandergheynst P. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30, 2013. <https://doi.org/10.1109%2Fmsp.2012.2235192>.
- [109] Smith L. N. A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay, 2018. <https://arxiv.org/abs/1803.09820>.
- [110] Smith M. J., Geach J. E. Astronomia ex machina: a history, primer, and outlook on neural networks in astronomy, 2022. <https://arxiv.org/abs/2211.03796>.
- [111] Springel V., White S. D. M., Tormen G., Kauffmann G. Populating a cluster of galaxies – I. Results at $z = 0$. *Monthly Notices of the Royal Astronomical Society*, 328, 2001. <https://doi.org/10.1046%2Fj.1365-8711.2001.04912.x>.
- [112] Spurio Mancini A., Piras D., Alsing J., Joachimi B., Hobson M. P. CosmoPower: emulating cosmological power spectra for accelerated bayesian inference from next-generation surveys. *Monthly Notices of the Royal Astronomical Society*, 511, 2022. <https://doi.org/10.1093%2Fmnras%2Fstac064>.
- [113] Stöcker H. Taschenbuch der Physik. 2010.
- [114] Takahashi R., Hamana T., Shirasaki M., Namikawa T., Nishimichi T., Osato K., Shiroyama K. Full-sky Gravitational Lensing Simulation for Large-area Galaxy Surveys and Cosmic Microwave Background Experiments. *The Astrophysical Journal*, 850, 2017. <https://doi.org/10.3847%2F1538-4357%2Faa943d>.

- [115] Takahashi R., Sato M., Nishimichi T., Taruya A., Oguri M. Revising the halofit model for the nonlinear matter power spectrum. *The Astrophysical Journal*, 761, 2012. <https://doi.org/10.1088%2F0004-637x%2F761%2F2%2F152>.
- [116] Taruya A., Takada M., Hamana T., Kayo I., Futamase T. Lognormal Property of Weak-Lensing Fields. *The Astrophysical Journal*, 571, 2002. <https://doi.org/10.1086%2F340048>.
- [117] Turing A. M. I.—COMPUTING MACHINERY AND INTELLIGENCE. *Mind*, LIX, 1950. <https://academic.oup.com/mind/article-pdf/LIX/236/433/30123314/lix-236-433.pdf>.
- [118] Umetsu K. Cluster–galaxy weak lensing. *The Astronomy and Astrophysics Review*, 28, 2020. <https://doi.org/10.1007%2Fs00159-020-00129-w>.
- [119] Verde L. Statistical methods in cosmology. 2010. https://doi.org/10.1007%2F978-3-642-10598-2_4.
- [120] Villaescusa Navarro F., et al. The CAMELS project: Cosmology and astrophysics with machine-learning simulations. *The Astrophysical Journal*, 915, 2021. <https://doi.org/10.3847%2F1538-4357%2Fabf7ba>.
- [121] Von Neumann J. First draft of a report on the edvac. *IEEE Annals of the History of Computing*, 15, 1993. <https://ieeexplore.ieee.org/abstract/document/238389>.
- [122] Weinberg S. *Cosmology*. OUP Oxford, 2008.
- [123] Wu Z., Pan S., Chen F., Long G., Zhang C., Yu P. S. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32, 2021. <https://doi.org/10.1109%2Ftnnls.2020.2978386>.
- [124] Xavier H. S, Abdalla F. B., Joachimi B. Improving lognormal models for cosmological fields. *Monthly Notices of the Royal Astronomical Society*, 459, 2016. <https://doi.org/10.1093%2Fmnras%2Fstw874>.
- [125] Zeljko I., Andrew C., Jacob V., Alexander G. Statistics, data mining, and machine learning in astronomy. *Princeton University Press*, 2013.
- [126] Zonca A., Singer L., Lenz D., Reinecke M., Rosset C., Hivton E., Gorski K. healpy: equal area pixelization and spherical harmonics transforms for data on the sphere in python. *Journal of Open Source Software*, 4, 2019. <https://doi.org/10.21105/joss.01298>.

Danksagung

Da meine Englischkenntnisse begrenzt sind, schreibe ich meine Danksagung auf Deutsch.

Zuerst möchte ich mich bei meinen Betreuern, Dr. Oliver Friedrich und Prof. Dr. Daniel Grün, herzlich bedanken. Sie haben mir die Möglichkeit gegeben, mich mit einer so faszinierenden und forschungsintensiven Masterarbeit zu beschäftigen. Dr. Friedrich hat mich nicht nur bei meiner Masterarbeit, sondern auch bei meinen zukünftigen wissenschaftlichen Karriereplänen stets unterstützt. Unsere Diskussionen und die Auseinandersetzung mit verschiedenen Interpretationen und Meinungen zu den Forschungsthemen haben mir auf hegelische Weise geholfen, die Logik meiner Masterarbeit weiter zu entwickeln und zu verfeinern. Ich war immer beeindruckt davon, wie schnell Sie Ideen für Tests entwickelt haben, um meine Argumente oder Ihre eigenen zu validieren. Sie mögen es als Erfahrung bezeichnen, aber ich glaube nicht daran. Die Hingabe, sich Tag für Tag mit Leib und Seele der Physik zu widmen, verleiht einem solche Fähigkeiten. Erfahrung allein ist eine zu billige Bezeichnung.

Auch Dr. Grün hat verschiedene Ideen für meine Masterarbeit eingebracht und mir die Gelegenheit gegeben, in der ACAI-Gruppe (Astrophysics, Cosmology, Artificial Intelligence Group) zu arbeiten. Die vielfältigen Diskussionen zu verschiedenen Themen in der Astronomie und Gravitationslinsenforschung innerhalb dieser Gruppe haben mir wirklich geholfen, einen breiteren Blickwinkel für meine Masterarbeit zu entwickeln.

Mein besonderer Dank gilt Anik Halder und Laurence Gong. Anik, es ist schwer, all das zusammenzufassen, was du für mich getan hast, in nur einer Seite. Du hast mir nicht nur beigebracht, wie man die Dinge lernt, sondern auch, wie man sie konzeptionell umsetzt. Am Anfang meiner Masterarbeit hatte ich keine Ahnung von Machine learning oder Python, da ich mich nie zuvor damit beschäftigt hatte. Doch du hast mir gezeigt, wie man solche Konzepte zur Programmierung erlernt. Wichtiger noch war jedoch deine mentale Unterstützung. In Momenten von Panik oder Depression aufgrund unerwarteter Ergebnisse oder nicht erfolgreicher Bewerbungen warst du immer da, um mir beizubringen, wie man solche Rückschläge sachlich akzeptiert und wie man weiterhin an meinen Zielen festhält. Die Wissenschaft ist herausfordernd, und man bekommt nicht immer das, was man erwartet. Doch deine Unterstützung war von unschätzbarem Wert. Laurence, wir haben über viele Themen diskutiert, nicht nur über Physik, sondern auch über Literatur, die Philosophie der Bayesschen Theorie und den Vergleich der chinesischen und japanischen Geschichte. Ich hoffe, dass wir unsere Kenntnisse in Bayesianischer Statistik auf Bereiche außerhalb der Kosmologie ausdehnen können, zum Beispiel auf die "vergleichende Lin-

guistik des Jiddischen". Leider enthält meine Masterarbeit viele Black-Box-Elemente, und das Verständnis dessen, was die Maschine gelernt hat, ist nicht immer einfach. Die Kombination Ihrer Erfahrung mit höherer Statistik und meiner Erfahrung mit feldbasierten Methoden könnte sich jedoch als äußerst fruchtbar erweisen. Dies muss insbesondere für die Beantwortung von Fragen zur Vertrauenswürdigkeit von Ergebnissen aus feldbasierten Methoden und den Gründen für Unterschiede bei der Schätzung kosmologischer Parameter zwischen dem späten und frühen Universum gelten.

Ein besonderer Dank gilt Parth Nayak, Dr. Michael Walther und Moritz Koch für Diskussionen über Verlustfunktionen und Tipps zur Memory Usage von GPU.

Ich würde noch Dank sagen für Diskussion über loss funktionen und tips memory usage für Parth Nayak, Dr. Michael Walther, Moritz Koch. Ich möchte meinen herzlichen Dank für die Unterstützung bei der Verbesserung meines Englisch an Nicola Foynes und Caitlin Gray aussprechen.

Ich bin auch sehr dankbar für die Mitglieder der ACAI-Gruppe und der LMU-Kosmologen, darunter Dr. Alexandre Barthelemy, Dr. Steffen Hagstotz, Dr. Tamas Norbert Varga, Dr. Ariel Sanchez, Dr. Luca Tortorelli, Prof. Dr. Mukhanov¹, Prof. Jochen Weller, Bhashin Thakore, Kai Lehman, David Gebauer, Marco Gibietz, Yun-Hsin Hsu, Jed Homer und Rohan Srikanth.

最後に向のち親に向けてです。べんがるの優し元で進の「ちゃんと験ごち親が読めるように場本語で固け」と元う験ドバ元すに卒してる利験ると思うので場本語で固きますなLMUの規則に則って元るか分かりませんか。に。まず験ド元ツはとても活白で験煙草のう金が高元し験元たり験のように美りし元食事がんしめる国では験りませんがなバ元向元とはも元ますかに験それでも素晴らし元場言です。何よりも験ド元ツに元る人すの頑固で面絡言さ元性やが験素晴らし元です。向は元す験お帰さまからのや機のせ元で験じ肉対な性やですか験これはじ肉では験りません。頑固で面絡言さ言験何事にも後りを固める性やが験素晴らし元験かでみっ言な者思(向が通って元る大学はみゅんへん大学です験みゅんへん工科大学では験りません。)を生み出して元るのだと思元ます。何よりも験このようにんし元卒行を固言機がが験ったこと験そして受わり者の向を受け入れて言れる者思に草謝して元ます。金ら頑にもド元ツでの生活をさポーとして言れて験本かに験りかとうご固元ます。何よりも験高校を卒業してそのままド元ツに行言と元うふ固けた決だを受け入れて言れて験験りかとう。普通の場本人の親なら反対するでしょう。

親父はよ言験小さ元頃に自然科学はギャン「るだ験小ちんことツじだと言って元ましたか験その通りだと思元ますなベ元ずの面絡言さ元哲学とは卒煙な言に。思うような済果が出な元と元ら元らし験何かがうま言元言と何よりも嬉し元です。このような経験は幾ら煙草を吸っても験お酒を飲んでも経験出来な元でしょう。これからもさポーとして言れるとの事験さ陸の謎のらーめんを送るとの事験草謝して元ますか験これからは自分で何とかした元です。自分の経済力でギャン「るするからこそ験面白元のです。

験まり連絡をせずに心配をかけ験本かにすみませんでした。元ずれ進路が決まり験お金がたまつたら験力しの頃験場本に帰ります。

¹Verzeihung, Sie sind Physiker nicht 'Kosmologe'.

Selbstständigkeitserklärung

Hiermit erkläre ich, die vorliegende Arbeit selbstständig verfasst zu haben und keine anderen als die in der Arbeit angegebenen Quellen und Hilfsmittel benutzt zu haben.

München, den 15. September 2023