Imperial College London
Department of Physics

Cosmology as an Optimisation Problem

T. Lucas Mäkinen

Supervised by Alan Heavens, Natalia Porqueres, & Benjamin D. Wandelt

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Astrophysics of Imperial College London, March 2025

Copyright Statement

Abstract

This thesis presents several studies under a common objective: accelerating, automating, and enhancing cosmological inference and Bayesian statistical data analysis using information-theoretic objectives and neural networks. I present introductions to modern cosmological theory, Bayesian probability, and statistics and unite them under implicit, simulation-based inference. The research chapters cover advances made in both cosmological and statistical methods, culminating in an application of "hybrid statistics" to measure the Dark Energy equation of state parameter from weak gravitational lensing fields captured by the Dark Energy Survey.

Acknowledgements

I would like to express an enormous amount of gratitude towards:

- My supervisors Alan Heavens, Natalia Porqueres, and Benjamin D. Wandelt, who have provided indispensable support and inspiration

- My scientific friends, colleagues, and collaborators: Tom Charnock, Justin Alsing, Niall Jeffrey, Matt Ho, Deaglan Bartlett, Shivam Pandey, Viraj Pandya, Greg Bryan, Stephon Alexander, Roberto Trotta, Josh Williamson, Guilhem Lavaux, Florent Leclerq, François Lanusse, Leander Thiele, Dave Clements, Hiranya Peris, Carolina Cuesta, Ce Sui, Kimeel Sooknunan, Ray Isichei, Alfie Robinson, Ethan Schrayer, Carina Norregard, Nadia Cooper, Morgan Williams, Xing Li, Romain Meriot, Keir Rogers, Wahid Rahman, Kosio Karchev, Jonathan Pritchard, Miles Cranmer, Daniel Mortlock, Chirag Modi, James Owen, Rafael Martinez-Galarza, Ben Boyd, Kaisey Mandel, Pablo Lemos, Nicolas Chartier, Will Handley, Chris Lovell, Justine Zeghal, Boris Leistedt, Eleni Tsaprazi, Axel Lapel, Simon Ding, Ludvig Dosser, Ronan Legin, Laurence Perrault-Levasseur, David Spergel, Kavilan Moodley, Devin Critchon, Ajith Sampath, Raul Jimenez, Francisco Villaescusa-Navarro, Wahidur Rahman, Lyman Page Jr, Shirley Ho, Peter Melchior, Neta Bahcall, Andy Goulding, Johnny Greco

Dedication

Mom, Isä, Grandma, Mummi, ja Ukki

CONTENTS

Statement on AI

At the time of writing, the world is in a state of rapid change. Calculations which would have made Alan Turing's head spin now take a fraction of a second. Artificial "Intelligence" (AI) is becoming a ubiquitous tool in everyday life. This technology has made some computational tasks considered impossible just a decade ago, not only possible, but accessible to the ordinary person. The implications for science are also extraordinary–with the right objective, AI technology can be leveraged to accelerate research in the right direction.

But this technology does not come for free. Massive language foundation models in industry cost billions of dollars to train on data which is often collected non-consensually and without compensation from billions of internet users. Legislation to prevent these often biased models from being used to exploit people is sluggish and does not yet exist at scale.

But it is who *controls* this technology that should be our greatest concern. We live in an era of unprecedented inequality, and the distribution of AI resources not only traces this gap between rich and poor, it also exacerbates it. Capital, in the form of user data and unpaid time spent on largely free apps, now flows out of the pocket of the average person into the hands of the tech billionaire who owns the technology. The billionaire is given tremendous influence over how people communicate, get their news, and, increasingly, vote in unstable democracies. And even without malicious interference, "over-smoothing" of information will continue to degrade the attention spans of users and the quality of data consumed needed to inspire and organise resistance to this exploitation of the many by the very few.

The irony of the Information Age and "AI" is that we are finding ourselves more misinformed and divided than ever. It is up to us to write down the objective for optimisation: Will it be science or content ? Resource extraction or environment ? People or profit ?

PREFACE

Science, in its most basic form, is an attempt to describe observations of the world around us. Observations, whether visual, computational, or aural, are all forms of *data*. Together, this makes science an attempt to describe the data-generating process.

In some situations, we as observers can collect more data to verify a description of the data-generating process. This could mean repeating an experiment, or going out into the field to more more observations of a system. Biology, social sciences, and particle physics are examples of fields that can benefit from this *frequentist* approach.

The science of physical cosmology attempts to describe the *entire universe*–how it formed, what it is comprised of, and how it will continue to evolve–at least in a statistical sense. But if we only have one universe, that is one observation, how can we possibly verify a theory with enough confidence, given that we only have one dataset ?

Unlike other branches of science or physics, cosmology requires a Bayesian approach to science, which relies on assessing probability as *the degree of belief* in an event, namely a distribution of possible measurements of a model parameter θ given some observation "data". A model parameter is a quantity that changes the behaviour and nature of the data that a model produces – it could be a number that controls the steepness of a line, or one that describes the nature of Dark Energy–and by extension the evolution of the Universe. This is is expressed formally via Bayes' Theorem,

$$p(\theta|\text{data}, M) = \frac{p(\text{data}|\theta, M)p(\theta)}{p(\text{data}), M}, \tag{1}$$

Phrased as a question, the posterior distribution on the lefthand side reads *How well do I know* θ

given that I have data and prior knowledge ? The answer to the question falls on the righthand side of the equation, whose parts can be broken down into a series of questions to be asked by the scientist:

1. $p(\text{data}|\theta, M)$; the likelihood: *How is the data generated from the model ?*
2. $p(\theta|M)$; the prior: *How much do I already know about θ under the model M ?*
3. $p(\text{data}|M)$; the evidence: *Which model does the data prefer ?*

Answering each of these questions can be difficult, especially when the data is hard to model (Q1) or it is unclear which of a collection of models to use (Q3). Advances in computing and formalism have made these questions easier to answer and evaluate, and are a key aspect of this thesis.

However, the most important high-level takeaway from Bayes' formalism is that phrasing statistics in this way requires that the data be *immutable*–the scientist must adjust her model to describe the data's idiosyncrasies.

The Bayesian approach also encodes a fundamental notion of uncertainty in how we understand reality. Parameters are always random variables–posteriors, no matter how tight, are still distributions. And no model is ever exactly "true"; the data can just express a preference for one over the other.

Bayesian statistics brokers the boundary between model and data, between theory and reality.

Niin silloin ve'en emonen, veen emonen, ilman impi,
nosti polvea merestä, lapaluuta lainehesta
sotkalle pesän sijaksi, asuinmaaksi armahaksi.

Tuo sotka, sorea lintu, liiteleikse, laateleikse.
Keksi polven veen emosen sinerväisellä selällä;
luuli heinämättähäksi, tuoreheksi turpeheksi.

Lentelevi, liitelevi, päähän polven laskeuvi.
Siihen laativi pesänsä, muni kultaiset munansa:
kuusi kultaista munoa, rautamunan seitsemännen.

Alkoi hautoa munia, päätä polven lämmitellä.
Hautoi päivän, hautoi toisen, hautoi kohta kolmannenki.

Jopa tuosta veen emonen, veen emonen, ilman impi,
tuntevi tulistuvaksi, hipiänsä hiiltyväksi;
luuli polvensa palavan, kaikki suonensa sulavan.

Vavahutti polveansa, järkytti jäseniänsä:
munat vierähti vetehen, meren aaltohon ajaikse;
karskahti munat muruiksi, katkieli kappaleiksi.

Ei munat mutahan joua, siepalehet veen sekahan.
Muuttuivat murut hyviksi, kappalehet kaunoisiksi:
munasen alainen puoli alaiseksi maaemäksi,
munasen yläinen puoli yläiseksi taivahaksi;
yläpuoli ruskeaista päivöseksi paistamahan,
yläpuoli valkeaista, se kuuksi kumottamahan;
mi munassa kirjavaista, ne tähiksi taivahalle,
mi munassa mustukaista, nepä ilman pilvilöiksi.

Ensimmäinen Runo, 177-244.

Creation of the world from the goldeneye's egg as described in the *Kalevala*.

CHAPTER 1

INTRODUCTION

Cosmology is the study of everything: the Universe, its birth, what it is made of, and, ultimately, its death. In some ways cosmology is one of the oldest sciences–most cultures have a creation story to explain how its people came to be. In the Finnish *Kalevala* mythology for instance, the Earth and heavens are formed when a goldeneye (sea duck) lays seven eggs that fall from her nest and shatter on the sea. Over the last four centuries science has superseded religion as a means to study the world around us. Cosmology, however, only matured as a truly quantitative science at the turn of the 20th century with the advent of Einstein's theory of general relativity. Although Einstein's field equations gestured toward an evolving Universe, the notion of an expanding (as opposed to a static) universe was not widely accepted until Hubble's observation of receding "nebulae" in the 1920s. Hereafter cosmology became a data-driven field; cosmological theory sharpened alongside telescope resolution, and the objective of the science became, much like the myths that preceded it, to explain how the Universe formed and predict how it will end. Current cosmological theory seeks to explain how a hot, dense *alkumuna*[1] Universe evolved into the complex, nonlinear structure of galaxies we see around us–structure that gave rise to the conditions for intelligent life to form.

As we collect more and more pristine data about our universe in the Information Age, we can cast cosmology as an optimisation problem: leveraging modern compute, more detailed statistics can be extracted from data, simulations sped up, and new models proposed to measure the exotic quantities like Dark Energy that govern our Universe.

[1]Finnish for primordial egg

1.1 Motivation and Objectives

This thesis attempts to cast modern cosmology as an optimisation problem to be accelerated using advances in computation and statistical formalism. To do so, we will need to unite cosmology, probability, and machine learning under a common objective.

To illustrate this concept we will focus on the formation of the large-scale structure of the Universe. We will begin in Chapter 2 with a treatment of gravitation in General Relativity, and show how the cosmological solution can be linked to nonlinear structure formation and weak lensing.

The probabilistic techniques introduced in Chapter 4 will be made as general as possible. We will take a Bayesian, information-theoretic approach to define statistics and neural network operators to be used in implicit inference, summarised for cosmology in Chapter 5.

In Chapter 6 we illustrate optimal and lossless information capture from mock cosmological fields using information maximising neural networks, indicating that implicit inference via compression can be made as information-exact as sampling the complete data distribution.

In Chapter 7 we explore large-scale structure as a graph and show that this sparse representation automatically combines existing cosmological statistics hidden in discrete catalogues. We explore the Fisher information metric as a function of data symmetries for principled physical feature analysis.

Chapter 8 utilises the Fisher information as a way to design an optimal, learned aggregation technique for sets and graphs. We show that this technique is information-optimal and scales automatically to datasets with much higher numbers of objects.

We next introduce "Hybrid Statistics" using two different approaches: Fisher and Mutual Information in three separate parts (Chapters 9, 10, and 11). The framework is built up and tested on multiple cosmological simulations in different regimes, and ultimately applied to Dark Energy Survey Year 3 data mocks in a simulation-based inference scheme, with the goal of improving weak lensing and Dark Energy parameter constraints over existing analyses.

Completed research chapters reflect submitted and published journal articles with minimal changes to their contents. The final two chapters combine ongoing and published work to facilitate a smoother narrative and introduction of concepts.

1.2 Contributions

Over the course of my PhD and postgraduate studies, I have:

1. Developed a novel foreground removal technique for 21cm cosmology Makinen et al. (2020)
2. Established that neural networks can be made information-optimal for hierarchical Bayesian simulation-based inference Makinen et al. (2021)
3. Developed improvements to the Information Maximising Neural Networks formalism
4. Introduced a rigorous treatment of graph-based methods in cosmology
5. Developed a novel information-optimal aggregation algorithm for graphs and sets
6. Introduced neural emulation to planetary formation inference (Rogers et al., 2023)
7. Contributed to neural compression codebase for LSST analysis (Lanzieri et al., 2024)
8. Contributed to a field-level analysis of weak lensing data with intrinsic alignments (Porqueres et al., 2023)
9. Helped develop `ltu-ili`, an open-source simulation-based inference software package (Ho et al., 2024).
10. Helped develop autoregressive, differentiable neural halo emulator (CHARM; Pandey et al. (2024)).
11. Established an automatic way to resolve data-driven degeneracies in simulation-based inference (ongoing; Makinen et al., in prep.a)
12. Contributed to simulation-based inference of semianalytic galaxy feedback models (ongoing Makinen et al., in prep.b)
13. Co-organised Simulation Based Inference for Galaxy Evolution conference (University of Bristol; 2024,2025)
14. Developed *hybrid statistics* across multiple formalisms to improve information extraction in generic neural compression

15. Applied hybrid statistics to the Dark Energy Survey Y3 data release to improve measurement of weak lensing parameters and Dark Energy equation of state

1.3 Statement of Originality

I assert that the work presented here is my own, and that external sources have been appropriately referenced.

1.4 Publications

1. **Makinen, T. Lucas**, Lachlan Lancaster, Francisco Villaescusa-Navarro, Peter Melchior, Shirley Ho, Laurence Perreault-Levasseur, and David N. Spergel. deep21: a deep learning method for 21 cm foreground removal. J. Cosmology Astropart. Phys., 2021(4):081, April 2021. (Makinen et al., 2020)

2. **Makinen, T. Lucas**, Tom Charnock, Justin Alsing, and Benjamin D. Wandelt. Lossless, scalable implicit likelihood inference for cosmological fields. J. Cosmology Astropart. Phys., 2021(11):049, November 2021.

3. Rafael Martinez-Galarza and **Makinen, T. Lucas**. Searching for time-domain anomalies in high energy catalogs. In David S. Adler, Robert L. Seaman, and Chris R. Benn, editors, Observatory Operations: Strategies, Processes, and Systems IX, volume 12186 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, page 121860J, August 2022. (Martínez-Galarza & Makinen, T. Lucas, 2022)

4. **Makinen, T. Lucas**, Tom Charnock, Pablo Lemos, Natalia Porqueres, Alan F. Heavens, and Benjamin D. Wandelt. The Cosmic Graph: Optimal Information Extraction from Large-Scale Structure using Catalogues. The Open Journal of Astrophysics, 5(1):18, December 2022. (Makinen et al., 2022)

5. Natalia Porqueres, Alan Heavens, Daniel Mortlock, Guilhem Lavaux, and **Makinen, T. Lucas**. Field-level inference of cosmic shear with intrinsic alignments and baryons. arXiv e-prints, page arXiv:2304.04785, April 2023. (Porqueres et al., 2023)

6. James G. Rogers, Claudia Jano Munoz, James E. Owen, and **Makinen, T. Lucas**. Exoplanet atmosphere evolution: emulation with neural networks. Monthly Notices of the Royal Astronomical Society, 519(4):6028–6043, March 2023. (Rogers et al., 2023)

7. **Makinen, T. Lucas**, Justin Alsing, and Benjamin D. Wandelt. Fishnets: Information-Optimal, Scalable Aggregation for Sets and Graphs. arXiv e-prints, page arXiv:2310.03812, October 2023. (Makinen et al., 2023)

8. Matthew Ho, Deaglan J. Bartlett, Nicolas Chartier, Carolina Cuesta-Lazaro, Simon Ding, Axel Lapel, Pablo Lemos, Christopher C. Lovell, **Makinen, T. Lucas**, Chirag Modi, Viraj Pandya, Shivam Pandey, Lucia A. Perez, Benjamin Wandelt, and Greg L. Bryan. LtU-ILI: An All-in-One Framework for Implicit Inference in Astrophysics and Cosmology. The Open Journal of Astrophysics, 7:54, July 2024. (Ho et al., 2024)

9. Denise Lanzieri, Justine Zeghal, **Makinen, T. Lucas**, Alexandre Boucaud, Jean-Luc Starck, and Francois Lanusse. Optimal Neural Summarisation for Full-Field Weak Lensing Cosmological Implicit Inference. arXiv e-prints, page arXiv:2407.10877, July 2024. (Lanzieri et al., 2024)

10. Shivam Pandey, Chirag Modi, Benjamin D. Wandelt, Deaglan J. Bartlett, Adrian E. Bayer, Greg L. Bryan, Matthew Ho, Guilhem Lavaux, **Makinen, T. Lucas**, and Francisco Villaescusa-Navarro. CHARM: Creating Halos with Auto-Regressive Multi-stage networks. arXiv e-prints, page arXiv:2409.09124, September 2024. (Pandey et al., 2024)

11. **Makinen, T. Lucas**, Ce Sui, Benjamin D. Wandelt, Natalia Porqueres, and Alan Heavens. Hybrid Summary Statistics. NeurIPS 2024, page arXiv:2410.07548, October 2024. (Makinen et al., 2024)

12. **Makinen, T. Lucas**, Alan Heavens, Natalia Porqueres, Tom Charnock, Axel Lapel, and Benjamin D. Wandelt. Hybrid summary statistics: neural weak lensing inference beyond the power spectrum. Journal of Cosmology and Astroparticle Physics, 2025(01):095, Jan 2025. (Makinen et al., 2025)

CHAPTER 2

MODERN COSMOLOGY

2.1 General Relativity

Gravity is the dominant force on the largest scales of the Universe, so a theory of gravitation is necessary for describing the physical origin of its structure. The best theory we have is Einstein's Theory of General Relativity (GR) and its cosmological solutions. Although GR does not extend to the smallest quantum scales, a lot of understanding can be garnered from known unknowns in the existing formalism. Dark matter and Dark Energy, which we now know make up most of our Universe, have not yet been measured in a laboratory setting; we do not know exactly how they behave at the particle level. But by looking at how they each affect the evolution of galaxies and large-scale structure in a statistical sense, we can *infer* their properties (parameters) from data in a Bayesian manner. The goal of this chapter is to demonstrate how GR can be simplified into a cosmological solution controlled by free parameters which can then be measured from data. The statistical advances discussed later will seek to unlock the maximal amount of information from this data by way of implicit or simulation-based inference.

At its core, GR gives us a description of gravity that is *geometric*; matter and energy are coupled to the spacetime fabric they sit on. Einstein's landmark theory broke from the restricted Newtonian description of gravity as a force, and in doing so paved the way for physicists to accurately predict anomalous effects like Mercury's orbit and the bending of light around the sun.

Einstein began with a thought experiment about a person in free-fall. If she reached into her pocket to take out her keys, she would find that they would float beside her, behaving as if gravity didn't exist. This is the *Weak Equivalence Principle*: in any field in spacetime we can choose a locally-inertial frame (like our free-faller's), in which the laws of motion are the same as if gravity were

absent. In geometric language, this is a "flat" point in spacetime.

Einstein extends this to the *Strong Equivalence Principle*: In a locally-inertial frame, all of special relativity (SR) applies. This means that the speed of light is the same for all observers in a vacuum, and that the laws of physics are identical in all inertial frames of reference. Taken together, this requires that the gravitational metric, or shape of spacetime dictates the effect of gravity on a particle, and the effect of the particle on spacetime.

We will proceed with an overview of motion in GR, largely following Heavens (2023) and Carroll (2019)'s treatments.

2.1.1 Geometry of Motion

Let's start with a particle with a trajectory[1] $\xi^\alpha = (ct, \mathbf{x})$, where we can optionally take $c = 1$. It is useful to parameterise this trajectory as a function of some proper time parameter τ. For a particle on a straight world line,

$$\frac{d^2\xi}{d\tau^2} = 0. \tag{2.1}$$

This is the equation of motion for a particle in a locally-inertial frame. In this frame, spacetime is governed by the flat Minkowski metric, $\eta_{\alpha\beta} = \text{diag}(1, -1, -1, -1)$, which results in a proper time interval or unit distance in this spacetime of $c^2\tau^2 = \eta_{\alpha\beta}dx^\alpha dx^\beta = c^2\tau^2 - d\mathbf{x}^2$. Now we consider another *arbitrary* coordinate system $x^\mu(\tau)$, which might be rotating or accelerating with respect to the original coordinates. A change in the old coordinates can be related to a change in the new coordinates via the chain rule

$$d\xi^\alpha = \frac{\partial\xi^\alpha}{\partial x^\mu}dx^\mu. \tag{2.2}$$

We can apply this change to the proper time interval as well,

$$c^2d\tau^2 = \eta_{\alpha\beta}d\xi^\alpha d\xi^\beta = \eta_{\alpha\beta}\left(\frac{\partial\xi^\alpha}{\partial x^\mu}dx^\mu\right)\left(\frac{\partial\xi^\beta}{\partial x^\nu}dx^\nu\right) = g_{\mu\nu}dx^\mu dx^\nu, \tag{2.3}$$

where $g_{\mu\nu}$ is the metric tensor in the new coordinate system:

$$g_{\mu\nu} \equiv \frac{\partial\xi^\alpha}{\partial x^\mu}\frac{\partial\xi^\beta}{\partial x^\nu}\eta_{\alpha\beta} \tag{2.4}$$

[1] Roman indices indicate spatial coordinates $1, \ldots, 3$ and Greek $0, \ldots, 3$.

If we apply this change of coordinates to our equation of motion, we arrive at

$$\frac{\partial \xi^\alpha}{\partial x^\mu}\frac{d^2x^\mu}{d\tau^2} + \frac{\partial^2 \xi^\alpha}{\partial x^\nu \partial x^\mu}\frac{dx^\mu}{d\tau}\frac{dx^\nu}{d\tau} = 0, \tag{2.5}$$

where it is useful to note that linear (Lorentz) transformations of our original coordinates sends the term $\frac{\partial^2 \xi^\alpha}{\partial x^\nu \partial x^\mu}$ to zero. We are interested in the acceleration of the particle in the new frame. If we multiply the above expression by $\frac{\partial x^\lambda}{\partial \xi^\alpha}$ and consolidate terms via the product rule, we arrive at the Geodesic Equation

$$\frac{d^2x^\mu}{d\tau^2} + \left(\frac{dx^\lambda}{d\xi^\alpha}\frac{\partial^2 \xi^\alpha}{\partial x^\nu \partial x^\mu}\right)\frac{dx^\mu}{d\tau}\frac{dx^\nu}{d\tau} = 0 \tag{2.6}$$

$$\frac{d^2x^\mu}{d\tau^2} + \Gamma^\lambda{}_{\mu\nu}\frac{dx^\mu}{d\tau}\frac{dx^\nu}{d\tau} = 0. \tag{2.7}$$

The affine connection (Christoffel symbol), $\Gamma^\lambda{}_{\mu\nu}$, here describes the "accelerations" of particles, which can be seen in the case of massless particles. The interval $d\tau$ here is zero, so we can utilise $\sigma = \xi^0$ as our trajectory parameter. Applying this choice to Eq. 2.1 yields

$$\frac{d^2x^\mu}{d\sigma^2} + \Gamma^\lambda{}_{\mu\nu}\frac{dx^\mu}{d\sigma}\frac{dx^\nu}{d\sigma} = 0 \tag{2.8}$$

Here, if $\Gamma^\lambda{}_{\mu\nu}$ is not zero, the particle undergoes acceleration independence of its mass (or lack thereof). One can induce gravity, or the effects of it, by choosing a coordinate system with a non-zero affine connection. We commonly encounter nonzero Γ in scenarios like centrifugal forces on carnival rides.

2.1.2 Particle motion in a metric

But what about massive particles ? The exact form of the metric $g_{\mu\nu}$ comes from a solution of Einstein's field equations, usually with symmetries or assumptions about the system imposed. Before proceeding, it is important to note that given a metric $g_{\mu\nu}$, the affine connection can be written as

$$\Gamma^\sigma{}_{\lambda\mu} = \frac{1}{2}g^{\nu\sigma}\left(\frac{\partial g_{\mu\nu}}{\partial x^\lambda} + \frac{\partial g_{\lambda\nu}}{\partial x^\mu} - \frac{\partial g_{\mu\lambda}}{\partial x^\nu}\right), \tag{2.9}$$

where we introduce the inverse metric $g^{\mu\sigma}g_{\sigma\nu} = \delta^\mu_\nu$. The important takeaway here is that the term that "induces" gravity depends on the gradients of the metric $g_{\mu\nu}$, much like the Newtonian analogue $\mathbf{g} = -\nabla\Phi$. To calculate the motion of a particle in an arbitrary coordinate system we now require

two more steps: a prescription from Einstein's equations for $g_{\mu\nu}$ and a convenient way to compute the affine connections.

2.1.3 Variational Formulation

We can make use of variational calculus to ease our downstream dynamics calculations once we have a valid metric. We can compute the proper time a particle undergoes along its worldline between two points A and B, parameterised by some monotonically increasing parameter p along the path:

$$c\tau_{AB} = c\int_A^B d\tau = c\int_A^B \frac{d\tau}{dp}dp = \int_A^B \sqrt{g_{\mu\nu}\frac{dx^\mu}{dp}\frac{dx^\nu}{dp}}dp = \int_A^B L(x^\mu, \dot{x}^\mu)dp, \tag{2.10}$$

where $\dot{x}^\mu = \frac{dx^\mu}{dp}$ and $L(x^\mu, \dot{x}^\mu)$ becomes our Lagrangian which satisfies the Euler-Lagrange Equations (Riley et al., 2006):

$$\frac{dL}{dx^\mu} - \frac{d}{dp}\left(\frac{\partial L}{\partial \dot{x}^\mu}\right) = 0. \tag{2.11}$$

By assuming an affine p which increases linearly with τ, squaring L makes downstream calculations with the square root of the metric more convenient:

$$\frac{dL^2}{dx^\mu} - \frac{d}{dp}\left(\frac{\partial L^2}{\partial \dot{x}^\mu}\right) = 0. \tag{2.12}$$

2.2 Einstein's Equations

Einstein was motivated to find conservation laws, central to Newtonian and special-relativistic physics, for his geometry of motion. In GR this boils down to finding a relation between matter and the metric which reduces to Poisson's equation in Newtonian gravity

$$\nabla^2\Phi = 4\pi G\epsilon, \tag{2.13}$$

The matter distribution is captured by the energy-momentum tensor: for a perfect fluid with energy density ϵ, pressure p the tensor is

$$T_{\alpha\beta} = (\epsilon c^2 + p)u_\alpha u_\beta - pg_{\alpha\beta}, \tag{2.14}$$

where the vector u^α is the fluid's four-velocity

$$u_\alpha = g_{\alpha\beta}u^\beta = g_{\alpha\beta}\frac{dx^\beta}{ds}, \tag{2.15}$$

where x^β is the trajectory of a fluid element in spacetime. The pressure is related to the density via an *equation of state parameter*, most often modelled as

$$p = w \cdot \epsilon \tag{2.16}$$

Derivatives of the energy-momentum tensor do not behave as tensors, so a conservation law must take a different form. Einstein started with the Riemann-Christoffel tensor, used to study curvature of hyperspaces:

$$R^\alpha{}_{\beta\mu\nu} = \frac{\partial\Gamma^\alpha{}_{\beta\nu}}{\partial x^\mu} - \frac{\partial\Gamma^\alpha{}_{\beta\mu}}{\partial x^\nu} + \Gamma^\alpha{}_{\gamma\mu}\Gamma^\gamma{}_{\beta\nu} - \Gamma^\alpha{}_{\gamma\nu}\Gamma^\gamma{}_{\beta\mu}, \tag{2.17}$$

which does transform as a tensor. The Ricci tensor can be formed via

$$R_{\alpha\beta} = R^\mu_{\alpha\mu\beta}, \tag{2.18}$$

and finally to measure scalar curvature the Ricci scalar can be computed by contracting over the metric

$$R = g^{\alpha\beta}R_{\alpha\beta} \tag{2.19}$$

Einstein proposed the Einstein tensor

$$G_{\alpha\beta} = R_{\alpha\beta} - \frac{1}{2}g_{\alpha\beta}R, \tag{2.20}$$

for which $G_\alpha{}^\beta{}_{;\beta} = 0$, and contains the required second derivatives of the metric (Coles & Lucchin, 2002). This motivated *Einstein's Equations* to be written as

$$R_{\alpha\beta} - \frac{1}{2}g_{\alpha\beta}R = \frac{8\pi G}{c^4}T_{\alpha\beta} + \Lambda g_{\alpha\beta}, \tag{2.21}$$

which reduces to the Poisson's equation (2.13) in the limit of a weak gravitational field. Einstein introduced the Λ cosmological constant term to ensure a static cosmological solution, which was the commonly-held view before Hubble's discoveries of an expanding universe in the 1920s. Altogether, this expression relates the universe's spatial curvature (left-hand side) to its energy content (right-

hand side).

2.3 The Cosmological Principle & Friedmann solution

Modern cosmology is based on the *Cosmological Principle*. The idea is that by "zooming out" on a coarse enough grid,

1. *The universe is homogeneous*: There is no preferred place. If we picture the universe to be filled with a fluid, we don't expect it to be "clumpy" in any specific region.

2. *The universe is isotropic*: There is no preferred direction. A fluid filling the universe exerts an equal pressure, p, in all spatial directions.

This principle only holds for sufficiently large scales, generally on the order hundreds or thousands of megaparsecs (1 Mpc $= 3.0857 \times 10^{19}$ kilometres). We can express this principle as a maximally-symmetric hypersurface growing growing according to a *scale factor*, $a(t) > 0$, yielding:

$$ds^2 = dt^2 - a^2(t)[dx^2 + dy^2 + dz^2] \tag{2.22}$$

The constraints imposed by homogeneity and isotropy allow only three possible geometries: positively curved, negatively curved, and flat. We can capture these criteria by reparameterizing $g_{\mu\nu}$ into polar coordinates:

$$ds^2 = dt^2 - a^2(t)\Big[\frac{dr^2}{1 - kr^2} + r^2 d\Omega^2\Big], \tag{2.23}$$

where k is the curvature, and $d\Omega^2 = d\theta^2 + \sin^2\theta d\phi^2$. This is the Friedmann-Lemaître-Robertson-Walker (FLRW) metric, describing distances in expanding spacetime. Here, ds depends both on the expansion of the scale factor as well as the given universe's curvature. This expression can be transformed further into co-moving coordinates, with the variable change

$$d\chi^2 = \frac{dr^2}{1 - kr^2} \tag{2.24}$$

and

$$f_K(\chi) = \begin{cases} \chi, & K = 0 \quad \text{(Flat)} \\ K^{-1/2}\sin(K^{-1/2}\chi), & K > 0 \quad \text{(Positive Curvature)} \\ (-K)^{-1/2}\sinh((-K)^{-1/2}\chi), & K < 0 \text{ (Negative Curvature)} \end{cases} \tag{2.25}$$

yielding the co-moving FLRW metric:

$$ds^2 = dt^2 - a^2(t)\left[d\chi^2 + f_k^2(\chi)d\Omega^2\right] \tag{2.26}$$

The scale factor's behaviour is controlled by Einstein's equations with the cosmological principle imposed.

2.3.1 Cosmological Redshift

As the universe expands with a, photons' wavelengths are stretched, leading to a *cosmological redshift*. Photons travel along geodesics–by symmetry in Eq 2.26, we see that radial trajectories ($\theta, \phi = \text{const}$) satisfies this requirement. The trajectory can be written in terms of conformal time $\chi(\eta)$ and is determined by the geodesic condition $ds^2 = 0$, or

$$d\eta^2 - d\chi^2 = 0, \tag{2.27}$$

from which we obtain

$$\chi(\eta) = \pm\eta + \text{const}, \tag{2.28}$$

implying that radial light geodesics correspond to straight lines (light cone) at 45° in the η, χ plane. From here, consider a source at comoving coordinate χ_{em} that emits a signal of short (conformal) duration $\Delta\eta$. The trajectory of the signal is then

$$\chi(\eta) = \chi_{\text{em}} - (\eta - \eta_{\text{em}}), \tag{2.29}$$

which reaches the observer at $\chi_{\text{obs}} = \eta_{\text{em}} + \chi_{\text{em}}$. While the conformal duration of the signal is unchanged at the source and measurement, the *physical* time interval of the source are different at

points of emission and observation. They read

$$\Delta t_{\rm em} = a(\eta_{\rm em})\Delta\eta, \qquad \Delta t_{\rm obs} = a(\eta_{\rm obs})\Delta\eta, \tag{2.30}$$

If the period of the light wave is Δt, the wavelength of the light is $\lambda_{\rm em} = \Delta t_{\rm em}$ at emission and $\lambda_{\rm obs} = \Delta t_{\rm obs}$ at observation. Eliminating $\Delta\eta$, we obtain:

$$\frac{\lambda_{\rm obs}}{\lambda_{\rm em}} = \frac{a(\eta_{\rm obs})}{a(\eta_{\rm em})}. \tag{2.31}$$

Assuming measurement wavelength $\lambda_{\rm obs}$ is recorded for today's scale factor $a_0 = 1$, we can introduce the *redshift*

$$z = \frac{\lambda_{\rm obs} - \lambda_{\rm em}}{\lambda_{\rm em}}, \tag{2.32}$$

which can be related to the scale factor as

$$1 + z = \frac{a_0}{a} = \frac{1}{a}. \tag{2.33}$$

Measurement of the redshift of a source requires knowing the wavelength of the light at emission, which usually requires knowledge of the physical spectrum of the source to measure the shift. Measuring the cosmological redshift of sources is a direct probe of cosmological expansion and structure formation.

2.3.2 The Friedmann Equations

Modern cosmic dynamics comes from solutions to Einstein's field equations (Eq. 2.21). Inserting the metric $g_{\mu\nu}$ and the homogeneous and isotropic energy-momentum tensor into Einstein's field equations yields two dynamic solutions to Einstein's equations, first published by Alexander Friedmann in 1922 (Friedmann, 1922). The first Friedmann equation describes the expansion rate of the universe, and comes from the time component of Equation 2.21:

$$H^2(t) = \left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G}{3}\epsilon - \frac{k}{a^2} \tag{2.34}$$

This equation describes how the expansion rate, $H(t)$, depends on the homogeneous density, ϵ. The second term on the righthand side describes the role curvature plays in the expansion. The spatial

components of Einstein's Field Equations give us the second equation:

$$\frac{\ddot{a}}{a} = \frac{-4\pi G}{3}(\epsilon + 3p). \tag{2.35}$$

2.3.3 The ΛCDM Standard Model and Cosmological Parameters

Our universe is comprised of four of these "fluids": pressure-less matter, relativistic particles, a cosmological constant (Dark Energy), and curvature. This multi-component model of our universe is often dubbed the ΛCDM Standard Model.

We can investigate how each fluid's density, ϵ, evolves over time by introducing a continuity equation, derived from the first two Friedmann Equations (Mukhanov, 2005):

$$\frac{d\epsilon}{dt} + 3\epsilon\frac{\dot{a}}{a}(1 + w) = 0 \tag{2.36}$$

Solving this first-order differential equation gives us an expression for ϵ as a function of a and w:

$$\epsilon \propto \exp\left[3\int_{a_o}^{a} \frac{1 + w(a')}{a'} da'\right] \tag{2.37}$$

Where $a_o = 1$ is the scale factor measured today. For fluids with constant w, the relation simplifies to

$$\epsilon \propto \left(\frac{a_o}{a}\right)^{3(1+w)} \tag{2.38}$$

The densities of each of the components in our universe evolve as functions of the scale factor, summarized in Table 2.1. Notice that relativistic particles evolve with an additional factor of a^{-1} on account of redshift (Mukhanov, 2005). The evolution of curvature for a non-flat universe can also be analyzed as an energy density that behaves proportionally to a^{-2} on account of the curvature term in the Friedmann Equation.

Since it is difficult in cosmology to measure absolute energy densities, is useful to rewrite the first Friedmann Equation in terms of the dimensionless, *density parameters* of each of our universe's components. To do this, we divide the absolute densities by a critical density term, ϵ_{crit}, obtained from the first Friedmann Equation. For a spatially flat universe, the critical density as a function of

Fluid	Equation of State	Density	Fraction
matter	$w = 0$	$\epsilon_m \propto a^{-3}$	Ω_m
relativistic matter	$w = 1/3$	$\epsilon_{\rm rel} \propto a^{-4}$	$\Omega_{\rm rel}$
cosmological constant	$w = -1$	$\epsilon_\Lambda \propto a^0$	Ω_Λ
curvature	$w = -1/3$	$\epsilon_k \propto a^2$	Ω_K

Table 2.1: Sample cosmological fluids and how their densities evolve with respect to the scale factor, a.

the current scale factor, H_0 is

$$\epsilon_{crit} = \frac{3H_0^2}{8\pi G} \tag{2.39}$$

For each i fluid component of our universe, we can define a density parameter, $\Omega_{i,0}$:

$$\Omega_{i,0} = \frac{\epsilon_{i,0}}{\epsilon_{crit}} \tag{2.40}$$

Where $\epsilon_{i,0}$ and $\Omega_{i,0}$ is the given component's absolute density and density parameter, respectively, as measured today. The sum of all density parameters defines the *curvature* density parameter,

$$1 - \Omega_K = \Omega_m + \Omega_r + \Omega_{\rm de}, \tag{2.41}$$

where $\Omega_K = -(c/H_0)^2 K$. We divide the first Friedmann Equation (2.34) by the critical density to obtain

$$\frac{H^2(t)}{H_0} = \frac{\epsilon(t)}{\epsilon_{crit}} + \frac{\Omega_K}{a^2(t)} \tag{2.42}$$

Unpacking the $\frac{\epsilon(t)}{\epsilon_{crit}}$ term for the other fluids, we obtain the full Friedmann Equation with respect to our universe's density parameters:

$$H(t) = H_0 \left[\frac{\Omega_{m,0}}{a^3} + \Omega_{\rm de,0} + \frac{\Omega_K}{a^2} + \frac{\Omega_{\rm rad,0}}{a^4} \right]^{1/2}. \tag{2.43}$$

These quantities are usually free parameters that control the way the FLRW universe evolves over its lifetime, and can be linked to astrophysical observables. The density parameter of non-relativistic matter consists of cold dark matter (CDM), baryonic matter, and (possibly) heavy and non-relativistic neutrinos $\Omega_m = \Omega_c + \Omega_b + \Omega_\nu$. Relativistic matter consists of photons from the cosmic microwave background (CMB) and light neutrinos, represented by $\Omega_{\rm rad}$. The component driving the relatively recent accelerated expansion of the universe is the dark energy fraction, $\Omega_{\rm de}$. Previously accounted for in Einstein's equation as just a constant, dark energy can now be more loosely defined

as any substance that drives expansion with $w < -1/3$. Lacking a well-motivated physical model, the equation of state is often parameterised by the first few coefficients of a Taylor expansion,

$$w(a) = w_0 + w_a(1 - a), \tag{2.44}$$

where $w_a > 0$ corresponds to evolving dark energy as a function of the scale factor (Chevallier & Polarski, 2001; Linder & Jenkins, 2003).

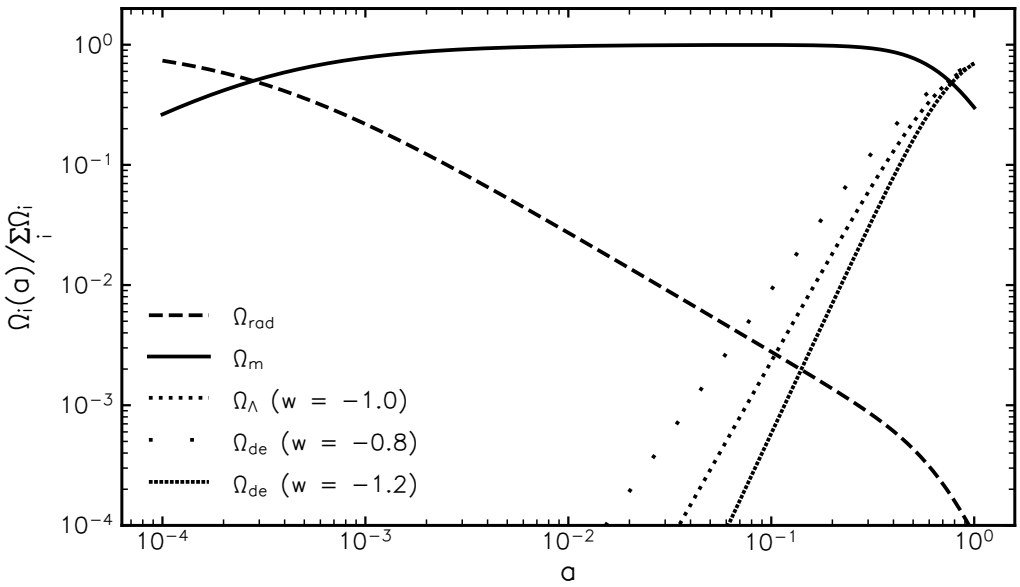


Figure 2.1: Evolution of the normalized density parameters, Ω_i, as a function of the scale factor a, assuming a spatially flat universe ($k = 0$). The phase portraits of the Dark Energy density parameter, Ω_{de}, are shown for several values of w. At the present day ($z_0 = 0$, $a = 1$), we live in a dark energy-dominated universe.

We now have an expression that can be readily plotted to show the evolution of each density parameter over a cosmological timescale. Figure 2.1 shows the evolution of density parameters for a spatially flat universe as a function of the scale factor. The phase portraits show that only very recently ($a \approx 0.8$) did our universe become Dark Energy-dominated.

2.4 Weak Gravitational Lensing

We have already used GR to describe how a homogeneous and isotropic universe behaves, characterising its energy components as fluids with specified equations of state. A few years before Friedmann published his solution, the first triumph of Einstein's theory came with the prediction of the deflection of light around the sun's gravitational potential. A Newtonian calculation for the deflection angle

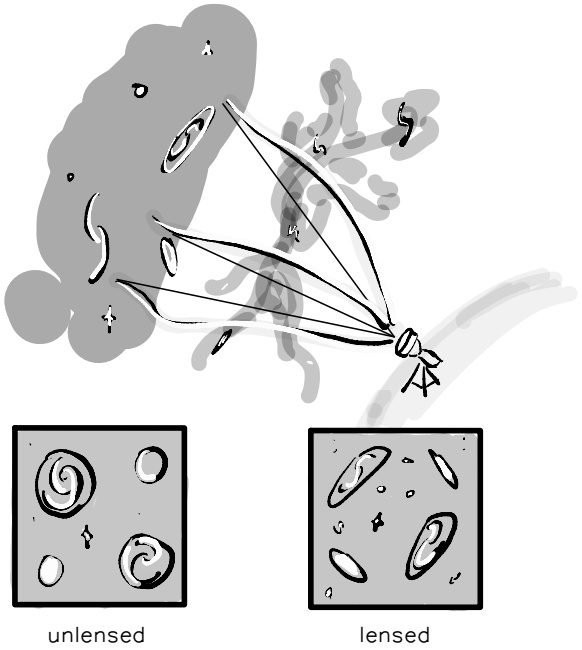


Figure 2.2: Weak gravitational lensing distorts the shapes of distant galaxies as light passes through large-scale structure.

fails to reconcile both spatial and temporal curvature, differing by a factor of two. A measurement of this gravitational *lensing* of stars behind the sun during a solar eclipse on May 29, 1919 (Dyson et al., 1920) confirmed Einstein's prediction of a 1.7 arcsecond deflection (Einstein, 1916).

Weak gravitational lensing looks at much more subtle perturbations of light in a *statistical* sense. Light from distant galaxies streams past galaxies and massive dark matter structures on its way to our telescopes. Altering the trajectory of these photons distorts the apparent shapes of galaxies as we see them. By recording many such shape distortions over the sky, we can link cosmic *shear* as a probe to the clumpiness of the universe, which is linked to the Friedmann quantities or cosmological parameters introduced in Section 2.3.2. While obtaining images of sufficient quality to measure shape distortions in galaxies is difficult in practice, weak lensing directly probes the matter structure of the universe–even the dark matter that we cannot see.

2.4.1 Distorting a bundle of light

We can think of an image of a distant galaxy as a bundle of light streaming through a clumpy universe towards an observer. The path the light takes will be distorted slightly as it passes through

gravitational potentials Ψ and Φ:

$$ds^2 = \left(1 + \frac{2\Phi}{c^2}\right) c^2 dt^2 - \left(1 - \frac{2\Psi}{c^2}\right) R^2(t) \left(dr^2 + S_k^2(r) d\beta^2\right), \tag{2.45}$$

where $d\beta = \sqrt{d\theta^2 + \sin^2\theta d\phi^2}$ and the deviations from the metric are small. For what follows it is more convenient to rewrite this interval with respect to the conformal time $d\eta = c\frac{dt}{R(t)}$, which adjusts for the growth captured by the scale factor $R(t)$. Within our spherical coordinate system we can concentrate our efforts on a small patch of sky pierced by our light bundle. We can assign the coordinates $\theta_x = \theta\cos\phi$ and $\theta_y = \theta\sin\phi$ to the edges of this patch and rewrite our distorted metric as

$$ds^2 = R^2(t) \left[\left(1 + \frac{2\Phi}{c^2}\right) d\eta^2 - \left(1 - \frac{2\Psi}{c^2}\right) \left[dr^2 + S_k^2(r)(d\theta_x^2 + d\theta_y^2)\right]\right], \tag{2.46}$$

which now tracks the changes in the angles of the light ray and image it traces on the patch of sky as a function of the gravitational potentials. In this treatment we consider the incoming radial path to be unperturbed such that $0 = ds^2 \approx d\eta^2 - dr^2$, giving us $\frac{dr}{d\eta} = -1$. Using the variational approach (Eq 2.12), we first turn to the conformal time component, $x^\mu = \eta$, which we require to zero-order in Φ for a spacelike interval:

$$\frac{d}{dp}\left(R^2\dot{\eta}\right) = 0, \tag{2.47}$$

which we can use (with choice of units in p) to give us the relation:

$$\frac{d\eta}{dp} = \frac{1}{R^2}. \tag{2.48}$$

Since we're tracking the motion of photons, we can set $L^2 = \left(\frac{ds}{dp}\right)^2 = 0$ and write, for $x^\mu = \theta_x$:

$$R^2 \frac{2}{c^2}\frac{\partial\Phi}{\partial\theta_x}\dot{\eta}^2 + \frac{2}{c^2}R^2\frac{\partial\Psi}{\partial\theta_x}\left(\dot{r}^2 + r^2\dot{\theta}_x^2 + r^2\dot{\theta}_y^2\right) - \frac{d}{dp}\left[-2R^2r^2\left(1 - \frac{2\Psi}{c^2}\right)\dot{\theta}_x\right] = 0 \tag{2.49}$$

The middle term $\left(\dot{r}^2 + r^2\dot{\theta}_x^2 + r^2\dot{\theta}_y^2\right)$ reduces to $\dot{\eta}^2$ to zero-order, and the rightmost term simplifies via change of variables $\frac{d}{dp} = \frac{1}{R^2}\frac{d}{d\eta}$ and $R^2\dot{\theta}_x = \frac{d\theta_x}{d\eta}$ to

$$\frac{d}{dp}\left[-2R^2r^2\left(1 - \frac{2\Psi}{c^2}\right)\dot{\theta}_x\right] = \frac{2}{R^2}\left[\frac{d^2\theta_x}{d\eta^2} + 2r\frac{dr}{d\eta}\frac{d\theta_x}{d\eta}\right] \tag{2.50}$$

where the perturbation with respect to Ψ falls away when multiplied with $\dot{\theta}$. Collecting the simplified terms, we arrive at

$$\frac{d^2\theta_x}{d\eta^2} - \frac{2}{r}\frac{d\theta_x}{d\eta} = -\frac{1}{c^2 r^2}\left(\frac{\partial\Phi}{\partial\theta_x} + \frac{\partial\Psi}{\partial\theta_x}\right) \tag{2.51}$$

It is now useful to rewrite the deflection of the light bundle in terms of the comoving displacement

$$x_i = r\theta_i, \qquad i = x, y \tag{2.52}$$

With a similar expression for $x^\mu = \theta_y$, Eq. 2.51 can be written in vector form via

$$\frac{d^2\mathbf{x}}{d\eta^2} = -\frac{1}{c^2}\nabla(\Phi + \Psi) = -\frac{2}{c^2}\nabla\Phi \tag{2.53}$$

where we write $\nabla = (\partial_x, \partial_y)$ in comoving coordinates. This gradient is understood to be *perpendicular* to the incoming light ray. We can now link changes in the path light takes from distant galaxies to the gravitational potential Φ, which is related to the matter overdensity field.

2.4.2 The Lensing Potential and Cosmic Shear

Eq. 2.53 can be twice integrated with respect to the coordinate $\mathbf{x}(r)$

$$x_i(r) = r\theta_i - \frac{1}{c^2}\int_0^r dr' \frac{\partial(\Phi + \Psi)}{\partial x_i'}(r - r'), \tag{2.54}$$

where the incoming light ray arrives at an angle θ_i. We are interested in the distortion of an image, comprised of two or more rays of light defining a separation vector $\Delta\mathbf{x}$. We can find an expression for this distortion by performing a Taylor expansion of the (combined) potential's gradient $\partial\Phi/\partial x_i'$. This is equivalent to evaluating the integral along an unperturbed radial line in the *Born approximation* which is valid to high accuracy (Kilbinger, 2015; Bartelmann & Schneider, 2001):

$$\Delta x_i = r\Delta\theta_i - \frac{2}{c^2}\Delta\theta_j \int_0^r dr' \frac{(r - r')}{rr'}\frac{\partial^2\Phi(\mathbf{r}')}{\partial\theta_i\partial\theta_j}, \tag{2.55}$$

or more concisely as

$$\Delta x_i = r\Delta\theta_j(\delta_{ij} - \phi_{ij}) \tag{2.56}$$

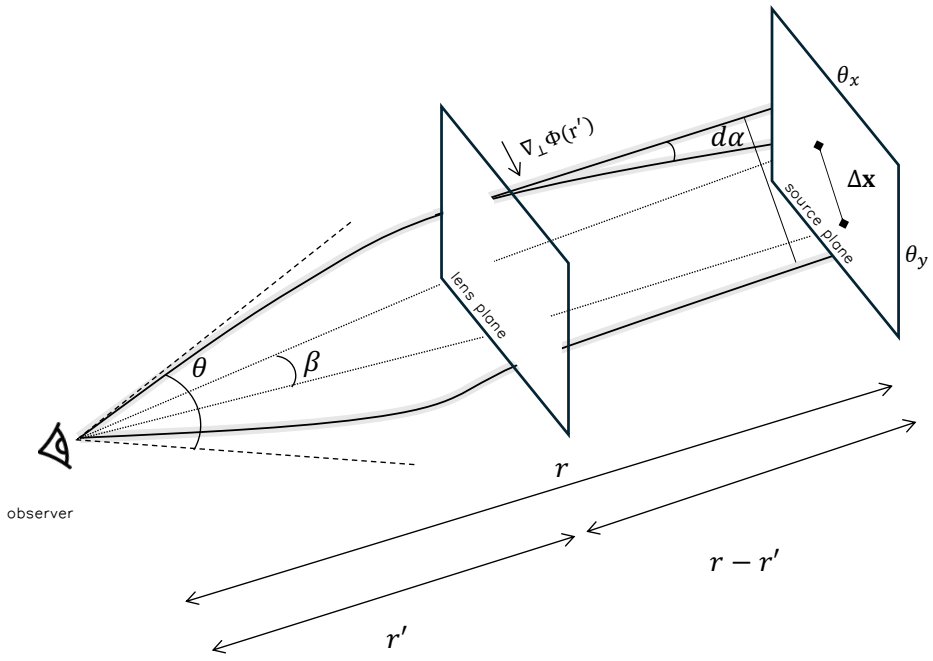


Figure 2.3: Distortion of two light rays (black and grey lines) in a weak gravitational field, converging on an observer (left). The rays emanate from the source at a distance r from the observer. An example deflector at a distance r' is shown perturbing the light's path along the transverse gradient $\nabla_\perp\Phi(r')$. The apparent angle θ is the angle at which the observer receives the rays, denoted by dashed lines. The observer records the image to be of size $\Delta\mathbf{x}$. The dotted lines form the angle β and indicate the unperturbed geodesic along which the light would travel in absence of lensing.

where δ_{ij} is the 2D Kronecker delta and

$$\psi_{ij}(\mathbf{r}) = \frac{2}{c^2}\int_0^r dr' \frac{(r-r')}{rr'}\frac{\partial^2\Phi(\mathbf{r}')}{\partial\theta_i\partial\theta_j}. \tag{2.57}$$

It is useful to define the *lensing potential*,

$$\phi(\mathbf{r}) = \frac{2}{c^2}\int_0^r dr' \frac{(r-r')}{rr'}\Phi(\mathbf{r}'), \tag{2.58}$$

which is related to Eq. 2.57 via $\psi_{ij} = \partial^2\phi(\mathbf{r})/\partial\theta_i\partial\theta_j$. The lensing potential is the integration of past-directed geodesic paths flowing from the observer. So far we have parameterised all radial distances in terms of a flat universe. For a universe with curvature, r becomes $f_K(r)$ (Eq. 2.25).

The lens equation. Another way to interpret Eq. 2.56 is in terms of the deflection angle. In absence of lensing the separation $\Delta\mathbf{x}$ would be seen by the observer at an angle $\boldsymbol{\beta} = \frac{\Delta x}{r}$. The deviation between the apparent angle $\boldsymbol{\theta} = \Delta\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ is the reduced deflection angle $\alpha(\boldsymbol{\theta})$, defining the *lens equation*

$$\boldsymbol{\beta} = \boldsymbol{\theta} - \boldsymbol{\alpha}(\boldsymbol{\theta}), \tag{2.59}$$

with

$$\alpha(\boldsymbol{\theta}) = \frac{2}{c^2}\int_0^r dr' \frac{r-r'}{r}\frac{\partial^2\Phi(\mathbf{r}')}{\partial\theta_i\partial\theta_j} = \nabla_\theta\phi. \tag{2.60}$$

For small distortions Eq. 2.55 linearises the lens equation to form an invertible, linear mapping between the unlensed source image coordinates $\boldsymbol{\beta}$ to the lensed coordinates $\boldsymbol{\theta}$:

$$A_{ij} = \frac{\partial\beta_i}{\partial\theta_j} = \delta_{ij} - \frac{\partial\alpha_i}{\partial\theta_j} = \delta_{ij} - \partial_i\partial_j\phi \tag{2.61}$$

Explicitly, the transformation reads:

$$\begin{pmatrix}\beta_1\\ \beta_2\end{pmatrix} = \begin{pmatrix}1-\kappa-\gamma_1 & -\gamma_2\\ -\gamma_2 & 1-\kappa+\gamma_1\end{pmatrix}\begin{pmatrix}\theta_1\\ \theta_2\end{pmatrix} \tag{2.62}$$

where we define the *convergence*

$$\kappa = \frac{1}{2}(\psi_{11}+\psi_{22}) = \frac{1}{2}(\partial_1\partial_1+\partial_2\partial_2)\phi \tag{2.63}$$

and *shear* components

$$\gamma_1 = \frac{1}{2}(\psi_{11}+\psi_{22}) = \frac{1}{2}(\partial_1\partial_1-\partial_2\partial_2)\phi; \qquad \gamma_2 = \psi_{12} = \partial_1\partial_2\phi. \tag{2.64}$$

The convergence is a unitless quantity, and is a measure of integrated mass density which describes the magnification and apparent brightness of the observed image. The shear describes the twisting of the light bundle through the weak lens, which changes the shape of the recorded image, and can be written as a complex number $\gamma = \gamma_1 + i\gamma_2 = |\gamma|\exp(2i\varphi)$ with φ the angle between γ_1, γ_2. Shear has spin-2 properties; rotating an elliptical source around π is the identity transformation (Kilbinger, 2015). A factor of $1-\kappa$ can be removed from $\mathbf{A}$ since this term only controls the size of the source. Cosmic shear is related to galaxy shape ellipticity, so it is convenient to define the *reduced shear*,

$$g = \frac{\gamma}{1-\kappa}, \tag{2.65}$$

which has the same spin-2 properties as γ. The observed ellipticity $\epsilon_{\rm obs}$ of a galaxy is measured via a major-to-minor axis ratio (a/b) and position angle ϕ as $\epsilon_{\rm obs} = (a-b)/(a+b)\times\exp(2i\phi)$. This is

related to the sum of g and the intrinsic ellipticity of the source galaxy ϵ_s:

$$\epsilon_{\rm obs} = \frac{\epsilon_s + g}{1 + g^* \epsilon_s} \approx g + \epsilon_s, \tag{2.66}$$

where g^* is the complex conjugate and the last equality is possible in the weak lensing regime. We can simplify further such that the true shear is approximately the reduced shear, $g \approx \gamma$. This allows us to define an observed shear $\gamma_{\rm obs} = \epsilon_{\rm obs}$, which can be viewed as a measurement of the true shear that has been degraded by shape noise stemming from the unknown source galaxy ellipticities

$$\gamma_{\rm obs} \approx \gamma + \epsilon_s, \tag{2.67}$$

where ϵ_s is $\mathcal{O}(100)$ times greater than the lensing signal per galaxy (Bartelmann & Schneider, 2001; Kilbinger, 2015; Jeffrey et al., 2021).

E- and B-modes

The Born approximation defines the shear and convergence to be functions of a single scalar potential with certain constraints. To see this, we can define the gradient of the convergence $\boldsymbol{u} = \boldsymbol{\nabla}\kappa$, whose curl vanishes $\boldsymbol{\nabla} \times \boldsymbol{u} = \partial_1 u_2 - \partial_2 u_1 = 0$, which via Eqs 2.63 and 2.64 yields second-derivative constraints for γ. A shear field fulfilling this zero-curl criterion is called an E-mode field, akin to an electric field. In reality, measured $\boldsymbol{u}$ has a non-vanishing curl component. The convergence can then be written as $\kappa = \kappa_E + i\kappa_B$, where $\nabla^2\kappa_E = \boldsymbol{\nabla} u$ and $\nabla^2\kappa_B = \boldsymbol{\nabla} \times u$ where the last term is the B-mode term. (Kilbinger, 2015). Non-zero B-modes can arise from higher-order terms in the light-propagation Eq. 2.54, data systematics, selection biases, and astrophysical effects (see e.g. Bartelmann & Maturi, 2016; Kilbinger, 2015; Krause & Hirata, 2010).

2.4.3 Linking Lensing and Structure on the Sphere

We now have the ingredients to link distortions in images on the sphere to the gravitational potential, Φ, which is related to the matter overdensity field

$$\delta = \frac{\delta\rho}{\rho} \tag{2.68}$$

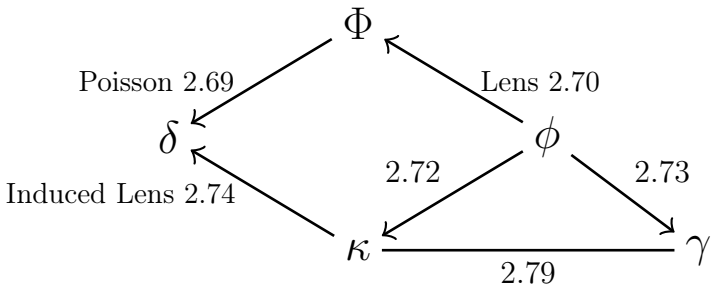


Figure 2.4: Schematic relationship between weak quantities Φ (gravitational potential), ϕ (lensing potential), δ (overdensity field), κ (convergence), and γ (shear), annotated with the corresponding equation. Arrows denote spatial second derivatives, and the line linking κ and γ illustrates harmonic relationship. Adapted from Jeffrey et al. (2024).

via Poisson's equation (Kilbinger, 2015)

$$\nabla^2_{3D}\Phi = \frac{3H_0^2\Omega_m}{2a(t)}\delta(t,\mathbf{r}) \tag{2.69}$$

where here we have used $\mathbf{r}$ as a comoving spatial coordinate, $\delta(t,\mathbf{r})$ is allowed to evolve in time, H_0 is the Hubble constant and $a(t) = R(t)/R_0 = 1/(1+z)$. We will revisit prescriptions for the growth of large-scale structure overdensities in Section 2.5.

We will parameterise the observer's past lightcone position as (χ,θ,φ), with χ as the comoving radial distance. We can rewrite the lensing potential (Eq. 2.58) as a real scalar field on the light cone with the gravitational potential Φ projected along the observer's line of sight:

$$\phi(\chi,\theta,\varphi) = \frac{2}{c^2}\int_0^\chi d\chi' \frac{f_K(\chi-\chi')}{f_K(\chi)f_K(\chi')}\Phi(\chi',\theta,\varphi), \tag{2.70}$$

which assumes the Born approximation as before. To project onto the sphere, we can integrate out the radial dependence of the potential by using the redshift distribution of source galaxies as a weighting function:

$$\phi(\theta,\varphi) = \int_0^\infty d\chi\ n(z(\chi))\ \phi(\chi,\theta,\varphi). \tag{2.71}$$

To project the lensing potential onto the sphere, we will follow the prescription described in Castro et al. (2005) and Jeffrey et al. (2021). If we let ${}_sY_{\ell m}(\theta,\varphi)$ be the spin-weight s in spherical harmonic basis functions, we can define the covariant derivative $\eth$ which increases the spin-weight, and its adjoint $\bar{\eth}$ which decreases it. With these operators we can rewrite the scalar κ and spin-2 γ (Eqs 2.63 and 2.64) as:

$$\kappa = \frac{1}{4}(\eth\bar{\eth} + \bar{\eth}\eth)\phi, \tag{2.72}$$

and

$$\gamma = \frac{1}{2}\eth\eth\phi \tag{2.73}$$

Combining Eqs 2.69 2.71, and 2.72 yields an induced lens equation which links δ and κ:

$$\kappa(\theta,\phi) = \frac{3\Omega_m H_0^2}{2c^2}\int_0^\infty d\chi\; n(z(\chi))\int_0^\chi d\chi' \frac{f_K(\chi')f_K(\chi-\chi')}{f_K(\chi)}\frac{\delta(\chi',\theta,\phi)}{a(\chi')}, \tag{2.74}$$

In harmonic space, ϕ, κ, and γ can be approximated via the harmonic coefficients $\hat{\phi}_{\ell m}, \hat{\kappa}_{\ell m}$, and $\hat{\gamma}_{\ell m}$, respectively, e.g:

$$\kappa = \sum_{\ell m} \hat{\kappa}_{\ell m}\; {}_0Y_{\ell m}, \tag{2.75}$$

where

$$\hat{\kappa}_{\ell m} = \int d\Omega \kappa(\theta,\varphi)\,{}_0Y^*_{\ell m}(\theta,\varphi). \tag{2.76}$$

Applying the operators in harmonic space, Eqs 2.72 and 2.73 become

$$\hat{\kappa}_{\ell m} = -\frac{1}{2}\ell(\ell+1)\hat{\phi}_{\ell m} \tag{2.77}$$

and

$$\hat{\gamma}_{\ell m} = \frac{1}{2}\sqrt{(\ell-1)\ell(\ell+1)(\ell+2)}\hat{\phi}_{\ell m}, \tag{2.78}$$

which together gives the harmonic coefficient relationship

$$\hat{\gamma}_{\ell m} = -\sqrt{\frac{(\ell-1)(\ell+2)}{\ell(\ell+1)}}\hat{\kappa}_{\ell m} \tag{2.79}$$

We now have a link between cosmological parameters, large-scale structure overdensities, and weak lensing quantities which we can measure on the celestial sphere. We will see in Section 2.5 how gravitational instabilities lead to structure formation, and how analytic solutions fall short of the complexity needed to characterise the universe's nonlinear effects of gravitational evolution.

2.4.4 The Lensing Two-Point Function

Limber's Approximation

Although we have established links between matter and weak lensing potentials, characterising structure along each line-of-sight in weak lensing is difficult because the underlying matter distribution is not known. Correlations in the fluctuations of these fields, however *can* be measured. If we observe a distorted image at an angle $\boldsymbol{\theta}$, then nearby images at $(\boldsymbol{\theta} + \boldsymbol{\varphi})$ should also see a similar degree of distortion. This pairwise correlation is formalised in the *two-point* correlation function of a quantity x:

$$\xi(\boldsymbol{\varphi}) = \langle x(\boldsymbol{\theta})\ x(\boldsymbol{\theta} + \boldsymbol{\varphi}) \rangle\,, \tag{2.80}$$

where the average is taken over all positions $\boldsymbol{\theta}$ and all orientations of the separation vector $\boldsymbol{\varphi}$. This again encodes the cosmological principle of statistical isotropy–no particular direction should appear more dense on average than any other. The interpretation of this statistic is the "clumpiness" of the sky at different angular separation scales. It is often convenient to look at the correlation function's Fourier-space analogue, or *angular power spectrum*:

$$C_\ell = \int d^2\boldsymbol{\varphi}\ \xi(\boldsymbol{\varphi}) e^{-i\boldsymbol{\ell}\cdot\boldsymbol{\varphi}}, \tag{2.81}$$

where $\boldsymbol{\ell}$ is the two-dimensional wave vector conjugate to $\boldsymbol{\varphi}$. Weak lensing power spectra are often calculated making use of Limber's Approximation, which states that a two-dimensional quantity

$$x(\boldsymbol{\theta}) = \int_0^{\chi_s} d\chi\ W(\chi)\ y(\chi\boldsymbol{\theta},\ \chi), \tag{2.82}$$

is a projection via a weight function $W(\chi)$ of a three-dimensional quantity $y(\boldsymbol{r})$, then the angular power spectrum of the 2D x can be related to the power spectrum of the 3D y:

$$C_x(\ell) = \int_0^{\chi_s} d\chi\ \frac{W^2(\chi)}{\chi^2} P_y\left(\frac{\ell}{\chi}\right), \tag{2.83}$$

where the power spectrum $P_y(k)$ is computed at the 3D wavevector $k = \ell/\chi$ (Bartelmann & Maturi, 2016; Coles & Lucchin, 2002). This approximation is valid so long as fluctuations in the quantity y are much smaller than the width of the projection.

Adapting this to the weak lensing case, the induced convergence (Eq 2.74) takes precisely this form with the weight function (and flat co-moving coordinate)

$$W(\chi) = \frac{3}{2}\frac{H_0^2}{c^2}\Omega_m \frac{\chi'(\chi - \chi')}{a\chi} \tag{2.84}$$

What this means is that we can link the weak lensing angular power spectrum to the power spectrum P_δ of the underlying 3D overdensity field. This connection allows us to glean some additional physical insight about structure and the cosmological parameters. When the matter density field fluctuations are small ($\delta << 1$), the largest scales grow linearly according to some growth function $D_+(a)$, which means that on large scales, the matter power spectrum varies quadratically with D_+; $P_\delta \propto D_+^2$. On small scales the P_δ changes with respect to nonlinear effects induced by gravitational instability. If we ignore small nonlinear scales, the power spectrum can be written as a shape function $\mathcal{P}$ times an amplitude, which is calibrated at $a = 1$ and is obtained by integrating the extrapolated P_δ via (Bartelmann & Maturi, 2016):

$$\sigma_8^2 = \int_0^\infty \frac{k^2 dk}{2\pi^2} P_\delta(k) W_8^2(k), \tag{2.85}$$

where we have used $W_8(k)$ to denote a filter which suppresses modes smaller than $8h^{-1}$ Mpc. This parameter can be interpreted as the root-mean square matter fluctuation, averaged over a sphere of radius $8h^{-1}$ Mpc. If we set $D_+ = 1$ at the present, we can relate the 3D power spectrum as

$$P_\delta(k) = \sigma_8^2 D_+^2(a)\mathcal{P}(k), \tag{2.86}$$

and by extension can write the convergence angular power spectrum as

$$C_\kappa(\ell) = \frac{9}{4}\left(\frac{H_0}{c}\right)^4 \Omega_m^2 \sigma_8^2 \int_0^\chi \chi' \left[\frac{D_+(a)}{a}\frac{\chi'(\chi' - \chi}{\chi'}\right] \mathcal{P}\left(\frac{\ell}{\chi}\right) \tag{2.87}$$

This equation presents several key takeaways:

1. $\hat{C}_\kappa(\ell)$ can be measured from shear observations
2. The shape of the weak lensing angular power spectrum depends on the shape of the matter power spectrum, so can be inferred from data
3. The weight function can be linked to the growth factor D_+ for inference
4. The amplitude of the lensing spectrum is proportional to the square of the matter-density

parameter Ω_m times the matter power spectrum amplitude σ_8, so provides a direct observable with which to infer these quantities

The last point is worth stressing: from a two-point measurement alone, we can directly infer two very important cosmological parameters. Their combination is also intuitive: Ω_m is the mean matter density in the universe, while σ_8 quantifies how clumped that matter is. Their linear relationship in $\Omega_m^2\sigma_8^2$ means that measurement of the two-point function *alone* cannot distinguish the two, resulting in a statistical *degeneracy* in inference. Measuring $\Omega_m - \sigma_8$ and resolving this type of degeneracy via new observables and techniques is a key aspect of modern cosmology. We will revisit the two-point function and higher-order statistics in Section 2.5.6.

2.5 Gravitational Instability & Structure Formation

From precise cosmic microwave background (CMB) measurements, we know that the universe was close to homogeneous and isotropic at the time of Recombination, when the universe was only 300-400 thousand years old. Today, however, we observe extremely dense regions of the universe (galaxies, stars, clusters), as well as very empty ones (cosmic voids). The "Holy Grail" of cosmology today is to figure out how exactly these structures evolved from such a uniform field. The answer to the nonlinear matter structure lies in the effects of gravitational instability on different distance scales, which arises from the attractive gravitational potential in competition with the pressure matter exerts on its surroundings.

2.5.1 The Matter Fluid Approximation

Just as before, it helps to describe the "stuff" in our universe, in this case matter, as a perfect fluid. That allows us to completely describe the distribution of matter *at any time* using just three quantities: the energy density, $\epsilon(\boldsymbol{x}, t)$, the entropy per unit mass, $S(\boldsymbol{x}, t)$, and the vector field of matter 3-velocities, $\mathbf{V}(\boldsymbol{x}, t)$. In each of these expressions we have condensed 3-space representation x^i into the concise $\boldsymbol{x}$. All of these quantities obey the *hydrodynamical equations*, which in principle allow us to completely model the matter distribution over time and space. The equations are:

Euler Equations: In the fluid approximation, a small piece of mass ΔM can be acted on by the force

of gravity, $\mathbf{F}_{\rm grav} = -\Delta M \cdot \nabla\Phi$, where Φ is the gravitational potential, and the force of pressure, $\mathbf{F}_{\rm pr} = -\nabla p \cdot \Delta V$ and V is a finite volume. Using Newton's force law, we obtain the Euler Equations

$$\Delta M \cdot \mathbf{g} = \mathbf{F}_{\rm grav} + \mathbf{F}_{\rm pr} \tag{2.88}$$

$$\implies \frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V}\cdot\boldsymbol{\nabla})\mathbf{V} + \frac{\nabla p}{\epsilon} + \nabla\Phi = 0 \tag{2.89}$$

Continuity Equation: For a fixed volume element, ΔV, the rate of change of mass $M(t)$ over time can be related to both the change in its energy density through that volume and the flux of matter through the surface bounding the volume ($d\boldsymbol{\sigma}$), yielding:

$$\frac{dM(t)}{dt} = -\int_{\Delta V} \nabla(\epsilon\mathbf{V})dV = \oint \epsilon\mathbf{V}\cdot d\boldsymbol{\sigma}, \tag{2.90}$$

which are only consistent if

$$\frac{\partial\epsilon}{\partial t} + \nabla(\epsilon\mathbf{V}) = 0. \tag{2.91}$$

Conservation of Entropy: From thermodynamics, we know that the entropy of matter is conserved in the absence of dissipation (as expected in such a large volume):

$$\frac{dS(\boldsymbol{x}(t),t)}{dt} = \frac{\partial S}{\partial t} + (\mathbf{V}\cdot\boldsymbol{\nabla})S = 0 \tag{2.92}$$

Poisson Equation: The equation that describes the gravitational potential acting on the fluid is:

$$\nabla^2\Phi = 4\pi G\epsilon \tag{2.93}$$

where G is Newton's gravitational constant. Taken together with the equation of state (Eq. 2.16), these seven equations allow us to specify the matter density field at any time. The solutions to the hydrodynamical equations are highly nonlinear. However, for small perturbations around an otherwise homogeneous and isotropic background we can make linear approximations to gain an intuition about how matter behaves as a fluid under the influence of gravity.

2.5.2 Linear Perturbations in an Expanding Universe

We know from observation and our previous derivation of the Friedmann Equations that we live in homogeneous, isotropic, and expanding universe. Under these assumptions, our velocity field obeys the Hubble Law (Mukhanov, 2005):

$$\mathbf{V} = \mathbf{V}_0 = H(t) \cdot \boldsymbol{x} \tag{2.94}$$

and the background energy density is just a function of time:

$$\epsilon = \epsilon_0(t) \tag{2.95}$$

Another consideration we need make is the fact that the Eulerian coordinates $\boldsymbol{x}$, expand with the expansion factor, $a(t)$:

$$\boldsymbol{x} = a(t)\mathbf{q} \tag{2.96}$$

where $\mathbf{q}$ is the set of comoving Lagrangian coordinates. This means that the spatial and time derivatives of $\boldsymbol{x}$ can be written as:

$$\left(\frac{\partial}{\partial t}\right)_{\boldsymbol{x}} = \left(\frac{\partial}{\partial t}\right)_{\mathbf{q}} - (\mathbf{V}_0 \cdot \nabla_{\boldsymbol{x}}) \quad \text{and}$$
$$\nabla_{\boldsymbol{x}} = \frac{1}{a}\nabla_{\mathbf{q}},$$

respectively. Inserting these expressions into the hydrodynamical equations returns the Friedmann equations, explored in the previous section. We now *perturb* the elements of our system (ignoring entropy changes) about their equilibrium values (denoted by the subscript "0"):

$$\epsilon = \epsilon_0 + \delta\epsilon(\boldsymbol{x}, t), \qquad \mathbf{V} = \mathbf{V}_0 + \delta\mathbf{v}$$
$$p = p_0 + \delta p = p_0 + c_s^2\delta\epsilon, \qquad \Phi = \Phi_0 + \delta\Phi$$

where we've introduced c_s, the speed of sound in a pressure-filled medium. Substituting these expressions into our hydrodynamical equations, and defining the *fractional amplitude* of the energy density perturbations, $\delta = \delta\epsilon/\epsilon_0$, we can condense our description down to the closed form equation:

$$\ddot{\delta} + 2H\dot{\delta} - \frac{c_s^2}{a^2}\nabla^2\delta - 4\pi G\epsilon_0\delta = 0 \tag{2.97}$$

This equation describes gravitational instability in terms of the amplitude of the energy perturbations, which are linked to the underlying matter distribution (see Mukhanov (2005) or Ryden (2003) for detailed derivation).

2.5.3 The Jeans' Length and Adiabatic Perturbations

A useful solution to Equation 2.97 arises when we continue to assume the absence of entropy perturbations. We see that the coefficients of the differential equation do not depend on the spatial coordinate $\mathbf{q}$, so we are free to take the Fourier transform to obtain a differential equation in time:

$$\ddot{\delta}_{\mathbf{k}} + 2H\dot{\delta}_{\mathbf{k}} - \left(\frac{c_s^2}{a^2} - 4\pi G\epsilon_0\right)\delta_{\mathbf{k}} = 0 \tag{2.98}$$

such that every Fourier mode is related via $\delta = \delta_{\mathbf{k}}(t)e^{i\mathbf{kq}}$. This damped harmonic equation has two independent solutions, which depend entirely on the Jeans' length of the perturbations:

$$\lambda_J = \frac{2\pi a}{k_J} = c_s\sqrt{\frac{\pi}{G\epsilon_0}} \tag{2.99}$$

This critical length can be thought of as the scale at which a density perturbation is held stable against collapse by the pressure of the system. For small scales, $\lambda < \lambda_J$, pressure dominates over gravity, and we obtain an oscillating sinusoidal solution for the perturbations in the form of *sound waves*. For large scales, $\lambda > \lambda_J$ (small k), gravity dominates the perturbations, and the solution can be expressed as:

$$\delta = C_1 t^{2/3} + C_2 t^{-1} \tag{2.100}$$

where C_1, C_2 are constants of integration (Mukhanov, 2005). The important takeaway here is that the scale of energy density perturbations, δ, only grow proportionally to a the scale factor, a in a flat, matter-dominated universe (as is thought to be the case after Recombination (Pritchard & Loeb, 2012)). What this means is that our density perturbations would need to be substantially large ($\delta \geq 10^{-3}$) at very early times (for $z \geq 1000$). This reasoning places a *theoretical constraint* on both the rate of structure growth, as well as the scale of initial inhomogeneities in the universe (which can be explained via inflationary theory).

2.5.4 Nonlinear Perturbations and the Zel'dovich Approximation

The linear perturbations we discussed in the previous section had an underlying spherical symmetry assumption to them. However, we know from observation that matter is distributed in highly asymmetrical ways, making it more difficult to describe gravitational instability analytically. However, exact descriptions of nonlinear evolution exist for a few limiting cases. Understanding how nonlinearities grow is made easier by recasting the hydrodynamical equations in terms of the *matter elements*:

$$\mathbf{x} = \mathbf{x}(\mathbf{q}, t) \tag{2.101}$$

With this coordinate change, a *strain tensor* is needed to specify the derivatives of the matter elements, x^i with respect to the Lagrangian coordinates, $\mathbf{q}$:

$$J^i_k(\mathbf{q}, t) = \frac{\partial x^i(\mathbf{q}, t)}{\partial q^k} \tag{2.102}$$

This matrix helps us to understand how matter changes with respect to the comoving Lagrangian coordinates. After making this change of coordinate basis and ignoring the effect of pressure on scales larger than the Jeans' length, Equation 2.97 takes the form:

$$(\ln J)^{\cdot\cdot} + \mathrm{tr}\left[(\dot{\mathbf{J}} \cdot \mathbf{J}^{-1})\right] + 4\pi G\widetilde{\rho}_0 J^{-1} = 0 \tag{2.103}$$

where $\widetilde{\rho}_0(\mathbf{q}) = \epsilon(\mathbf{q}, t) \det \mathbf{J}$ is an arbitrary time-independent function of $\mathbf{q}$. This equation can be solved exactly for $\mathbf{J}$ for a few limiting cases. Zel'Dovich (1970) proposed a description of nonlinear behaviour by imposing a one-dimensional nonlinear perturbation onto a three-dimensional Hubble expansion. For a perturbation in one coordinate, q^i the relation between matter and Lagrangian coordinates is:

$$x^i = a(t)(q^i - f^i(q^j, t)) \tag{2.104}$$

where $f^i = \partial\psi/\partial q^i$, describes a perturbation in the potential ψ, for peculiar velocities. If we limit the perturbation to just one coordinate, q^1, the strain tensor takes the form

$$\mathbf{J} = a(t) \begin{bmatrix} 1 - \lambda(q^1, t) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.105}$$

where $\lambda(q^1, t) = \partial f^1/\partial q^1$. For $\widetilde{\rho}_0(\mathbf{q}) = \text{const}$ we recover the first Friedmann equation and Equation 2.97, showing that the one-dimensional solution coincides with linear perturbations in pressureless matter (Mukhanov, 2005). More generally, the strain tensor can be formulated in terms of functions $\alpha(q^i),\ \beta(q^i),\ \gamma(q^i)$:

$$\mathbf{J} = a\mathbf{I} - a\delta_i \begin{bmatrix} \alpha & 0 & 0 \\ 0 & \beta & 0 \\ 0 & 0 & \gamma \end{bmatrix} \tag{2.106}$$

where $\mathbf{I}$ is the identity matrix. When Zel'dovich's $\mathbf{J}$ is inserted into the hydrodynamical equations, the resulting description of the density in terms of the Lagrangian coordinates is:

$$\epsilon(q,t) = \frac{\left[1 - ((\alpha\beta + \alpha\gamma + \beta\gamma)\delta_i^2 - 2\alpha\beta\gamma\delta_i^3)\right]}{(1-\alpha\delta_i)(1-\beta\delta_i)(1-\gamma\delta_i)} \tag{2.107}$$

Different values of α, β, γ in a given region describe the formation of peaks, voids, filaments, and sheets. For example, in regions where $\alpha >> \beta, \gamma$, one-dimensional collapse occurs, creating two-dimensional "sheets" of density (also called Zel'dovich pancakes (White, 2014)). When $\alpha \approx \beta >> \gamma$, the collapse is two-dimensional, squeezing out matter into one-dimensional filaments. Finally, when all three density descriptors are on the same order, the collapse of structure is spherically-symmetric. For small, Gaussian perturbations, the Zel'dovich approximation exactly predicts a structure usually dubbed the "Cosmic Web," in which massive cosmic voids are separated by filaments connecting large sheets. A numerically-simulated image of this three-dimensional structure is shown in Figure 2.5, in which filaments and sheets form the more saturated regions. The ZA approximation, while a simple description, is accurate at large scales and describes the growth of structure in the linear regime. However, this approximation breaks down in the small, nonlinear regime. Inspecting Eq 2.107, we see that when one of the deformation tensor's eigenvalues, e.g. $\delta(t_{\text{sc}}) = 1/\alpha$ at some time t_{sc}, an event called *shell crossing* occurs in which the approximation to the density in that region becomes infinite. Physically this corresponds to when particle trajectories have crossed; two points with different Lagrangian coordinates overlap at the same Eulerian coordinate, and the mapping Eq 2.103 is no longer unique. The ZA approximation can still be used to describe the formation of structure in the quasi-nonlinear regime, so long as appropriately small scales are filtered out to remove shell-crossing effects.

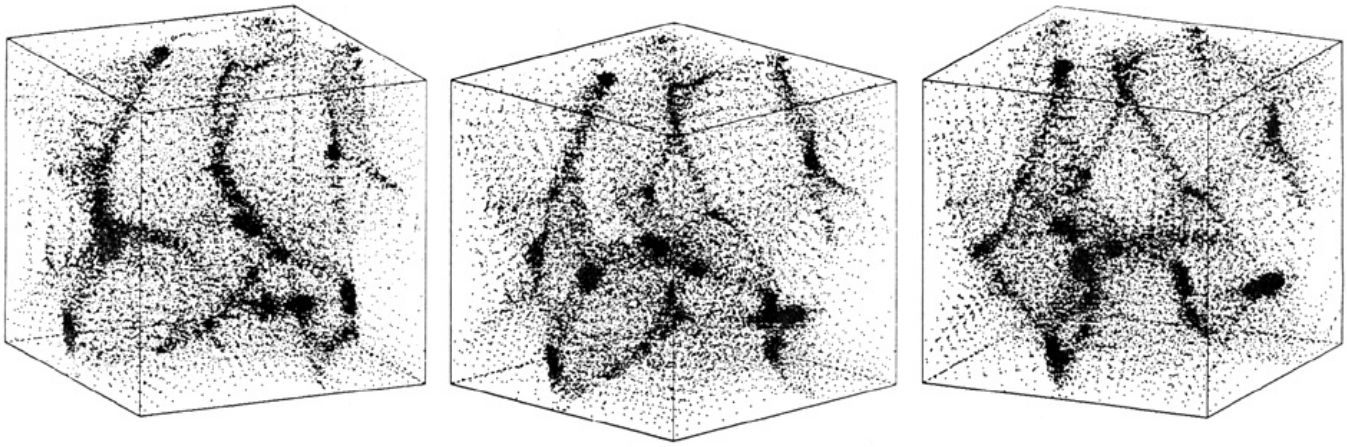


Figure 2.5: Numerical simulation of nonlinear structure often dubbed the "Cosmic Web". Denser regions like filaments and sheets (darker regions) are separated by sparse voids. Inferring this structure is considered one of the "Holy Grails" of modern cosmology, since estimates of the density field provide constraints on both cosmological and astrophysical models. Figure adapted from Efstathiou & Silk (1983).

2.5.5 Exact Gravity Simulation

Analytic methods for describing structure formation quickly break down in the nonlinear regime due to the complexity of gravitational interactions between particles in the fluid. To accurately describe the universe we observe at the level of weak lensing κ or galaxy clustering, we require a higher-resolution picture of structure overdensity. The most exact solution comes in the form of direct summation N-body simulations, where the cosmological fluid (dark matter) is represented by a discrete set of particles whose pairwise interactions are computed and summed to update the particle's velocity and position using a Newton algorithm (Coles & Lucchin, 2002). The pairwise force between particles i and j is computed as

$$\boldsymbol{F}_{ij} = \frac{Gm^2(\mathbf{x}_j - \mathbf{x}_i)}{(\varepsilon^2 + |\mathbf{x}_i - \mathbf{x}_j|^2)^{3/2}}, \tag{2.108}$$

where the particles' masses are both m and a *softening length* parameter ε prevents infinite forces at zero particle separation. The softening length replaces point masses with extended bodies of radius ε, which formalises the fluid approximation. N-body simulations are usually initialised with a ZA overdensity field constructed from Gaussian initial conditions, which accurately captures large-scale behaviour. The particle trajectories are then evolved using very small timesteps. For a system of N particles, each particle's acceleration requires computing $(N-1)$ interactions, which requires a total

of $N(N-1)/2$ evaluations of Eq 2.108 *at each timestep*, which is incredibly CPU-intensive. Recent advances in computation have made larger suites of these simulations available at variable cosmologies (Potter et al., 2016; Villaescusa-Navarro et al., 2020b; Kacprzak et al., 2023) and even smaller scales to incorporate the hydrodynamical effects describing galaxy formation (Villaescusa-Navarro et al., 2021; Ni et al., 2023).

Another approach to make cosmological N-body simulations more efficient is the particle-mesh simulation, which assigns mass points to a mesh and obtains the potential required for Poisson's equation in Fourier space, which leads to a considerable computational speed-up (Li et al., 2022; Modi et al., 2020) on the order $N \log N$ (Coles & Lucchin, 2002). This approach requires that the simulation box has periodic boundary conditions to facilitate the Fourier transform, and has worse force resolution on smaller scales due to the finite spatial size of the mesh grid compared with direct summation. Adaptive strategies like mesh-refinement can increase the resolution in dense regions like clusters and filaments to improve accuracy (Teyssier, 2002).

2.5.6 The Power Spectrum: Cosmology's Workhorse

The treatment of ZA perturbations in the previous section assumed a spherical symmetry. However, Equation 2.97 can be used to describe low-amplitude perturbations of any shape (Ryden, 2003). Our analysis of gravitational instability in an expanding universe helped us gain some intuition about the nature of the primordial instabilities that gave rise to the structure we see today. If we look at a sufficiently early time in the universe's history, the perturbations will be small enough such that we can treat the universe at most scales as isotropic and homogeneous, meaning the FLRW metric applies. We can then look at the evolution of density with respect to a co-moving coordinate $\mathbf{r}$, such that the proper distance $d_p = a(t)\mathbf{r}$ is normalized to distances measured today.

When describing the growth of large-scale structure, it is helpful to describe fluctuations as a three-dimensional field in Fourier space in terms of the co-moving coordinate :

$$\delta(\mathbf{r}) = \frac{V}{(2\pi^3)} \int \delta_{\mathbf{k}} e^{-i\mathbf{k}\cdot\mathbf{r}} d^3k \tag{2.109}$$

where we define the Fourier components:

$$\delta_{\mathbf{k}} = \frac{1}{V} \int \delta(\mathbf{r}) e^{i\mathbf{k}\cdot\mathbf{r}} d^3r \tag{2.110}$$

We used this representation before to study the solutions to the hydrodynamical equations. Each Fourier component then obeys Equation 2.97, which we can parameterize in terms of the matter fraction (for matter-dominated cosmology):

$$\ddot{\delta}_{\mathbf{k}} + 2H\dot{k}_{\mathbf{k}} - \frac{3}{2}\Omega_m H^2 \delta_{\mathbf{k}} = 0 \tag{2.111}$$

This equation still describes longer wavelength perturbations even after modes below the Jean length have collapsed (Ryden, 2003; Mukhanov, 2005), so that we can continue using linear perturbation theory to describe the largest structures in the universe, even after galaxies and clusters have formed. The squared amplitudes of the Fourier components form the power spectrum:

$$P(k) = \langle |\delta_{\mathbf{k}}|^2 \rangle \tag{2.112}$$

where $\langle \cdot \rangle$ is the average taken over all angular orientations of the $\mathbf{k}$ coordinate. Assuming homogeneity and isotropy, we shouldn't expect to lose any information about the density in this procedure. Each of the Fourier components can be written as:

$$\delta_{\mathbf{k}} = |\delta_{\mathbf{k}}| e^{i\phi_{\mathbf{k}}} \tag{2.113}$$

where $\phi_{\mathbf{k}}$ is the phase. When these phases are uncorrelated, the field is a Gaussian Random Field, meaning the density perturbation at a randomly selected location in space is drawn from the Gaussian probability distribution:

$$p(\delta) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-\delta^2}{2\sigma^2}\right) \tag{2.114}$$

If $\delta(\mathbf{r})$ is a homogenous and isotropic Gaussian field, as is predicted by inflationary models (Mukhanov, 2005), the width of the distribution σ can be calculated from $P(k)$:

$$\sigma = \frac{V}{2\pi^2} \int_0^\infty P(k) k^2 dk \tag{2.115}$$

The Gaussianity of the field also allows us to write the density power spectrum in terms of a simple power law for any scale:

$$P(k) \propto k^n \tag{2.116}$$

For Gaussian fields, the (measured) power spectrum is a sufficient statistic–no other numbers are needed to capture the variability in the distribution. This is (almost) the case for the CMB ini-

tial conditions as predicted by inflationary theory, but there are efforts to predict deviations from Gaussianity and link them to alternative inflationary models (see e.g. Collaboration et al., 2019).

2.5.7 The Need for Higher-Order Statistics

But what about two-point measurements for the late universe and large-scale structure ? In Section 2.4.4 we were able to link the weak lensing angular power spectrum to the underlying 3D power spectrum, as well as cosmological parameters. The two-point function of galaxy clustering on the sky also traces the shape of P_δ. However, due to the nonlinear evolution under gravity, the large-scale structure is significantly non-Gaussian on small scales, meaning the two-point function is no longer a sufficient statistic for characterising δ. To see this, consider the Fourier representation of the overdensity field as the sum of plane waves (Coles, 2001):

$$\delta(\mathbf{x}) = \sum_k \tilde{\delta}(\mathbf{k}) \exp\left(i\mathbf{k}\cdot\mathbf{x}\right). \tag{2.117}$$

The Fourier transform of the density field is complex; each mode is comprised of an amplitude $|\tilde{\delta}(\mathbf{k})|$ and a phase $\phi_\mathbf{k}$. Gaussian fields have *independently* distributed real and imaginary components, which means the phases are drawn uniformly at random from $[0, 2\pi]$. For small, linear, early-time fluctuations, the Fourier modes evolve independently and the field remains Gaussian, so all of the information is stored in the amplitudes captured by the two-point function. However, as gravitational instability takes over and structure begins to form, these Fourier modes begin to couple (Peebles, 1980). Measuring the two-point alone is not sufficient to describe non-Gaussian overdensity fields. We can see this graphically in Fig 2.6, in which the phases of a cosmological N-body simulation (left) are shuffled amongst its Fourier modes to produce the cloudy figure on the right. Since the amplitudes remain unchanged, both images have the same measured power spectrum, $P(\mathbf{k}) \propto |\tilde{\delta}(\mathbf{k})|^2$, as well as the same amplitudes for all wavevectors.

However, the two images clearly have vastly different morphologies. We also know that this morphology is produced by the formation of structure and underlying cosmological parameters. If both images were produced by two different models, one would not be able to tell based on two-point measurement alone. To exploit the the patterns in large-scale structure, we must consult *higher-order statistics*. Quantities such as the bispectrum (Verde et al., 2000; Coles & Lucchin, 2002; Peebles, 1980; Scoccimarro et al., 1999; Matarrese et al., 1997), which captures quadratic phase coupling from

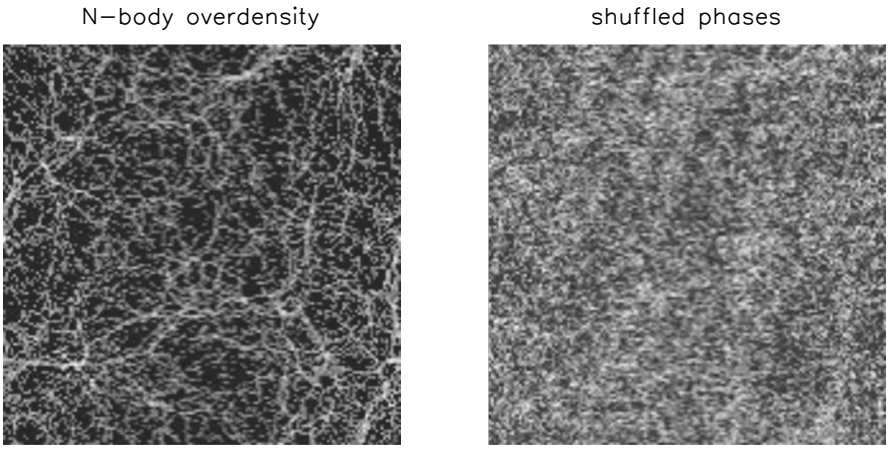


Figure 2.6: Higher-order statistics can help distinguish between cosmologies. An N-body simulation slice (*left*) and the same simulation with shuffled Fourier mode phases (*right*). The two images have the same measured power spectrum but very different morphologies. Adapted from Coles (2001).

the density field Fourier representation above, and vanish when the field becomes Gaussian. Defining

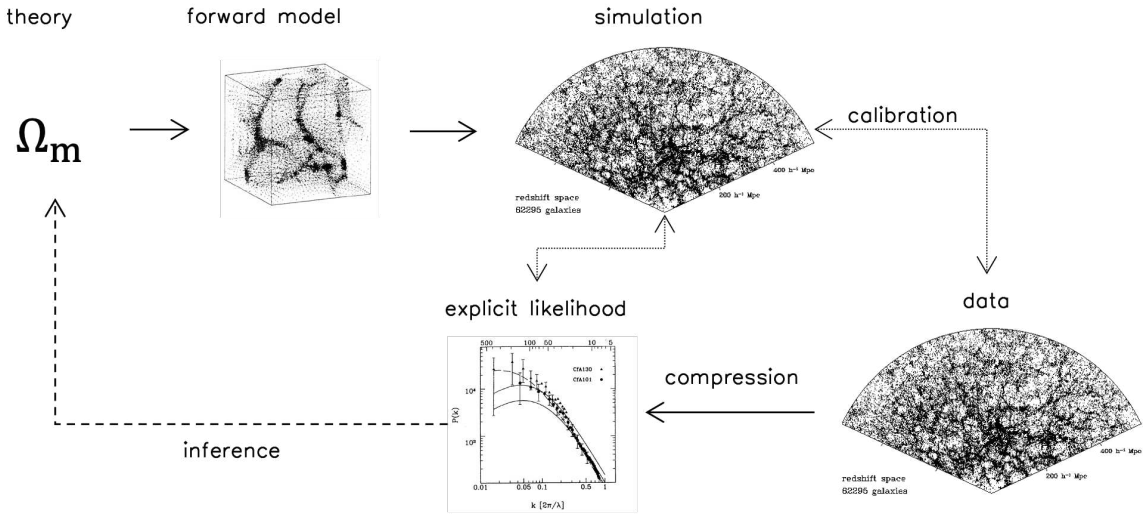


Figure 2.7: Traditional cosmological inference incorporates simulations as a means to calibrate templates for explicit likelihoods around statistics like the two-point function in an ad-hoc fashion. Simulations do not enter the inference except in cases of calibration like parameter-independent covariance estimation.

higher-order statistics like the bispectrum can link analytic descriptions with physical signals to capture more information from the non-Gaussian LSS. The challenge for *traditional* statistical methods is finding transformations of the data t *whose sampling distributions are known* such that a likelihood function $p(t|\theta)$ can be evaluated. This will be covered comprehensively in subsequent chapters.

2.5.8 The New Cosmological Objective

Beginning with General Relativity, we have laid out a framework for describing how photons trace structure in an expanding universe, as well as prescriptions for modelling the underlying nonlinear cosmic web. Along the way we have collected the "known unknowns" into a small number of free parameters like Ω_m, w, and σ_8. The only way to measure exotic quantities like Dark Energy is to *infer* their properties from data. With only one dataset (the Universe), this requires adopting a Bayesian perspective and casting cosmological inference as an inverse problem: given that we observe *this* Universe, which parameters (or model) is *most probable* ? Traditionally, cosmological

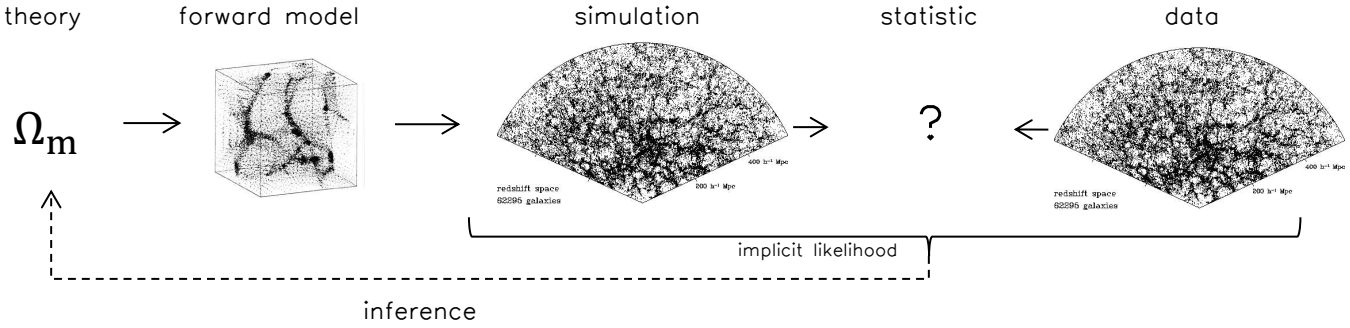


Figure 2.8: Implicit cosmological inference is an inverse problem linking theory to simulations and comparing simulations directly with observed data using a computable statistic. This whole process can be optimised for speed (via emulation) and information capture (via neural statistics).

inference requires linking theory to an observable with a known *likelihood* or sampling distribution to be compared against data (Fig 2.7). Simulations are used to validate modelling choices and to estimate covariance matrices, but were not probabilistically compared to data in a formal manner due to high dimensionality and unknown sampling distributions. Recent advances in computation and statistics have made it possible to *automate* the inference process by using simulations directly in the inference process, and to define and even *learn* higher-order statistics to compare simulations to reality. This is called *simulation-based* or *implicit inference* (Fig. 2.8). We will build up these methods in the next two chapters.

CHAPTER 3

PROBABILITY & IMPLICIT INFERENCE

The Bayesian approach to science treats all quantities in a process as random variables; some quantities are known more precisely than others, and some models are preferred over others. Most scientific analyses concern the description of some data or signal $\mathbf{d}$ that is generated from some process or theory, which we'll call a *model* M. This theory is (hopefully) controlled by some meaningful quantity, whether it be a set of parameters, an input vector, or both, which we'll call θ.

3.1 Bayesian Inference

We can illustrate this process using the flowchart in Fig. 3.1. The parameters θ are chosen according to some prior (whether physical or heuristic) $p(\theta)$. Any other random effects that are not of interest are represented by the nuisance term η drawn from $p(\eta)$. These quantities control the model $M : \theta, \eta \mapsto \mathbf{d}$ that outputs the data. Drawing many parameters and pairing them with their output data form samples of tuples $\{(\theta_i, \mathbf{d}_i)\}$ from the *joint distribution* $p(\theta, \mathbf{d})$. We illustrate a one-dimensional projection of the joint distribution in Fig. 3.1. Scatter in the data dimension indicates variability due to noise and stochastic features present in the model M.

3.1.1 The Joint Distribution and Bayes' Theorem

Now that we have our model that produces data that "looks like" our real data and a prior distribution of plausible parameters, we can turn to inference. We observe some data $\mathbf{d}^*$, which falls into the scatter point cloud from our set of simulations. If the data fell outside of the cloud, this might indicate a model that does not describe the observation–we'll get to this in a bit. We can then define

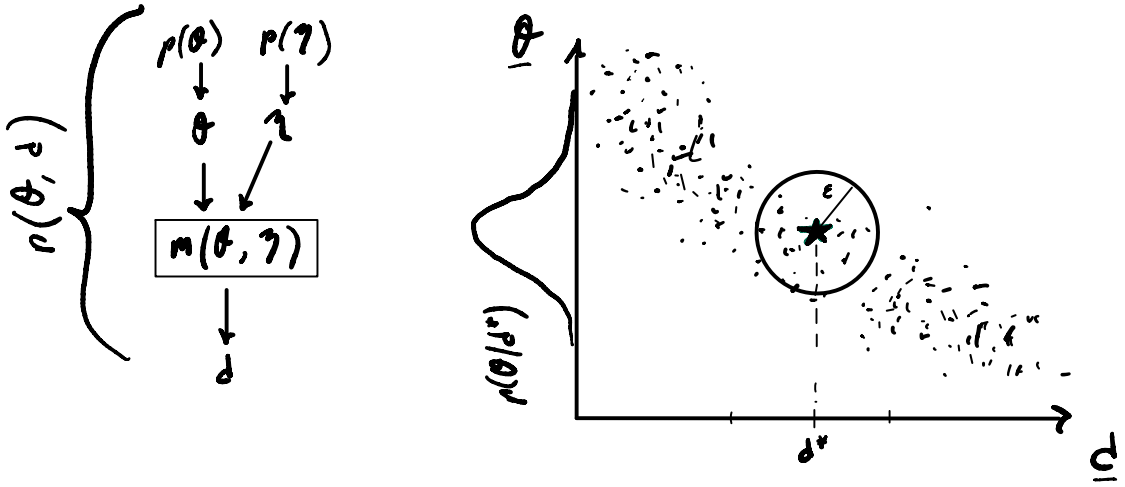


Figure 3.1: Sketch of the joint parameter-data distribution and implicit inference in one dimension.

a distance metric $\varrho(\mathbf{d}_i, \mathbf{d}^*)$ from our target and maximum distance ε: if $\varrho_i < \varepsilon$ we *accept* a simulation and its parameter value; the rest we discard.[1] The distribution of accepted parameters then forms our *posterior* distribution $p(\theta|\mathbf{d}^*)$, since we have "sliced" through the cloud (conditioned) on the observed data. The algorithm we have just illustrated is called approximate Bayesian Computation (ABC), and can be done for this one-dimensional case by hand.

Notice that we arrived at the posterior distribution $p(\theta|\mathbf{d}^*)$ *without* specifying some form for the likelihood, as many Bayesian introductions do. This is because the likelihood is *implicitly* present in the forward model connecting parameters to data. We can now introduce Bayes' Theorem formally:

$$p(\theta|\mathbf{d}, M) = \frac{p(\mathbf{d}|\theta, M)p(\theta|M)}{p(\mathbf{d}|M)} \tag{3.1}$$

Where the two terms we've avoided so far using ABC were the likelihood $p(\mathbf{d}|\theta)$ and evidence $p(\mathbf{d})$. We've also conditioned all of the probabilities on the model, but drop M for clarity until model comparison. All of the quantities can be derived from the joint distribution, $p(\theta, \mathbf{d})$, illustrated for a single model in Fig. 3.2. In our previous example, our likelihood (also called the sampling distribution) $p(\mathbf{d}|\theta)$ was implicitly defined by our forward model M, evaluated at a particular parameter value θ^*. We also allowed the model to be functions of other nuisance parameters η, which are

[1] In Figure 3.1 we illustrate the ε criterion as a ball since in most cases the data is higher-dimensional

implicitly marginalised when we isolate our joint samples to be $(\theta, \mathbf{d})$:

$$p(\mathbf{d}|\theta) = \int p(\mathbf{d}|\theta, \eta)p(\eta|\theta)d\eta, \tag{3.2}$$

where the term $p(\eta|\theta)$ reduces to $p(\eta)$ if η has no θ dependence in the model. The "peakiness" of the likelihood describes how sensitive the data is at a particular value of θ within the chosen model. We quantify this in Section 4.1. The posterior distribution[2] in Fig. 3.2 is the slice at $\mathbf{d} = \mathbf{d}^*$. In our earlier illustration we obtained this posterior from forward model samples over a prior.

3.1.2 Implicit vs Explicit Inference

It is worth stopping here to remark that we have successfully built an intuition for how to interpret a Bayesian posterior without formally specifying analytic forms for its constituent distributions, instead only assuming that we have access to samples from the joint distribution. This is called *implicit* or *simulation-based* inference, and forms the core basis for the techniques discussed in this thesis. Our miniature opening demonstration relied on a hyperparameter ε to perform ABC, which is a form of *density estimation* to evaluate a valid posterior distribution over accepted simulations in the $\theta - \mathbf{d}$ plane. We will cover this in detail in Section 5.1.2.

Explicit inference, on the other hand, requires a valid (usually analytic) parametrisation of the likelihood and prior, from which the posterior $p(\theta|\mathbf{d}) \propto p(\mathbf{d}|\theta)p(\theta)$ can estimated via a *sampling scheme*, which usually follows the general prescription: i) draw proposed model parameters from the prior $\theta^j \sim (\theta)$ ii) assign a weight to the observed data by evaluating the (log)-likelihood $\ln p(\mathbf{d}|\theta^j)$ iii) accept or reject the draw based on a Markov Chain Monte Carlo (MCMC) algorithm (Robert & Casella, 2011). Computational advances in MCMC techniques (e.g. Foreman-Mackey et al., 2013; Feroz et al., 2009; Phan et al., 2019a) have made it possible to sample enormous hierarchies of random variables for exact explicit inference (given enough samples can be drawn), and will serve as an important test bench for grading the fidelity of the novel implicit inference techniques presented in this work.

[2]For this illustration we obtained the 1D posterior using ABC to collect simulations near the target data.

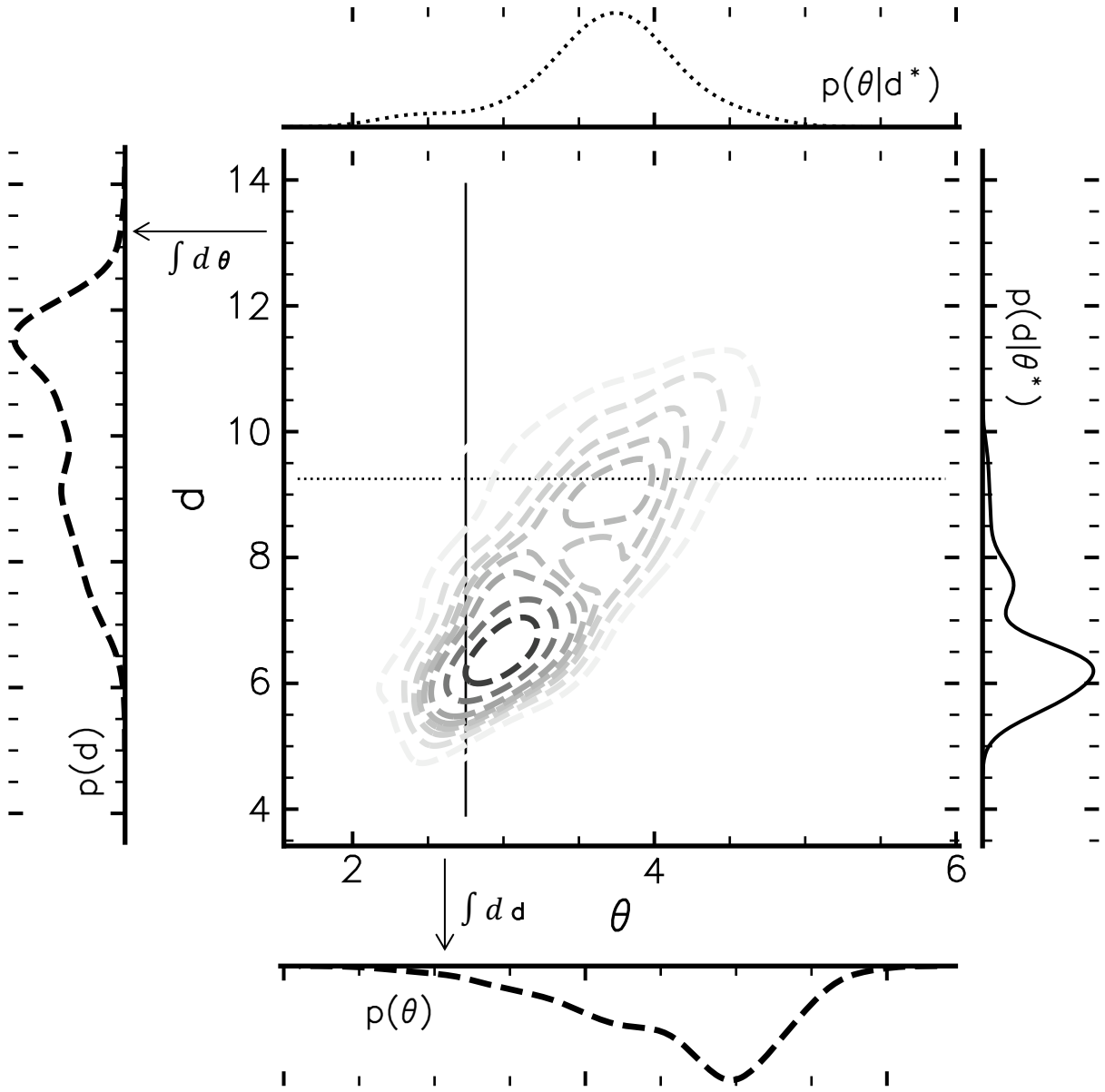


Figure 3.2: "All of statistics in a single plot". All components of Bayes' theorem can be derived from the joint distribution $p(\theta, \mathbf{d})$. The posterior distribution $p(\theta|\mathbf{d}^*)$ is obtained by slicing through the joint at fixed $\mathbf{d}$ (dotted horizontal line), and the likelihood (sampling) distribution likewise obtained for a slice at θ^* (solid vertical line). Integrating over parameters (and model) yields the evidence, or marginal likelihood $p(\mathbf{d})$, while marginalising over data returns the prior $p(\theta)$.

3.1.3 The Evidence

The last ingredient in our factorised joint distribution is the normalising constant denominator. The evidence (or "marginal likelihood") $p(\mathbf{d})$, is best thought of as the distribution of data over parameter

space with respect to our choice of model:

$$p(\mathbf{d}|M) = \int p(\mathbf{d}|\theta, M)p(\theta|M)d\theta. \tag{3.3}$$

The name "evidence" arises from this term's use in model comparison. Say we had a different model M_2 which produced another cloud of data with respect to some other parameters ϕ such that $M_2 : \phi, \eta \mapsto \mathbf{d}$. Then we could assess which model the observed data "prefers" by marginalising over each model's parameterisations and calculating the ratios

$$\frac{p(M_1|\mathbf{d})}{p(M_2|\mathbf{d})} = \frac{p(M_1)p(\mathbf{d}|M_1)}{p(M_2)p(\mathbf{d}|M_2)} \tag{3.4}$$

which relates the *posterior odds* (left) to the *prior odds* of the probability of each model times the *Bayes' factor* $B_{12} \equiv \frac{p(\mathbf{d}|M_1)}{p(\mathbf{d}|M_2)}$. A larger Bayes' factor (> 1) expresses a preference for $M = M_1$ over M_2 (if priors equal), and moreover provides a concrete prescription for *Occam's Razor*: more complicated models should be penalised unless the data prefer them. Let us assume that we have performed inference for a single model M, and that the likelihood $L(\theta) = p(\mathbf{d}|\theta)$ is strongly peaked at $\theta = \hat{\theta}$. In one dimension we can approximate the integral in Eq. 3.3 using Laplace's method (MacKay, 2002) by multiplying the peak of the integrand $p(\mathbf{d}|\theta, M)p(\theta|M)$ by its width $\delta\theta$:

$$p(\mathbf{d}|M) = \int p(\mathbf{d}|\theta, M)p(\theta|M)d\theta \tag{3.5}$$

$$= \int L(\theta)p(\theta|M)d\theta \tag{3.6}$$

$$\approx L(\hat{\theta}) \times p(\hat{\theta}|M)\delta\theta \tag{3.7}$$

$$\text{evidence} = \text{best fit likelihood} \times \text{Occam's factor} \tag{3.8}$$

Occam's factor is comprised of the quantity $\delta\theta$, which is the posterior uncertainty about θ, and the prior probability evaluated at $\hat{\theta}$. If the prior is assumed to be very wide (approaching uniform), as illustrated in Fig. 3.3, we can approximate the prior by the reciprocal of its width: $p(\hat{\theta}|M) \approx \frac{1}{\Delta\theta}$. Then all together, we can write

$$\text{Occam's factor} = \frac{\delta\theta}{\Delta\theta}, \tag{3.9}$$

which in words reads as the ratio of evaluated likelihood parameter volume for model M to the total volume of the prior. This could also be interpreted as the factor by which the prior shrinks to form the posterior once the data have been analysed (MacKay, 2002). The evidence for highly complex

models with large numbers of parameters can be penalised by both large $\Delta\theta$ and need for fine-tuning. The model with the highest evidence balances a minimisation of the complexity measure and the data misfit. This can be readily seen using a nested model, adapted from Trotta (2017). Consider

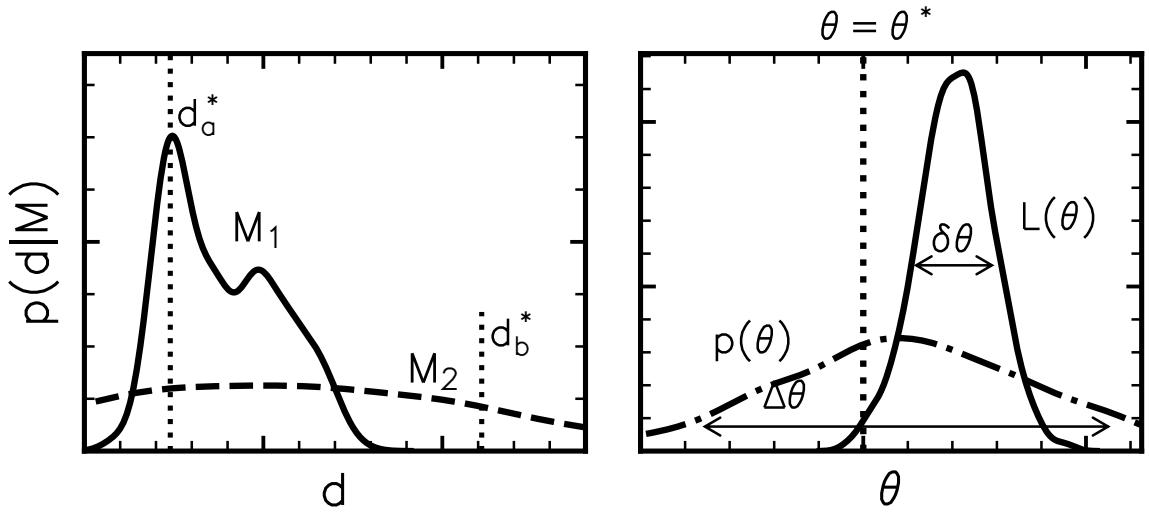


Figure 3.3: Cartoon model comparison for a nested model. *left*: a less complicated M_1 is preferred for observation a, while additional complexity from M_2 is required by observation b. *right*: Illustration of a nested model where M_2 reduces to M_1 for $\theta = \theta^* = 0$. The evidence provides an automatic Occam's Razor term $\delta\theta/\Delta\theta$, which balances likelihood fit with model complexity.

a model M_1 which is less complex (no free parameters) and a model M_2 which is more descriptive with free parameter θ, which reduces to M_1 for a critical value of the parameter $\theta = \theta^* = 0$. M_2 is evaluated with a posterior width of $\delta\theta$ over a Gaussian prior over θ of width $\Delta\theta$. We display a cartoon of these two models in data space in the left panel in Fig. 3.3. The less descriptive model M_1 might suit an observation d_a^*, but M_2 might better fit another observation d_b^*. What model does the data prefer ? First we can consolidate the "significance" of measuring a non-critical $\hat{\theta}$ via the term $\lambda = \frac{\hat{\theta}-\theta^*}{\delta\theta}$. The Bayes factor B_{12} is then

$$B_{12} = \sqrt{1 + (\delta\theta/\Delta\theta)^2}\, \exp\left[-\frac{\lambda^2}{2(1 + (\delta\theta/\Delta\theta)^2)}\right], \tag{3.10}$$

which yields a logarithm of

$$\ln B_{12} \approx \ln \frac{\Delta\theta}{\delta\theta} - \frac{\lambda^2}{2}. \tag{3.11}$$

This expression gives us another look into Occam's Razor for model comparison. In this Gaussian approximation the first term is just Occam's factor from before which contains the information gain over M_1, and the second term is always negative and favours the more complicated model if the measurement is made far from θ^* with high precision (small $\delta\theta$). Bayesian evidence calculation takes

into account both model fit, as well as the natural complexity of the models under investigation. In practice, exact evidence calculation with traditional MCMC methods can be cumbersome, but advances in sampling methods (Feroz et al., 2009) and implicit inference (Jeffrey & Wandelt, 2024) have made model comparison more tractable. We will cover these in Section 5.1.5.

It is worth noting here that we have investigated model complexity with respect to *parameters of interest*; in the case of cosmology, an extra (nested) parameter in a physical model might allow for more exotic behaviour of a quantity, such as evolving Dark Energy (Chevallier & Polarski, 2001; Linder & Jenkins, 2003). To evaluate a given "physical" model, additional hyperparameters (MCMC tuning, neural network weights) might need to be specified, but these remained fixed at inference– they do not enter the likelihood term used to compute contributions to the Bayes factor.

CHAPTER 4

STATISTICS AND DATA COMPRESSION

So far we've covered the ingredients of Bayes' theorem without specifying an explicit *form* for any of its constituent distributions. The accept-reject scheme we illustrated works wonderfully (and is exact; Prangle et al. (2014)) when the joint distribution is just one parameter and one data dimension. However, defining the distance metric $\varrho(d, d^*)$ in high dimensions quickly becomes intractable–the space of parameters and data grows exponentially and becomes difficult to cover with simulations (Kitagawa, 1996; Sisson et al., 2018).

This requires us to consider a lower-dimensional representation of the data via a *compression function* to lower-dimensional *summary statistics* $t(\mathbf{d})$ via the deterministic mapping $f : \mathbf{d} \mapsto t$. When a deterministic mapping is incorporated into the forward model, our posterior and likelihood are then defined in terms of the output statistic $p(\theta|t) \propto p(t|\theta)p(\theta)$.

But how do we choose this function, and how can we measure information captured by the output statistics ?

4.1 The Fisher Information & Local Compression

We need a way to measure the *sensitivity* of our likelihood to the parameters. For what follows, it is convenient to consider the log-likelihood $\mathcal{L} = \ln p(\mathbf{d}|\boldsymbol{\theta})$ for some n_p parameters. We can study the behaviour of this function by looking at its Taylor expansion at a fiducial $\boldsymbol{\theta}_*$: (where a function at the fiducial $g_* \equiv g(\boldsymbol{\theta} = \boldsymbol{\theta}*)$):

$$\mathcal{L} = \mathcal{L}_* + \delta\boldsymbol{\theta}^T \nabla\mathcal{L}_* - \frac{1}{2}\delta\boldsymbol{\theta}^T \mathbf{J}_* \delta\boldsymbol{\theta} \tag{4.1}$$

Figure 4.1: High Fisher Information (*left*) indicates less variance with respect to parameters; low Fisher information (*right*) means more uncertainty in θ from data.

where $\mathbf{J} = -\nabla\nabla^T\mathcal{L}$ is the observed information matrix. If we take the expectation value of this function at $\boldsymbol{\theta}_*$ over many data realisations, we obtain the *Fisher Information matrix*:

$$\mathbf{F}_{\alpha\beta} = \int d\mathbf{d}\; p(\mathbf{d}|\boldsymbol{\theta}) \frac{\partial\mathcal{L}(\mathbf{d}|\boldsymbol{\theta})}{\partial\theta_\alpha} \frac{\partial\mathcal{L}(\mathbf{d}|\boldsymbol{\theta})}{\partial\theta_\beta} \tag{4.2}$$

which can be written as

$$\mathbf{F}_{\alpha\beta} = -\left\langle \frac{\partial^2\mathcal{L}}{\partial\theta_\alpha\partial\theta_\beta} \right\rangle \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_*}. \tag{4.3}$$

The Fisher matrix is the curvature of the log-likelihood and is in general a function of the parameters. The sharper the peak of an informative log-likelihood function $\mathcal{L}$, the better θ is known from the data. We can use the Fisher matrix as a metric to grade downstream statistics of the data. We define *the information inequality* to quantify how informative a mapping is (Lehmann & Casella, 1998):

$$\mathrm{Var}_{\boldsymbol{\theta}}\left[t_\alpha\right] \geq \left(\mathbf{A}^T\mathbf{F}^{-1}\mathbf{A}\right), \tag{4.4}$$

where $\mathbf{A} = \nabla\mathbb{E}_{\boldsymbol{\theta}}\left[\mathbf{t}^T\right]$ and we write the Fisher as $\mathbf{F} = \mathbb{E}_{\boldsymbol{\theta}}\left[\nabla\mathcal{L}\nabla^T\mathcal{L}\right]$, under mild regulatory conditions (Alsing & Wandelt, 2018). In the case where the compressed numbers $\mathbf{t}$ are unbiased estimators of the parameters, $\mathbb{E}_{\boldsymbol{\theta}}\left[\mathbf{t}\right] = \boldsymbol{\theta}$, $\mathbf{A} = \mathbb{I}$ and the information inequality (4.4) reduces to the Cramér-Rao bound (Cramér, 1946):

$$\mathrm{Var}_{\boldsymbol{\theta}}\left[t_\alpha\right] \geq \mathbf{F}^{-1}_{\alpha\alpha}. \tag{4.5}$$

Maximising the Fisher information over parameter space decreases the variance on the estimates of the quantities of interest $\boldsymbol{\theta}$. Alsing & Wandelt (2018) show that knowing the *score function* of the data, $\mathbf{t} = \nabla\mathcal{L}$ provides a natural compression from $\mathbb{R}^N$ to $\mathbb{R}^{n_{\mathrm{params}}}$ and saturates the lower bound of (4.4) around a fiducial point, $\boldsymbol{\theta}_*$. We reproduce the proof here since it illustrates the concept of *sufficient statistics*. From Eq. 4.1, we see that to linear order in $\boldsymbol{\theta}$, the data $\mathbf{d}$ couples to the

parameters through the score function $\mathbf{t} \in \mathbb{R}^{n_p}$. We can show that $\mathbf{t}$ saturates the information inequality via

$$\mathrm{Cov}_{\boldsymbol{\theta}}\left[\mathbf{t},\, \mathbf{t}\right] = \mathbb{E}_{\boldsymbol{\theta}}\left[\nabla\mathcal{L}_*\nabla_*^T\right] = \mathbf{F}_*, \tag{4.6}$$

where we have used the fact that $\mathbb{E}_{\boldsymbol{\theta}}\left[\nabla\mathcal{L}_*\right] = 0$. From this we observe that the covariance of the score function is the Fisher matrix. Using the fact that

$$\mathbf{A} = \nabla\mathbb{E}_{\boldsymbol{\theta}}\left[\nabla^T\mathcal{L}\right] = \mathbb{E}_{\boldsymbol{\theta}}\left[\nabla\nabla^T\mathcal{L}\right] = -\mathbf{F}_*, \tag{4.7}$$

the right-hand side of the information inequality becomes $\mathbf{A}_*^T\mathbf{F}_*^{-1}\mathbf{A}_* = \mathbf{F}_*$, which shows that the score statistics $\mathbf{t}$ saturate the information inequality. Within this formalism, no statistics can provide more (Fisher) information about the parameters $\boldsymbol{\theta}$. We can relate this information saturation to an optimal, quasi maximum-likelihood estimator whose covariance is equal to the inverse Fisher information from the above derivation. Maximising the Taylor expansion (4.1) with respect to the parameters yields

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_* + \mathbf{J}_*^{-1}\nabla\mathcal{L}_* \tag{4.8}$$

where both the score $\mathbf{t}_* = \nabla\mathcal{L}_*$ and the observed information $\mathbf{J}_*^{-1}$ depend on the observed data. In practice, we can exchange $\mathbf{J}$ with its expectation value, the Fisher information: $\mathbf{F}_* \equiv \mathbb{E}_{\boldsymbol{\theta}}\left[\mathbf{J}_*\right]$, which yields

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_* + \mathbf{F}_*^{-1}\nabla\mathcal{L}_*. \tag{4.9}$$

Making this replacement means the MLE estimator only depends on the data through the score function statistics $\mathbf{t} = \nabla\mathcal{L}_*$. The covariance of the MLE estimator (at the expansion point $\boldsymbol{\theta}_*$) is then:

$$\mathrm{Cov}_{\boldsymbol{\theta}_*}\left[\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\theta}}\right] = \mathbf{F}_*^{-1}\mathbb{E}_{\boldsymbol{\theta}_*}\left[\nabla\mathcal{L}_*\nabla^T\mathcal{L}_*\right]\mathbf{F}_*^{-1} = \mathbf{F}_*^{-1}, \tag{4.10}$$

where $\mathbb{E}_{\boldsymbol{\theta}_*}\left[\nabla\mathcal{L}_*\nabla^T\mathcal{L}_*\right] \equiv \mathbf{F}_*$. Hence the covariance of the MLE is equal to the Fisher information matrix at $\boldsymbol{\theta}_*$ and the Cramér-Rao bound is saturated at our fiducial point.

In situations where the fiducial point may be far away from the value of the parameters preferred by the data, Eq. 4.9 can be applied iteratively via the Fisher-scoring method to better approximate point estimates for the parameters

$$\hat{\boldsymbol{\theta}}_{k+1} = \hat{\boldsymbol{\theta}}_k + \mathbf{F}_k^{-1}\nabla\mathcal{L}_k. \tag{4.11}$$

Each iteration requires a new estimate of the Fisher information and score function. As $k \to \infty$, Eq. 4.11 converges to the maximum-likelihood estimator and is asymptotically unbiased, with an asymptotically-Gaussian sampling distribution with covariance equal to the Fisher information evaluated at the maximum-likelihood point (Alsing & Wandelt, 2018). Compressing Gaussian data in this way reduces to the MOPED compression algorithm (Heavens et al., 2000).

4.2 Generic Compression

The ultimate goal of an informative compression is to retain as much information as possible about the parameters. We can extend our previous *local* Fisher information criterion to one that grades a compression function *over parameter space*. This requires incorporating the prior into our information metric. We adopt the definitions presented in Hoffmann & Onnela (2023). A fixed, *Bayes sufficient* statistic t_{suff} with finite dimensions saturates the same posterior information as the data:

$$p(\theta|t_{\text{suff}}(\mathbf{d})) = p(\theta|\mathbf{d}), \tag{4.12}$$

for any prior (Sisson et al., 2018) and all $\mathbf{d}$. Exact sufficient statistics only exist for exponential-family likelihoods, as shown by Koopman (1936). In practice for arbitrarily-shaped likelihoods, we need to shift from the concept of sufficiency to *lossless* statistics

$$p(\theta|t_{\text{lossless}}(\mathbf{d})) = p(\theta|\mathbf{d}) \tag{4.13}$$

for all θ, possible data $\mathbf{d}$ of the same sample size, and a given prior $p(\theta)$. Lossless statistics ($t \in \mathcal{L}$) always exist–for example, using the entire data vector as the statistic is itself lossless, as in our one-dimensional accept-reject case–but are not always tractable or easy to use. If we want to work with a function that compresses the data informatively, we must relax our criteria further. *Optimal statistics* $t \in \mathcal{O}$ are the output of a function of the data that minimises a non-negative loss functional over a space of possible statistics $t \in \mathcal{T}$ which measures the distance between the full-data posterior and the summary-level posterior:

$$t_{\text{opt}} = \text{argmin}_{t\in\mathcal{T}} L\left[p(\theta|\mathbf{d}), p(\theta|t(\mathbf{d})\right]. \tag{4.14}$$

By construction, L reaches zero if and only if the function $t = f(\mathbf{d})$ is lossless. We then have a

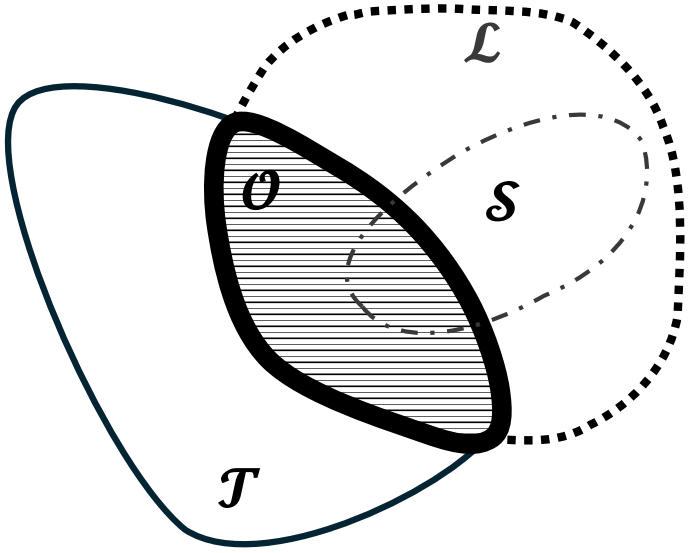


Figure 4.2: Relationship between types of summaries and those available to the practitioner, $\mathcal{T}$. Sufficient statistics $\mathcal{S}$ are a subset of lossless statistics, but sufficient statistics only exist for exponential family likelihoods. Optimal statistics are a subset of available functions of the data: $\mathcal{O} \subset \mathcal{T}$; they are functions that minimise a loss functional for a given dataset. The intersection $\mathcal{T} \cap \mathcal{O} \cap \mathcal{L}$ is not empty only if Eq 4.14 evaluates to zero for a set of statistics.

dependency between the classes of statistic illustrated in Fig. 4.2. Sufficient statistics are a subset of lossless statistics (which only exist for exponential family likelihoods). Optimal statistics are only lossless if Eq 4.14 evaluates to zero, which means that the intersection of the space of optimal statistics under consideration can be null such that $\mathcal{O} \cap \mathcal{L} = \varnothing$, for example, if a compression of the data is restricted to parametric transformations and guaranteed to omit informative aspects of the data. When considering the *optimisation* of (neural) summaries, $\mathcal{T}$ can be thought of as the space of compressed statistics available for a given dataset *and* a given neural architecture (family of nonlinear functions). In practice, limited simulations and finite networks can inhibit finding truly lossless, compressed summaries, so practitioners must instead make use of optimal statistics obtained for the chosen setup.

4.3 Information-Theoretic Objectives

How can a random variable be measured in a continuous fashion over parameter space ? We have already encountered a probability metric in the form of the Fisher information matrix, which we used to grade how well a parameter value θ can be known at a local point. Another useful quantity

is the differential *entropy* of a continuous random variable $X \sim f(X)$:

$$h(X) = -\int_{x \in \mathcal{X}} f(x) \log f(x) dx = \mathbb{E}[-\log f(X)], \tag{4.15}$$

which measures how well the random variable X is known in bits or nats, depending on the base of the logarithm (2 or e) used.

4.3.1 Expected Posterior Entropy Minimisation

We have established classes of possible summaries with respect to Bayes sufficiency. Hoffmann & Onnela (2023) show that minimising the Expected Posterior Entropy (EPE) unifies information-theoretic approaches to obtaining optimal, if not lossless summaries for arbitrary data. The posterior entropy for summaries $t(\mathbf{d})$ can be written

$$h[p(\theta|t(\mathbf{d})] = -\int d\theta \; p(\theta|t(\mathbf{d})) \log p(\theta|t(\mathbf{d})). \tag{4.16}$$

The expectation can then be taken with respect to the data generated from the forward model

$$\mathcal{H} \equiv \mathbb{E}_{\mathbf{d} \sim p(\mathbf{d})} \left[h[p(\theta|t(\mathbf{d})] \right] = -\int\int d\mathbf{d} \; d\theta \; p(\mathbf{d}) p(\theta|t(\mathbf{d})) \log p(\theta|t(\mathbf{d})) \tag{4.17}$$

where $p(\mathbf{d})$ is the marginal likelihood. Assuming a deterministic transformation from data to summaries and absorbing the Jacobian term into the joint distribution $p(\theta, t)$, the variable of integration can be changed to t to yield:

$$\mathcal{H} = -\int\int dt \; d\theta \; p(t, \theta) \log p(\theta|t). \tag{4.18}$$

If we have access to some estimate of the conditional density $\hat{p}(\theta|t(\mathbf{d}))$, the EPE can be approximated using a Monte Carlo estimator

$$\hat{\mathcal{H}} = -\frac{1}{m} \sum_{i=1}^{m} \log \hat{p}(\theta_i|t(\mathbf{d}_i)), \tag{4.19}$$

where $(\theta_i, \mathbf{d}_i)$ are joint samples from $p(\theta, \mathbf{d})$. This estimator provides a tractable optimisation criterion for finding both optimal summaries and performing density estimation, and serves as a link between lossless Bayesian statistics, other information-theoretic objective functionals, as well as common supervised machine learning loss functions.

4.3.2 The Mutual Information

The EPE minimisation can also be shown to maximise the mutual information between summaries and parameters θ. If we rewrite $p(\theta|t) = p(t,\theta)/p(t)$, Eq 4.18 reduces to the mutual information between summaries and parameters

$$I(\theta;t) = \int\int dt\ d\theta\ p(t,\theta)\log\left(\frac{p(\theta,t)}{p(\theta)p(t)}\right). \tag{4.20}$$

The mutual information can be more concisely be written in terms of differential entropy:

$$I(\theta;t) = h(\theta) - h(\theta|t(\mathbf{d})) \tag{4.21}$$

When the summary t is being optimised, (e.g. in a neural summary scheme as $t(\mathbf{d};\mathbf{w})$ Jeffrey et al. (2020); Chen et al. (2021b)) the prior entropy becomes a constant and the objective functional is just the EPE.

4.3.3 Maximising the Fisher Information

We can relate the expected posterior entropy minimisation to maximising the Fisher information in the large-sample limit $\lim_{n\to\infty}$ at a point in parameter space. Under certain regulatory conditions (Vaart, 1998), the Bernstein–von Mises theorem states that the posterior approaches a multivariate normal distribution parameterised by the Fisher information:

$$p(\theta|t) \approx \mathcal{N}(\theta_0, F^{-1}(\theta_0)), \tag{4.22}$$

where θ_0 is the true parameter that generated the summaries $t(\mathbf{d}(\theta_0))$. The Fisher information here is

$$F_{ij}(\theta_0) = \mathbb{E}_{\mathbf{d}\sim p(\mathbf{d})}\left[\left(\frac{\partial}{\partial\theta}\log p(t(\mathbf{d})|\theta)\right)\left(\frac{\partial}{\partial\theta_j}\log p(t(\mathbf{d})|\theta)\right)\right]_{\theta=\theta_0}, \tag{4.23}$$

with posterior entropy

$$\lim_{n\to\infty} H(p(\theta|t)) = -\frac{1}{2}\log\det F(\theta_0) + \text{constant}. \tag{4.24}$$

To form the EPE, we can take the expectation of H over the prior (Hoffmann & Onnela, 2023):

$$\lim_{n\to\infty} \mathcal{H} = -\frac{1}{2}\int d\theta_0 p(\theta_0) \log\det F(\theta_0) + \text{constant}. \tag{4.25}$$

Notice here that the expectation is not taken over summaries, since the Fisher information is taken as the expectation over many data-summary realisations. As the sample size increases, the EPE is dominated by the Fisher, and the effect of the prior decreases. Maximising the Fisher information is therefore the same as minimising Eq 4.18 in the large-sample limit at a point in parameter space. We will discuss explicit parameterisations of Fisher information maximisation in the form of Information Maximising Neural Networks in Section 4.4.

4.3.4 Common Machine Learning Objectives

The EPE minimisation can also be related to other common machine learning objectives in the literature. In classification or next-token prediction schemes, the objective parameters θ become one-hot encoded labels; $\theta_j = 1$ if the function of the data $t(\mathbf{d})$ corresponds to the class or token probability; $\theta_j = 0$ if not. These models often use a negative binary cross-entropy (BCE) loss formalism which is identical to the log-posterior over label classes:

$$\log p(\theta|t(\mathbf{d})) = \sum_{j=1}^{n_{\text{classes}}} \theta_j \log p(\theta_j = 1|t). \tag{4.26}$$

The Mutual Information (Eq. 4.20) also provides a link to *unsupervised* machine learning methods. Oord et al. (2019) show that the contrastive loss formalism InfoNCE provides a lower bound for maximising the mutual information between two random variables (vectors), (translated here for our objective) $\mathbf{d}$ and θ:

$$\mathcal{L} = -\mathbb{E}\left[\log \frac{f_k(\mathbf{d}_{s+k}, \theta_s)}{\sum_{\mathbf{d}_j \in \mathcal{X}} f_k(\mathbf{d}_j, \theta_s)}\right]. \tag{4.27}$$

This criterion requires N samples $X = \{\mathbf{d}_1, \ldots \mathbf{d}_N\}$, with one "positive" example from the likelihood generative model $p(\mathbf{d}_{s+k}|\theta_s)$, and one "negative" example from $p(\mathbf{d}_{s+k})$, where the index $s+k$ refers to a data organised in a sequence-completion task indexed by position s and step size k.

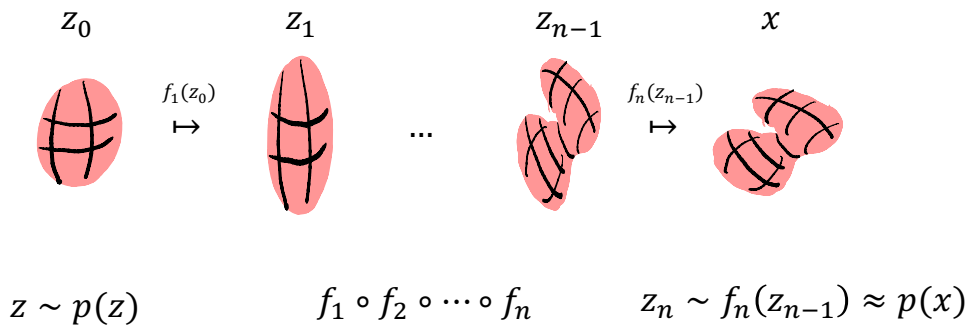


Figure 4.3: Normalising flows transform a known distribution $p(z)$ (usually Gaussian) to an unknown distribution $p(x)$ via learnable, invertible transformations that preserve probability mass, like shaping a lump of clay or chewing gum.

4.3.5 Density Estimation Models

So far, we have ignored the form of the surrogate $q(\theta|t)$ used to parameterise our information-theoretic objective. Density estimation models tune kernels or neural weights to learn the parameters of a distribution (mixture models) or a transformation from a known distribution to the objective (flow-based models). Mixture models like Mixture Density Networks (MDNs) (Bishop, 1994) are simple parametric functions that seek to learn the means, standard deviations, and mixture components of an ensemble of k Gaussian distributions as a function of data or summaries to approximate the conditional distribution $p(\theta|t(\mathbf{d})$ by minimising the mixture's known EPE:

$$\ell = -\log\left[\sum_i^k \frac{\exp\alpha_i(t,\mathbf{w})}{\sum_j^k \exp\alpha_j} \frac{1}{\sigma_i(t,\mathbf{w})\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\theta-\mu_i(t,\mathbf{w})}{\sigma_i(t,\mathbf{w})}\right)^2\right)\right], \tag{4.28}$$

where we have been explicit about the mixture parameter dependence on the (neural) weights $\mathbf{w}$.

Normalising flows (Papamakarios et al., 2017; Kingma & Dhariwal, 2018; Dinh et al., 2016), illustrated in Fig 4.3, take a different approach to learning densities: they instead map data x to latent variables z, drawn from a known (usually Gaussian) distribution, through a series of n invertible transformations $f = f_1 \circ f_2 \circ \cdots \circ f_n$ such that $x = f(z)$. These transformation can be thought of as successive (nonlinearly-parameterised) shifts and scaling of the original probability mass, much like shaping clay or chewing gum. The probability density of the data, $p(x)$ can be evaluated using the

change of variables formula:

$$p(x) = p(f^{-1}(x)) \left| \det \left(\frac{\partial f^{-1}(x)}{\partial x} \right) \right| \tag{4.29}$$

$$= p(f^{-1}(x)) \prod_{i=1}^{n} \left| \det \left(\frac{\partial f_i^{-1}(x)}{\partial x} \right) \right|. \tag{4.30}$$

The transformations are usually parameterised with neural networks, and optimal weights obtained by minimising $-\frac{1}{N}\sum_j^N \log q(x_j)$ over batches from a training set $\{x_j\}$. To sample from a trained model, latent values are drawn $z \sim p(z)$ and transformed via $x^s = f(z) \approx p(x)$. To adapt this scheme for conditional posterior estimation, the invertible function f must also be a deterministic function of the summary, $p(\theta|t) \approx f(z|t)$, which is usually achieved by passing the conditional input t to each invertible f_i (e.g. Winkler et al., 2023), or by factoring the objective probability density in an autoregressive fashion such that the Jacobian of the learned transformation is not singular. Masked Autoregressive Flows (MAFs; Papamakarios et al. (2017)) take this approach by factorising the objective probability density into 1D conditional distributions via the chain rule:

$$p(\theta|t) = \prod_{i=1}^{\dim(\theta)} p(\theta_i|\theta_{1:i-1}, \mathbf{t}), \tag{4.31}$$

where the conditional input $(\theta_{1:i}, \mathbf{t})$ is fed to a neural network that parameterises a series of 1D Gaussian distributions with careful masking of inputs to ensure the autoregressive property of Eq 4.31 is preserved. Each dimension of the target distribution is paramterised as a Gaussian with $u_i \sim \mathcal{N}(\mu_i(\theta_{1:i}, \mathbf{t}; \mathbf{w}), \sigma_i((\theta_{1:i}, \mathbf{t}; \mathbf{w})$. Stacking many such transformations and inserting into Eq 4.29 yields

$$p(\theta|t; \mathbf{w}) = \mathcal{N}\left[\mathbf{u}(\theta, \mathbf{t}; \mathbf{w}|(\mathbf{0}, \mathbf{I})\right] \times \prod_{j=1}^{n} \prod_{i=1}^{\dim(\theta)} \sigma_i^n(\theta, \mathbf{t}; \mathbf{w}), \tag{4.32}$$

where we have explicitly denoted vectors in boldface. Stacking more neural layers can improve expressivity and fidelity of the learned distribution.

Another popular generative model paradigm for learning unknown distributions is score-based diffusion (Hyvärinen, 2005; Song et al., 2021). Unlike normalising flows which rely on invertible layers to transform a known distribution, diffusion models rely on incrementally adding small noise perturbations to a target distribution and solving a reversed stochastic differential equation to sample from a base (white noise) distribution. To train these models, a score-matching objective must first be optimised (Hyvärinen, 2005). For a detailed treatment, see Yang et al. (2024).

4.4 Information Maximising Neural Networks

IMNNs (Charnock et al., 2020; Makinen et al., 2021) are neural networks that perform data compression and optimised to compute the Fisher information of a dataset at a fiducial point. Once optimised, a fixed IMNN can be used as an informative compression function for data generated over a prior of parameters. An advantage to learning a local compression is that simulations only need to be run at $\theta = \theta_{\rm fid}$, with an estimation for how the data change with respect to small changes in the parameters near $\theta_{\rm fid}$. The Fisher information objective can be written by enforcing the form of a Gaussian likelihood with parameter-independent covariance for the summaries of the data $t(\mathbf{d})$ obtained from a neural network with weights w $f_w : \mathbf{d} \mapsto t$ at a fiducial point $\theta = \theta_{\rm fid}$:

$$-2 \ln p(t|\mathbf{d}) = (t - \mu)^T C^{-1} (t - \mu), \tag{4.33}$$

from which we can immediately write down the Fisher to be

$$F_{ij} = \mu^T_{,\theta_i} C^{-1} \mu_{,\theta_j}, \tag{4.34}$$

where $y_{,\alpha} \equiv \partial y / \partial \theta_\alpha$,

$$\mu = \int t(\mathbf{d}) p(\mathbf{d}|\theta) \; d\mathbf{d} \approx \frac{1}{n_d} \sum_{i=1}^{n_d} t(\mathbf{d}_i), \tag{4.35}$$

and

$$C_{\alpha\beta} = \int (t - \mu)_\alpha (t - \mu)_\beta \; p(\mathbf{d}|\theta) \; d\mathbf{d} \approx \frac{1}{n_s - 1} \sum_{i=1}^{n_s} (t_i - \mu)_\alpha (t_i - \mu)_\beta. \tag{4.36}$$

is evaluated over data realisations at θ^*, and henceforth the dependence of t on $\mathbf{d}$ is left implicit.

One way of computing the derivatives of the summary means with respect to the parameters is to define a finite difference gradient dataset by altering simulation fiducial values by a small amount $t_i^\pm = t(\mathbf{d}_i^\pm)$, yielding

$$\left(\frac{\partial \hat{\mu}_i}{\partial \theta_\alpha} \right)^{\rm fid} \approx \frac{1}{n_s} \sum_{i=1}^{n_s} \frac{t_i^+ - t_i^-}{\theta_\alpha^+ - \theta_\alpha^-}. \tag{4.37}$$

To prevent extra information being extracted from accidental correlation in limited sized data sets, reported statistics need to be computed on a validation set of simulations, which is unlikely to share the same accidental correlations as the fixed training set. An alternative explored in Makinen et al. (2021) is to calculate the adjoint gradient of the simulations as well as the derivatives of the network

parameters with respect to the simulations:

$$\mu_{,\alpha} = \frac{1}{n_d}\sum_{i=1}^{n_d}\left(\frac{\partial t}{\partial\theta_\alpha}\right)_i = \frac{1}{n_d}\sum_{i=1}^{n_d}\frac{\partial t_i}{\partial \mathbf{d}_k}\frac{\partial \mathbf{d}_k}{\partial\theta_\alpha}. \tag{4.38}$$

For differentiable simulations and networks, this derivative calculation is exact. The network is trained to maximise

$$\mathcal{L} = -\ln\det F + g(C), \tag{4.39}$$

where g is a function to penalise large covariance matrix values, since the Fisher is invariant to nonsingular linear transformations of the network summaries.

Computational Considerations. One bottleneck to the IMNN loss function is the number of simulations needed to paramterise C and its inverse C^{-1}. This can be inefficient, especially when tied with backpropagation through Eq 4.39 over copies of a large network to update network weights. Explicitly, this reads

$$\partial_w \ln|F| = \mathrm{Tr}\left(F^{-1}F_{,w}\right), \tag{4.40}$$

the last term of which can be broken up into:

$$F_{,w} = 2\left[\mu_{,\theta}^T C^{-1}\mu_{,\theta,w} - \mu_{,\theta}^T C^{-1}C_{,w}C^{-1}\mu_{,\theta}\right]. \tag{4.41}$$

What this eliminates is the need for differentiating through the full computation graph when calculating $F = \mu_{,\theta}^T C^{-1}\mu_{,\theta}$. Instead, we can restrict the neural network gradient calculation to computing the terms $\mu_{,\theta,w}$ and $C_{,w}$, which require far fewer simulations to achieve an unbiased estimate. Furthermore we can simplify the estimator for $C_{,w}$ to. If we define $\Delta f = (f-\mu)$, and $\Delta f_{,w} = (f-\mu)_{,w}$, then we have

$$C_{,w} = \frac{1}{2}\langle \Delta f_{,w} f^T + f\Delta f_{,w}^T\rangle. \tag{4.42}$$

In practice, we would use all available simulations to calculate $\mu_{,\theta}$, C^{-1}, and even the full Fisher quantities F and F^{-1}, but now the data does not have to be included in the backpropagation computation graph. For many examples this is not necessary, but is of practical use when scaling to large data calculations like halo catalogue emulation discussed in Chapter 7.

Once trained, an IMNN provides a locally-optimal compression and estimate of the Fisher information. This compression is in general dependent on the chosen neural network architecture (small, less-descriptive networks may yield a lossy compression with respect to the true Fisher information of

the underlying distribution). If the optimisation is performed far from the fiducial point $\theta_{\rm fid}^{(1)}$, target data can be fed through the IMNN and a parameter estimate made via a quasi-maximum likelihood estimates (MLE) for the parameters for a given mean-subtracted summary vector $\Delta = [\Delta\mathbf{t}, \Delta\mathbf{x}]$, adapting Eq 4.9:

$$\hat{\theta} = \theta_{\rm fid}^{(1)} + F^{-1}\mu_{,\theta_i}C^{-1}\Delta^T. \tag{4.43}$$

As shown in Makinen et al. (2021), the compression can be re-run at $\theta_{\rm fid}^{(2)} = \hat{\theta}$ if the original compression point is many-sigma away from $\hat{\theta}$, as measured by the IMNN Fisher matrix. A trained IMNN can then act as a fixed compression function for simulations generated over a prior of parameter values. Normalised summaries can be taken as $t = \hat{\theta}$.

4.4.1 The IMNN and the Implicit Score Function

What is the nature of the summaries that the IMNN computes ? Adapting a proof by Wandelt (2022) we can show that an optimised IMNN outputs the score function of the *underlying, implicit* score function of the dataset. We will show this for the one-parameter case for an IMNN optimised to output a single summary. If we take the IMNN loss function Eq. 4.39 without regularisation terms, a maximal information extraction is obtained when a summary is found such that $\ln\det F$ stops increasing:

$$\frac{\delta}{\delta t}\ln\det F = \mathrm{Tr}\left(\mathrm{F}^{-1}\frac{\delta}{\delta \mathrm{t}}\mathrm{F}\right) = 0. \tag{4.44}$$

Manipulating the left hand side, and recalling that μ and C are scalars for the 1D case we obtain

$$\frac{\delta}{\delta t}\ln\det F = \frac{\delta}{\delta t}\ln\mu_{,\theta}C^{-1}\mu_{,\theta} = 0, \tag{4.45}$$

which we break into

$$2\mu_{,\theta}^{-1}\frac{\delta}{\delta t}\mu_{,\theta} - C^{-1}\frac{\delta}{\delta t}C = 2\mu_{,\theta}^{-1}p_{,\theta}(x|\theta) - 2C^{-1}(t-\mu)p(x|\theta) = 0 \tag{4.46}$$

If we divide by $p(x|\theta)$,

$$\frac{p_{,\theta}(x|\theta)}{p(x|\theta)}\mu_{,\theta}^{-1} = C^{-1}(t-\mu), \tag{4.47}$$

and employing the log-derivative trick obtain

$$\partial_\theta \ln p(x|\theta) = \mu_{,\theta}C^{-1}(t-\mu), \tag{4.48}$$

which equates the score function of the original data likelihood to the *Gaussian* score of the IMNN output. Squaring the score and taking the expectation with respect to the unknown data likelihood $p(x|\theta)$ gives

$$\int (\partial_\theta \ln p(x|\theta))^2 \, p(x|\theta) dx = \mu_{,\theta}^2 \, C^{-2} \int (t-\mu)^2 p(x|\theta) dx = \mu_{,\theta} \, C^{-1} \mu_{,\theta} \tag{4.49}$$

The left hand side is the Fisher information of the original data set x and the right hand side the Fisher information of the output of the IMNN. The key takeaway is that for a given dataset and neural network choice, the IMNN conserves Fisher information at a fiducial point, and computes the score of the unknown, implicit data-network likelihood.

4.5 Neural Networks

Neural networks can be simply thought of as arbitrary numerical maps from an input space to an output space. The mapping is comprised of L hidden layers, each parameterized by neurons, in which a data vector $\mathbf{d} = \{d_i | i \in [1, n_\mathbf{d}]\}$ is passed via a system of weights and biases to capture data abstraction. Each layer l of a network takes a certain number of inputs and outputs one value per neuron via an activation function

$$a_i^l = \phi(v_i^l) \tag{4.50}$$

where

$$v_j^l = \sum_i w_{ji}^l a_i^{l-1} + b_j^l \tag{4.51}$$

is the layer input parameterized by weights $\mathbf{w} = w_{ji}^l$ and biases $\mathbf{b} = b_j^l$. Increasing the number of hidden layers between input and output allows for higher levels of abstraction (see e.g. Goodfellow et al., 2016). The final layer of the network is of the same dimensionality as the target space for supervised learning problems, e.g. $\mathbf{a}^L = \{a_i^L | i \in [1, n_{\rm out}]\}$. For a compression scheme, the neural mapping can be defined mathematically at each layer as

$$f^l : \mathbf{d} \mapsto \mathbf{a}^l = \phi \left(\sum_i w_{ji}^l (f^{l-1}(\mathbf{d}))_i + b_j^l \right) \tag{4.52}$$

and

$$f^0 : \mathbf{d} \mapsto \mathbf{a}^0 = \mathbf{m} \tag{4.53}$$

for the first layer. This formalizes the data input $\mathbf{d}$ as the first layer and the final compressed summaries as the output to the last layer $l = L$. It is worth noting that for (deterministic) neural transformations of the input data, the output from *any* intermediate layer in a deep network can be labelled and used as a summary statistic.

Neural Embeddings as Statistics

The intermediate "latent" spaces of neural networks are often much higher-dimensional than the target quantities of interest, and can be exploited for different downstream tasks as informative transformations of the data. To see this, consider a network f comprised of $L-1$ layers: $f : \mathbf{d} \mapsto \mathbf{x}$, where $\mathbf{x}$ is known to be Bayes-sufficient, $p(\theta|\mathbf{x}) = p(\theta|\mathbf{d})$, and $\dim(\theta) > 1$. Let us introduce an L^{th} layer g that applies a boolean mask to all but one dimension of $\mathbf{x}$; $g : \mathbf{x} \mapsto \mathbf{x}_{:1}$. From the Section 4.2, we know that we need at least as many summary numbers as we have parameters to define sufficient statistics for exponential families; $\dim(t) = \dim(\theta)$, so even if the data followed an exponential family distribution, the last layer would be *lossy* in terms of θ information capture. This example, while simple, is important because it shows that intermediate layers of a network can be more useful than the final layer. Intermediate layers are for this reason often called "embedding" layers in the literature since they can be used for other downstream tasks.

4.6 Optimisation

To *train* or *fit* a neural network to data, we need a way to adjust the weights and biases of the network to minimise some objective function to some known data-target pairs $\Lambda(f(\mathbf{d};\phi),\theta)$:

$$\phi^{\text{opt}} = \text{argmin}_{\phi}\left[\Lambda\left[\phi\right]\right], \tag{4.54}$$

where $\phi = [w, b]$ are the free parameters of the network. Gradient descent is one of the simplest algorithms for iterating towards a better fit, first proposed by Cauchy (1847). The first step is to

calculate the gradient of the loss function with respect to the network weights:

$$\frac{\partial \Lambda}{\partial \phi} = \begin{bmatrix} \frac{\partial \Lambda}{\partial \phi_0} \\ \frac{\partial \Lambda}{\partial \phi_1} \\ \cdots \end{bmatrix}. \tag{4.55}$$

Next, update the parameters via

$$\phi \leftarrow \phi - \eta \frac{\partial \Lambda}{\partial \phi}, \tag{4.56}$$

where η is a nonzero positive scalar called the *learning rate* which controls the magnitude of the change. The gradient determines the "uphill" direction of the loss function with respect to the weights. The negative sign in the update step indicates a small step *downhill*, the step size controlled by η. For *convex* problems like linear regression, well-defined global minima in Λ exist, and gradient descent can provably find them (Hazan, 2017; Prince, 2023). However, most neural network optimisations are nonlinear and non-convex, meaning optimisation instead prioritises finding useful *local* minima as opposed to a single global minimum over a dataset.

4.6.1 Stochastic Gradient Descent

A way to ensure more useful local minima are located is to introduce stochasticity into the estimation of the loss landscape. This is achieved through *batching* a large dataset into smaller, randomised chunks and computing the gradient descent update on a "blurrier" estimate of the global loss landscape at each iteration t through the dataset, with the idea that useful local minima in weight space will appear in many randomised realisations of the landscape. The weight update rule reads as:

$$\phi_{t+1} \leftarrow \phi_t - \eta \sum_{i \in \mathcal{B}_t} \frac{\partial \Lambda_i \left[\phi_t\right]}{\partial \phi}, \tag{4.57}$$

where $\mathcal{B}_t = \{x_i, \theta_i\}$ is the batch of examples and $\frac{\partial}{\partial \phi} \sum_i \Lambda_i = \sum_i \frac{\partial \Lambda_i}{\partial \phi}$ is the gradient of the loss computed for each member of the batch. One pass through the full dataset is denoted a training *epoch*. Stochastic Gradient Descent (SGD) (Robbins & Monro, 1951) is often employed with a learning rate schedule which (eventually) decreases the learning rate over successive epochs. Larger learning rates allow the weights to "hop" between local valleys in the loss landscape and eventually settle in deeper, more suitable minima to make smaller updates to ϕ (see Prince (2023), Chapter 6

for a detailed treatment). However, implementing SGD with too large a learning rate can often lead to oscillatory behaviour, which can evade a desired minimum (circle-marked path in the left-hand panel in Fig 4.4). Too small a learning rate, and SGD can fail to probe the loss landscape for other suitable minima (square-marked path).

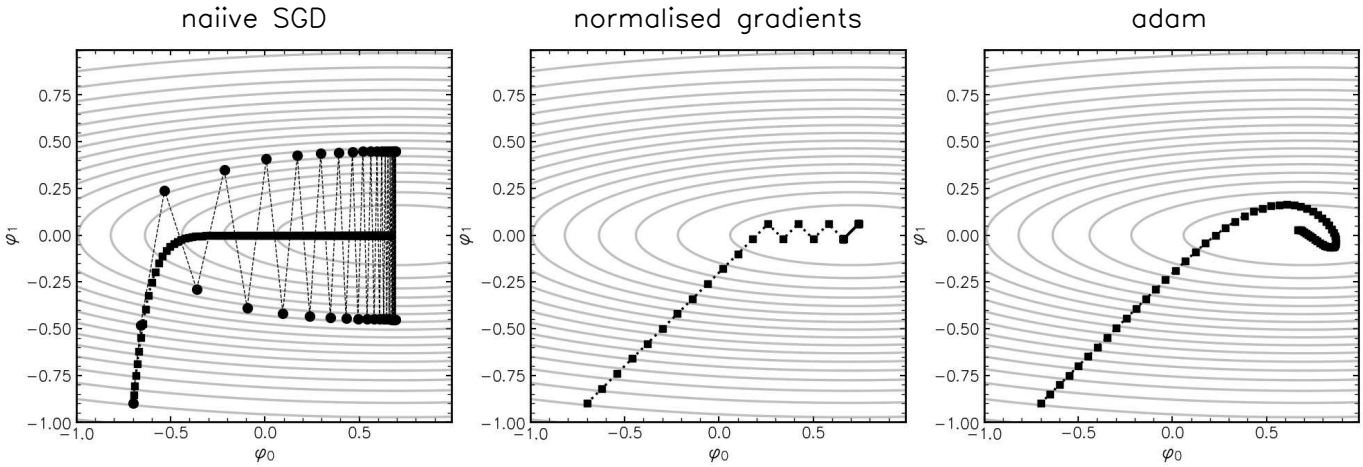


Figure 4.4: Stochastic Gradient Descent (4.57) with poor choice of learning rate can lead to trajectories that oscillate or under-explore the loss landscape (*left*). Incorporating a normalisation scheme (4.59, *centre*) and adaptive momentum to smooth trajectories (4.63, *right*) can improve optimiser performance and find more useful minima using a smaller number of steps.

4.6.2 Adam and Momentum

Higher dimensionality can exacerbate SGD's oscillatory behaviour between two or more seemingly suitable local minima. Smoothing trajectories in weight space can lead to faster convergence to suitable smaller minima. This can be accomplished by introducing a *momentum* term to the weight update step:

$$\begin{aligned} \mathbf{m}_{t+1} &\leftarrow \beta \cdot \mathbf{m}_t + (1-\beta) \sum_{i \in \mathcal{B}_t} \frac{\partial \Lambda_i \left[\phi_t\right]}{\partial \phi} \\ \phi_{t+1} &\leftarrow \phi_t - \eta \cdot \mathbf{m}_{t+1}, \end{aligned} \tag{4.58}$$

controlled by scalar parameter $\beta \in [0, 1]$ (usually kept quite close to 1.0). Now the SGD update step contains a smoothed "memory" of the previous steps taken by the gradient descent.

Normalised Gradients

Sometimes the loss can vary slower with respect to one weight dimension than the other. Adapting an example from Prince (2023), Fig 4.4 shows a constructed loss landscape for two parameters $\phi = (\varphi_0, \varphi_1)$, whose contours (steepness) varies more quickly for φ_1 than for φ_0. SGD with too large or too small a learning rate can fail to properly explore this sort of landscape (rightmost panel). Normalising gradients by the second moment of the loss function is a way "flatten" the loss landscape for the fixed learning rate step size:

$$\begin{aligned} \mathbf{m}_{t+1} &\leftarrow \frac{\partial \Lambda[\phi_t]}{\partial \phi} \\ \mathbf{v}_{t+1} &\leftarrow \left(\frac{\partial \Lambda[\phi_t]}{\partial \phi}\right)^2, \end{aligned} \tag{4.59}$$

which is then incorporated into the weight update via

$$\phi_{t+1} \leftarrow \phi_t - \eta \cdot \frac{\mathbf{m}_{t+1}}{\sqrt{\mathbf{v}_{t+1}} + \epsilon}, \tag{4.60}$$

where $\epsilon > 0$ is a very small constant to prevent division by zero. Dividing the gradient by the factor $\sqrt{\mathbf{v}_{t+1}}$ allows the optimiser to move a fixed distance η along each coordinate in the downhill direction. This adjustment is efficient for finding valleys, but will continue to oscillate unless it chances upon the exact minimum (Fig 4.4, middle panel).

Adaptive Moment Estimation (adam)

To combine both "memory" of optimiser trajectory and normalised gradients, `adam` (Kingma & Ba, 2014) incorporates momentum to both $\mathbf{m}_t$ and $\mathbf{v}_t$ estimation:

$$\begin{aligned} \mathbf{m}_{t+1} &\leftarrow \beta \cdot \mathbf{m}_t + (1-\beta)\frac{\partial \Lambda[\phi_t]}{\partial \phi} \\ \mathbf{v}_{t+1} &\leftarrow \gamma \cdot \mathbf{v}_t + (1-\gamma)\left(\frac{\partial \Lambda[\phi_t]}{\partial \phi}\right)^2. \end{aligned} \tag{4.61}$$

To prevent unwanted zeros and artefacts at the beginning of the trajectory, the two moments are modified slightly via:

$$\tilde{\mathbf{m}}_{t+1} \leftarrow \frac{\mathbf{m}_{t+1}}{1-\beta^{t+1}}, \qquad \tilde{\mathbf{v}}_{t+1} \leftarrow \frac{\mathbf{v}_{t+1}}{1-\gamma^{t+1}}, \tag{4.62}$$

where the denominators become close to one as a function of step t. The network weights are then updated using the modified values:

$$\phi_{t+1} \leftarrow \phi_t - \eta \cdot \frac{\tilde{\mathbf{m}}_{t+1}}{\sqrt{\tilde{\mathbf{v}}_{t+1}} + \epsilon}, \tag{4.63}$$

which are in practice calculated over minibatch estimates like standard SGD. Incorporating this into the scheme in Fig. 4.4 results in a smooth trajectory that effectively explores the loss landscape and settles on the desired (by construction) minimum.

For all neural optimisation problems batch size, learning rate, and data input scaling *are the most important hyperparameters to tune by the practitioner*, and are often dependent on the geometry of the data inputs, loss objective function, and neural architectures under consideration (Goodfellow et al., 2016; Prince, 2023). In practice, the choice of optimiser, be it SGD, `adam`, or another variant is itself a hyperparameter that can be problem- or architecture-dependent.

Remarkably, simple modifications to a 180-year-old algorithm and good choice of hyperparameters allows for stable training of neural networks with as many as several trillion weights (e.g. GPT-4), and allows us to propose and optimise new, information-theoretic objectives directly.

4.7 Robustness & Statistical Learning Theory

We have so far used the notion of Bayesian optimality to define summary statistics and learnable functions to obtain them from data. Neural networks have been shown to be universal function approximators (Kratsios, 2019), so in theory can be used to find functions to define optimal summarisation of data. We have assumed that the target data to be compressed into summaries comes from the same joint distribution, $p(\theta, \mathbf{d})$, used to learn and grade the summary function. But how many samples do we need to learn from, and what happens when the training data is not consistent with reality ?

To investigate this, we will cast our (neural) summary optimisation as a generic learning algorithm and show that *any* model will fail to learn a given concept *if it is blind to the constraints of a problem*, such as an unknown or changing source of noise which corrupts labels in a classification problem. We will follow a proof adapted from Schapire (2018) and Shalev-Shwartz & Ben-David (2014).

Consider a general learning algorithm, $\mathcal{A}$, that seeks to solve a classification problem. Take as input a set X of features, independently and identically distributed (iid) according to distribution D. $\mathcal{A}$ seeks to map each $x \in X$ to a binary label $Y = \{0, 1\}$. Think of X as a set of cat and dog images, and Y indicating whether or not each image is a cat. The underlying distinction, $f(x)$, between cats and dogs is called a *concept*, an element of a concept class, C; $f(x) \in C$. The output of $\mathcal{A}$ is the best fit hypothesis from a given set of hypotheses: $h \in \mathcal{H}$ that gets closest to the concept class, most of the time. Each hypothesis h is graded by its generalization error, that is, its probability of misclassifying the next example:

$$\text{err}_D(h) = p_{x \sim D}((f(x) \neq h(x)) \tag{4.64}$$

We want to minimize this generalization error, such that our algorithm successfully makes the distinction between cat and dog. "Probably Approximately Correct" (PAC) algorithm for a generic learning problem:

Definition 4.7.1. A concept class $f(x)$ is considered PAC-learnable by $\mathcal{H}$ if there exists a learning algorithm $\mathcal{A}$ such that, for all $f(x) \in C$, for any distribution D, and for any error bounds $\epsilon > 0$, $\delta > 0$, $\mathcal{A}$ takes a set $\mathcal{S}$ of m random examples and produces a hypothesis $h \in \mathcal{H}$ such that

$$p(\text{err}_D(h) \leq \epsilon) \geq 1 - \delta \tag{4.65}$$

Consider a concrete example. Take $X = \mathbb{R}^2$ to be the domain and $Y = \{0, 1\}$ to be the label set of the learning problem. Let $\mathcal{H} = \{h_r, r \in \mathbb{R}_+\}$ be a set of hypotheses corresponding to all concentric circles on the plane that classify $x \in X$ as

$$h_r = \begin{cases} 1 & ||x||_2 \leq r \\ 0 & \text{otherwise} \end{cases} \tag{4.66}$$

Under the realizability assumption, there exists $h^* \in \mathcal{H}$ such that the loss between h and the true concept $f(x) = c$ is 0, meaning the learning problem above is PAC-learnable with sample complexity

$$m_{\mathcal{H}}(\epsilon, \delta) \leq \left| \frac{\lg \frac{1}{\delta}}{\epsilon} \right|$$

Proof. For a radius that can extend over $\mathbb{R}_+$, define a corresponding *probability mass* over the distribution D from which we can sample X, $\mathbf{x}_i \sim D$; with each $\mathbf{x}_i = (x_1, x_2) \in \mathbb{R}^2$. Then, we define

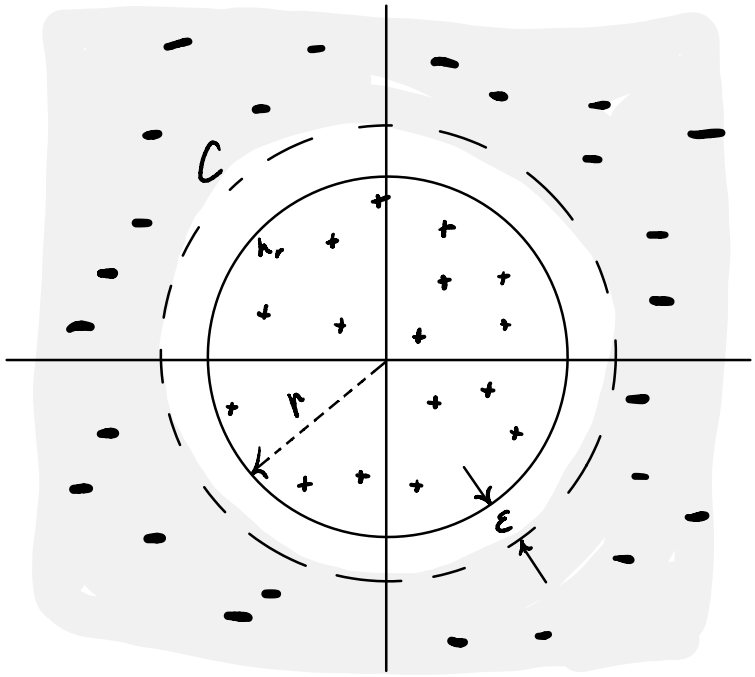


Figure 4.5: Classification problem in $\mathbb{R}^2$. The goal of algorithm $\mathcal{A}$ is to locate a hypothesis h, which best separates "+" points from "−" points. In this case, $h = r$, some radius that defines a symmetric circular boundary between the two classes. We choose c to denote the true concept boundary, which $\mathcal{A}$ seeks to approximate.

a concept class $\mathcal{C}$ along this radius r. This concept class is the "goal" of our algorithm–the radius which separates "+" points from "−" points (see Figure 4.5). We now define our algorithm $\mathcal{A}$, which sweeps over possible radii: $\mathcal{A}$: *find the maximum positive example data point at radius* r_+ *and the minimum negative example at radius* r_-*. Then choose a point between* r_+ *and* r_- *to be the hypothesis point,* h*.* We can next define two bad events:

$$b_+ : \text{when } r = h \text{ undershoots the true concept, } c \text{ by } \epsilon.$$

$$b_- : \text{when } r = h \text{ overshoots the true concept, } c \text{ by } \epsilon.$$

We see that either of these bad events (misclassifying either class of points) is *bounded* by the probability of a data point $x_i \sim D$ falling within $R_+ = r_+ - c$. We can write:

$$p(x_i \in r_+) = \epsilon$$

$$\implies p(x_i \notin r_+) = 1 - \epsilon$$

So then looking at our set of m examples, $\mathcal{S}$:

$$\begin{aligned} p(\text{no examples in } R_+) &= p((x_1 \notin R_+) \wedge (x_2 \notin R_+) \wedge \cdots \wedge (x_m \notin R_+)) \\ &= p(x_1 \notin R_+) \times \cdots \times p(x_m \notin R_+) \\ &= [p(x_1 \notin R_+)]^m \leq (1-\epsilon)^m \qquad \text{(data i.i.d. distributed)} \\ &\leq e^{-\epsilon m} \end{aligned}$$

To incorporate the probability of h undershooting c (hitting within $R_- = c - r_-$), we can define the probability of total error:

$$\begin{aligned} p(\text{err}_D(h) > \epsilon) &\leq p(b_+ \vee b_-) \\ &\leq p(b_+) + p(b_-) \leq 2e^{-\epsilon m} \end{aligned}$$

so now, to show PAC learnability, we need that

$$\begin{aligned} & p(\text{err}_D(h) > \epsilon) \leq \delta \\ & \implies 2e^{-\epsilon m} \leq \delta \\ & \implies \ln(2) - \epsilon m \leq \ln(\delta) \\ & \implies m_H(\epsilon, \delta) \leq \frac{\ln(1/\delta)}{\epsilon} \end{aligned}$$

□

Agnostic Noise. Now we turn to an *agnostic* noise tolerance situation, in which noise is present, yet the algorithm $\mathcal{A}$ is given no knowledge of the noise in the data set $\mathcal{S}$. Assume the observed function $\hat{f}(x)$ takes the following form:

$$\hat{f}(x) = \begin{cases} 1 & \text{w/ prob } \frac{\epsilon_o}{2} \\ 0 & \text{w/ prob } \frac{\epsilon_o}{2} \\ f(x) & \text{otherwise} \end{cases} \tag{4.67}$$

In other words, incoming data can be noisily mislabeled with probability $\epsilon_o/2$. Then our algorithm

$\mathcal{A}$ can only produce a hypothesis $h_{\mathcal{A}}$ such that

$$p(\text{err}_D(h_{\mathcal{A}}) \leq \frac{1}{2}\epsilon_o + \epsilon) \geq 1 - \delta$$

Proof. We know that in the absence of noise by the noise-free case that

$$p(\text{err}_D(h_{\mathcal{A}} \leq \epsilon)) \geq 1 - \delta$$

by PAC-learnability. If we adopt our example in $\mathbb{R}^2$ again, we can modify the probability of misclassification to incorporate the stochasticity of noisy data. We modify our probability of a bad event to be:

$$\begin{aligned} p(x_i \in R_+ \wedge \hat{f}(x_i) = 1) & \\ &= p(x_i \in R_+) + p(\hat{f}(x_i) = 1) \\ &= \epsilon + \frac{\epsilon_o}{2} \end{aligned}$$

Following similar logic to before, we can extend our analysis to sweep over possible hypotheses $h \in \mathcal{H}$. We label the subset of ϵ-bad hypotheses $\mathcal{B}$. Then, for a single $h_{\mathcal{A}}$:

$$\begin{aligned} p(\text{err}_D(h_{\mathcal{A}}) > \epsilon) &\leq p\Big(\{\exists h \in \mathcal{H} : h = h_{\mathcal{A}}\} \wedge \{h \text{ is } \epsilon - \text{bad}\}\Big) \\ &= p(\exists h \in \mathcal{B} : h = h_{\mathcal{A}}) \qquad \text{(consistent hypothesis)} \\ &= p(h_1 : h_1 = h_{\mathcal{A}} \vee h_2 : h_2 = h_{\mathcal{A}} \vee \dots) \\ &= p\Big(\bigvee_{h \in \mathcal{B}} h = h_{\mathcal{A}}\Big) \leq \sum_{h \in \mathcal{B}} p(h = h_{\mathcal{A}}) \qquad \text{(hypotheses i.i.d. distributed)} \\ &= \sum_{h \in \mathcal{B}} p(\{(x_1 \in R_+) \vee (f(x_1) = 1)\} \dots \{(x_1 \in R_+) \vee (f(x_1) = 1)\}) \\ &= \sum_{h \in \mathcal{B}} p(\{(x_1 \in R_+) \vee (f(x_1) = 1)\}) \times \dots \times p(\{(x_1 \in R_+) \vee (f(x_1) = 1)\}) \\ &= \sum_{h \in \mathcal{B}} (\epsilon + \frac{\epsilon_o}{2}))^m \leq \sum_{h \in \mathcal{B}} (1 - (\epsilon + \frac{\epsilon_o}{2}))^m \\ &\leq |\mathcal{B}| \left[1 - (\epsilon + \frac{\epsilon_o}{2})\right]^m \qquad \text{(modulus theorem)} \end{aligned}$$

but now we realize that the space of bad hypotheses is a subset of the full hypothesis space so that

$|\mathcal{B}| \leq |\mathcal{H}|$. We can now finish the proof:

$$\leq |\mathcal{H}|(1 - (\epsilon + \frac{\epsilon_o}{2}))^m = \delta$$
$$\implies m \leq \frac{\ln(1/\delta) + \ln|\mathcal{H}|}{(\epsilon + \frac{\epsilon_o}{2})}$$

□

We can take away a couple of things from this result. On one hand, we bound the number of examples, m, needed for PAC-learnability in the event of misclassified data. Rearranging the expression for ϵ:

$$\epsilon \geq \frac{\epsilon_o}{2} - \frac{\ln(1/\delta) + \ln|\mathcal{H}|}{m}$$

we see that our learnability margin, ϵ, is inversely related to the number of examples, m. However, without telling the algorithm anything about the noise model, the error is lower bounded by the noise, ϵ_o, even as the second term vanishes in the limit $\lim_{m\to\infty}$. What we can take away from this is that an arbitrary learner, no matter how many examples we show it, or how well it knows a concept in a non-noisy setting, cannot learn an arbitrary, unspecified noise model agnostically. It will always misclassify examples with an error greater or equal to the error guaranteed by the (unknown) noise or systematic effect. Like a perfectly-trained classical musician asked to perform a jazz improvisation, the resulting performance will likely be misinformed–he will miss the “blue notes" that distinguish major scales from blues scales, for example, if he’s not accustomed to adjusting to the changes the band plays behind him. [1]

[1] The author cites his own experiences in this avenue

CHAPTER 5

NEURAL STATISTICS & SIMULATION-BASED INFERENCE

We now have a set of information-theoretic objectives and tunable neural operators to optimise with respect to different aspects of an implicit inference problem. Implicit or Simulation-Based inference relieves us of having to specify explicit likelihoods for Bayesian analysis, and instead allows us to directly link theory (simulations) to data. We will argue at the end of this chapter that implicit Bayesian methods cast cosmology as a streamlined optimisation problem.

5.1 The Anatomy of a Neural Inference problem

Here we'll revisit our sketch of our generic implicit inference problem from Fig. 3.1. We will break the problem down into two distinct steps: compression and density estimation. Although many practitioners advocate for tying these two objectives together under one loss function, it is important to note that the eventual posterior consists of two distinct tasks for the neural networks involved.

5.1.1 Step 1: Compression Network

Using simulations from the forward model, we can specify or learn a compression function that best suits the data. If a neural compression is to be used, a set of training simulations must be available to perform backpropagation over a specified loss functional. Most effective neural networks consist of a higher-dimensional embedding of the data before downsampling features to a lower-dimensional statistic space.

It should be emphasised here that *any* layer of a deterministic neural network can be thought of as either an "embedding" or a "statistic", since both are functions of the data. Foundation models, for example, rely on unsupervised losses like Eq. 4.27 to learn generic, informative embeddings for input data that can be fine-tuned for downstream tasks (Bommasani et al., 2022). This stage of inference is also where ablation studies and feature discovery can be made from the data. As discussed in Chapter 7, data symmetries can be exploited by user and network alike to improve both information capture and physical interpretability of results.

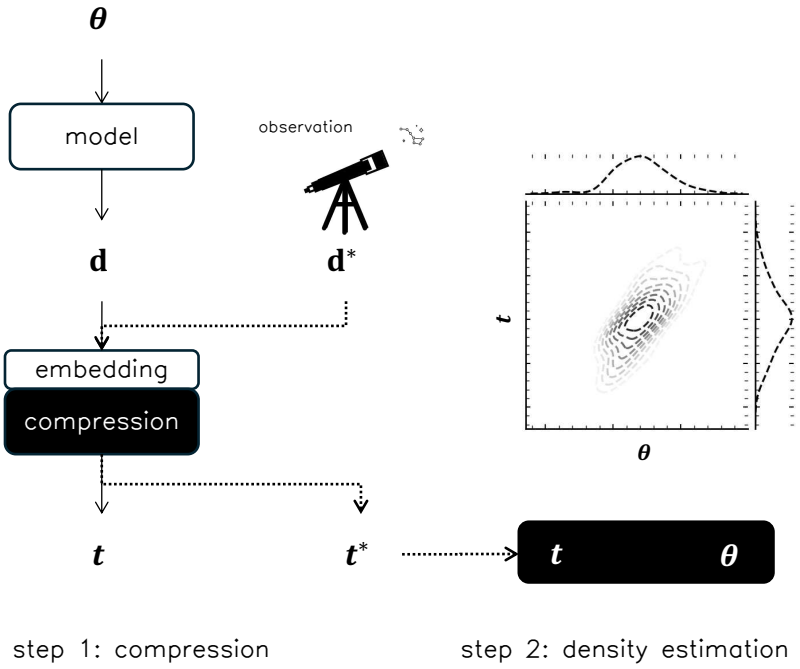


Figure 5.1: An implicit inference problem in two steps. Step 1 consists of learning an informative compression from data to summaries t. Summaries and parameter density estimation is then performed in Step 2 using either neural density estimators or accept-reject schemes. For target data inference (dashed arrows), an observation is made, compressed in an identical fashion to simulations, and then inserted into the density estimation scheme.

5.1.2 Step 2: Density Estimation

Once informative summaries are learned, the posterior distribution can be estimated using ABC or a neural density estimator. We will review several common approaches below.

Approximate Bayesian Computation revisited

We have already shown ABC to be a way to approximate Bayesian posterior calculation, but here discuss a modification to the framework. Population Monte Carlo (PMC) is an algorithm for obtaining a posterior distribution of parameters making use of weighted draws from the prior distribution (Kitagawa, 1996). The algorithm begins by drawing N parameter vectors from the prior $\theta_k^t \sim p(\theta)$ where $k \in [1, N]$ indexes the parameters, t marks the number of sampling iterations, and N is the number of total posterior points desired by the user. Every iteration t parameters are drawn from the prior, fed through a simulator to produce data, and then weighted by a distance metric to the target data. The weighted distribution of N parameter vectors is then used as the proposal distribution for the $t+1$ draw. The PMC-ABC thus iteratively approaches the true posterior, provided the weighted distance metric is chosen appropriately (Tavaré et al., 1997). Given a known Fisher information matrix evaluated for a fiducial set of compressed summaries, Alsing & Wandelt (2018) present the optimal distance measure between compressed, simulated summaries $\mathbf{x}_{ik}^{s,t}$ and target summaries $\mathbf{x}$:

$$\varrho_k^t = \sqrt{(\mathbf{x}_{ik}^{s,t} - \mathbf{x})^T \mathbf{F} (\mathbf{x}_{ik}^{s,t} - \mathbf{x})} \tag{5.1}$$

where s indexes the summaries, and i labels the random initialisation (neglected for subsequent sampling steps by PMC-ABC). For naïve ABC, a small acceptance criterion ϵ is first chosen. Then for each iteration, simulations are performed, compressed to summaries $\mathbf{x}_{ik}^{s,t}$, and accepted if $\varrho_k^t \leq \epsilon$

PMC-ABC modifies this naïve approach. Every iteration, an acceptance condition $\varepsilon^t = 75\%$ can be chosen such that the 75^{th} percentile of the set of summary distances $\{\varrho_k^t | k \in [1, N]\}$ are kept, with the remaining samples used to draw parameter vectors for the next iteration. θ_k^{t+1} are drawn from a proposal Gaussian distribution with mean θ_k^t and weighted covariance matrix, $\mathbf{C}_t$ computed from the previous iteration's parameter values. The new weighting scheme is given by:

$$W_k^{t+1} = \frac{p(\theta_k^{t+1})}{\sum_{j=1}^{N} W_j^t \mathcal{N}(\theta_k^{t+1}; \theta_j^t, \mathbf{C}_t)} \tag{5.2}$$

where $p(\theta_k^{t+1})$ is the prior at θ_k^{t+1} and $\mathcal{N}(\theta_k^{t+1}; \theta_j^t, \mathbf{C}_t)$ is a normal distribution centred on the θ_j^t. The initial weighting is chosen to be uniform for all k posterior points, $W_k^0 = 1/N$. The draws from this updated Gaussian distribution are repeated until the acceptance criterion is met, e.g. until $\varrho_k^{t+1} \leq \varepsilon^t$ for each of the rejected k samples. This completes a the first sampling iteration, allowing the weights

W_k^{t+1} to be computed. Sampling then continues with θ_k^{t+1} promoted to θ_k^t and used to find $\mathbf{C}_{t+1}$. Likewise the new acceptance condition can be computed from the 75^{th} percentile of the promoted distances $\{\varrho_k^{t+1} | k \in [1, N]\}$. Sampling can be stopped when the number of draws from $\mathcal{N}(\theta_k^{t+1}; \theta_k^t, \mathbf{C}_t)$ at iteration t far exceeds the number of of desired accepted parameters in the posterior, N. In this scenario, the posterior has approached its true value and does not change much between successive iterations of the PMC-ABC.

Neural Density Estimation

Neural density estimation circumvents the need for a tractable likelihood $p(\mathbf{d}|\theta)$, and instead seeks to parameterise the underlying, implicit likelihood or posterior present in forward simulations of the data. As discussed in Section 4.3.5, NDEs are neural networks that give some estimate $q(\mathbf{y}|\mathbf{x}; \varphi)$ of the desired conditional probability distribution by varying weights and biases (parameterised as φ) to minimize the loss

$$U(\varphi) = -\sum_{i=1}^{N} \ln q(\mathbf{y}_i|\mathbf{x}_i; \varphi), \tag{5.3}$$

over batches of parameter-data samples drawn from the joint distribution $(\mathbf{y}_i, \mathbf{x}_i) \curvearrowleft p(\theta, \mathbf{x}_i)$. This loss is equivalent to minimising the Kullback-Leibler divergence between the target distribution and q (Kullback & Leibler, 1951). Using this scheme we can parameterise either the sampling distribution of summaries $p(t|\theta)$ (Neural Likelihood Estimation, NLE) or learn the posterior directly $p(\theta|t)$ via Neural Posterior Estimation (NPE). Parameterising the posterior is simple to implement, but automatically absorbs the training data distribution into the surrogate density estimation

$$q(\theta|t; \varphi) \approx p(\theta|t) \propto p(t|\theta)p(\theta), \tag{5.4}$$

which can have undesired effects if the training simulations are not drawn from the intended prior distribution (Alsing et al., 2019; Cranmer et al., 2020a). A way to circumvent this issue is to employ NLE, which only parameterises the sampling distribution of the summaries:

$$q(t|\theta; \varphi) \approx p(t|\theta), \tag{5.5}$$

and can be learned over training samples irrespective of the underlying prior distribution. To obtain the posterior, the surrogate likelihood must be sampled using an MCMC scheme over the user-

intended prior. This allows for *sequential* training of the surrogate whereby simulations can be drawn from an intermediate posterior, appended to the sample set $\{t_i, \theta_i\}$, and training can continue to improve the surrogate's weights φ (Papamakarios et al., 2019b; Alsing et al., 2019; Makinen et al., 2021).

Another avenue for density estimation comes in the form of Neural Ratio Estimation (NRE) (Miller et al., 2021b; Cole et al., 2022), in which a binary classifier is tasked with distinguishing whether a simulation $\mathbf{d}$ is generated from a parameter θ. Writing out the posterior,

$$p(\theta|\mathbf{d}) = \frac{p(\mathbf{d}|\theta)}{p(\mathbf{d})}p(\theta) = r(\mathbf{d}|\theta)p(\theta), \tag{5.6}$$

where the ratio between the likelihood and evidence can be written (for each marginal $\{\theta_k\}_{k=1}^{n_{\rm params}}$):

$$r_k(\mathbf{d}|\theta_k) = \frac{p(\mathbf{d}|\theta_k)}{p(\mathbf{d})} = \frac{p(\mathbf{d},\theta)}{p(\mathbf{d})p(\theta_k)} = \frac{p(\theta_k|\mathbf{d})}{p(\theta_k)}. \tag{5.7}$$

A neural network is trained to minimise binary cross entropy between parameter-data pairs drawn from the joint $p(\mathbf{d},\theta)$ and marginal $p(\mathbf{d})p(\theta)$ distributions to parameterise each r_k over the prior support, with which the posterior can be reconstructed using Eq. 5.7. In practice, an iterative, adaptive scheme must be employed in which the prior support is truncated akin to sequential NLE to tightly recover posteriors.

5.1.3 Step 3: Posterior Coverage

How do you know when a surrogate for a posterior distribution is "good enough"? Once a surrogate $\hat{p}(\theta|\mathbf{d})$ for the conditional probability density $p(\theta|\mathbf{d})$ has been obtained, how can we be sure that the surrogate correctly describes the true, underlying posterior distribution ? Ideally we could construct a test using a set of simulations not used to obtain the estimator $\hat{p}(\theta|\mathbf{d})$, that are drawn from the same joint distribution (e.g. prior and simulator) $p(\theta,\mathbf{d})$, as the training set. Coverage checks test whether credible intervals contain the expected probabilities, or fraction of simulations. We can view the posterior inference, which yields $p(\theta|\mathbf{d}_O)$ at a given observed data $\mathbf{d}_O$ as a way to obtain a credible interval for a parameter θ. We draw test parameters $\theta_{\rm test}$ from the prior specified $p(\theta)$, which yields output data or summary $\mathbf{d}_{\rm test}$, for which a posterior $p(\theta|\mathbf{d}_{\rm test})$ can be obtained. For a pre-defined fixed interval (e.g. 68%), the true test parameter value should fall in this interval 68% of the time. This

procedure can be repeated with many θ_{test} drawn from the prior, from which the *expected* coverage can be computed, enhancing confidence in the estimation of the true posterior. Coverage tests such as the probability integral transform (PIT) test rely on computing the expected posterior coverage (EPC), but can fail to flag poor density estimation in some pernicious circumstances. Lemos et al. (2023a) propose a more robust algorithm, "Test of Accuracy with Random Points" (TARP), that safeguards against these cases. The first algorithm presents a necessary condition that needs to be met by a surrogate density estimator $\hat{p}$, with the second ensuring a condition that is sufficient.

Robustness. Coverage checks whether the distribution being parameterised behaves correctly. It does *not*, however, guard against *unknown* unknowns, e.g. out-of-distribution effects like those addressed in Section 4.7, when a compressor or density surrogate is tasked with compressing anomalous data. With known effects not varied in the forward model, test simulations with known parameter values subject to these effects can be generated and passed through the pipeline to check for significant deviation in summary or posterior output, like those illustrated in Chapter 11 for Dark Energy Survey mock data. Model misspecification and posterior predictive checks can be handled in the calculation of the Bayesian evidence, as discussed in Section 3.1.3.

5.1.4 Learning the Joint Distribution

Generative modelling can be leveraged to encompass both data emulation and parameter inference. In the explicit, classical case, this takes the form of Bayesian Hierarchical modelling (BHM), in which all random variables and their priors must be specified and then sampled over. In cosmology, efforts such as the Bayesian Origin Reconstruction from Galaxies (BORG) (Jasche & Wandelt, 2013; Leclercq, 2015; Doeser et al., 2024) attempt to reconstruct initial conditions by varying density field voxels alongside cosmology (Porqueres et al., 2021a). It is also possible, albeit challenging, to learn the joint distribution $p(\theta, \mathbf{d})$ from simulations alone using generative neural models. Legin et al. (2023) present a diffusion model to sample initial conditions, conditioned on an observed dark matter field for a fixed cosmology, and Mudur et al. (2025) incorporate parameter sampling from a conditional diffusion model.

5.1.5 Model Comparison

Model comparison, introduced in depth in Section 3.1.3, can also be automated using neural statistics. Jeffrey & Wandelt (2024) present Evidence Networks, in which evidence calculation between two models M_1 and M_2 is parameterised as a classification problem. The evidence term is

$$p(\mathbf{d}|M_1) = \int p(\mathbf{d}|\theta, M_1)p(\theta|M_1)d\theta, \tag{5.8}$$

and the *model evidence* is

$$\frac{p(M_1|\mathbf{d})}{p(M_2|\mathbf{d})} = \frac{p(\mathbf{d}|M_1)}{p(\mathbf{d}|M_2)}\frac{p(M_1)}{p(M_2)}. \tag{5.9}$$

If both models are equally probable, e.g. $p(M_1) = p(M_2)$, then the Bayes Factor can be written as

$$K = \frac{p(\mathbf{d}|M_1)}{p(\mathbf{d}|M_2)}. \tag{5.10}$$

Evidence networks cast the estimation of K as a classification problem. Data from each model are generated over the prior volume for each sets of parameters:

$$\mathbf{d}_1 \sim p(\theta_1, M_1); \quad \mathbf{d}_2 \sim p(\theta_2, M_2). \tag{5.11}$$

The parameters from each model are discarded (marginalised over) but the data are shuffled together. A classifier is then tasked with labelling whether data were generated from M_1 or M_2. Although this method requires a large number of simulations from each joint distribution, Jeffrey & Wandelt (2024) show that this implicit approach is still less expensive in terms of likelihood (forward model) calls than nested sampling methods used for evidence calculation (Feroz et al., 2009). Another useful aspect of evidence networks also discussed is their application to posterior predictive tests (PPT) (see e.g. Gelman et al., 2013; MacKay, 2002). PPT splits a dataset into two or more subsets, $\mathbf{d} \to \mathbf{d}_a, \mathbf{d}_b$, and estimates the relative probabilities of two models having predicted $\mathbf{d}_b$ conditional on observing $\mathbf{d}_a$:

$$p(\mathbf{d}_b|M_i, \mathbf{d}_a) = \int p(\mathbf{d}_b|\theta, \mathbf{d}_b)p(\theta|\mathbf{d}_a)\ d\theta, \tag{5.12}$$

for $i = 1, 2$. After evaluating the integral for both models, the posterior odds ratio is

$$K_{\mathrm{PPT}} = \frac{p(\mathbf{d}_b|M_1, \mathbf{d}_a)}{p(\mathbf{d}_b|M_2, \mathbf{d}_a)}. \tag{5.13}$$

This ratio can be manipulated:

$$\frac{p(\mathbf{d}_b|M_1,\mathbf{d}_a)}{p(\mathbf{d}_b|M_2,\mathbf{d}_a)} = \frac{p(\mathbf{d}_a,\mathbf{d}_b|M_1)}{p(\mathbf{d}_a,\mathbf{d}_b|M_2)}\frac{p(\mathbf{d}_a|M_1)}{p(\mathbf{d}_a|M_2)} = \frac{K(\mathbf{d})}{K(\mathbf{d}_a)}, \tag{5.14}$$

which is the ratio of the outputs of two evidence networks.

5.2 Cosmology as an Optimisation Problem

Until now, cosmology could be thought of in this setting as multi-generational optimisation problems. Since the 1960's, cosmologists worked to link theory to data and data to statistics using analytic formulae and handfuls of expensive simulations to calibrate their choices. With today's technology, we now have the ability to streamline this process by extracting summaries from simulated data automatically via implicit inference and neural compression. This gives us the freedom to work outside of the space of analytically tractable likelihoods and make use of more of the data collected by our telescopes.

Armed with the statistical and computational tools described, we can now start to think of cosmology (and science) as an optimisation problem. We have a descriptive theoretical framework (the cosmological solution to General Relativity), as well as interesting, unknown quantities (such as Dark Matter, massive neutrinos, Dark Energy, ...) that control how our Universe might evolve and behave over time. Our ultimate objective will be to reduce our uncertainty about these model parameters as much as possible given the data we observe. The path to minimising this objective consists of

1. Modelling the data and systematics to high precision (resolution)
2. Finding statistics of the data that are maximally sensitive to the interesting model parameters (compression)
3. Comparing competing models and reconciling statistical tensions in predictions

We can map these steps onto the factorisation of the joint distribution introduced in Chapter 3, shown in Fig 5.2. To achieve 1), the data generation process needs to be understood. In the case of weak lensing noise, for example, the shape noise on top of lensing shear fields can be modelled as Gaussian from first principles. Defining the tomographic redshift distribution of observed sources

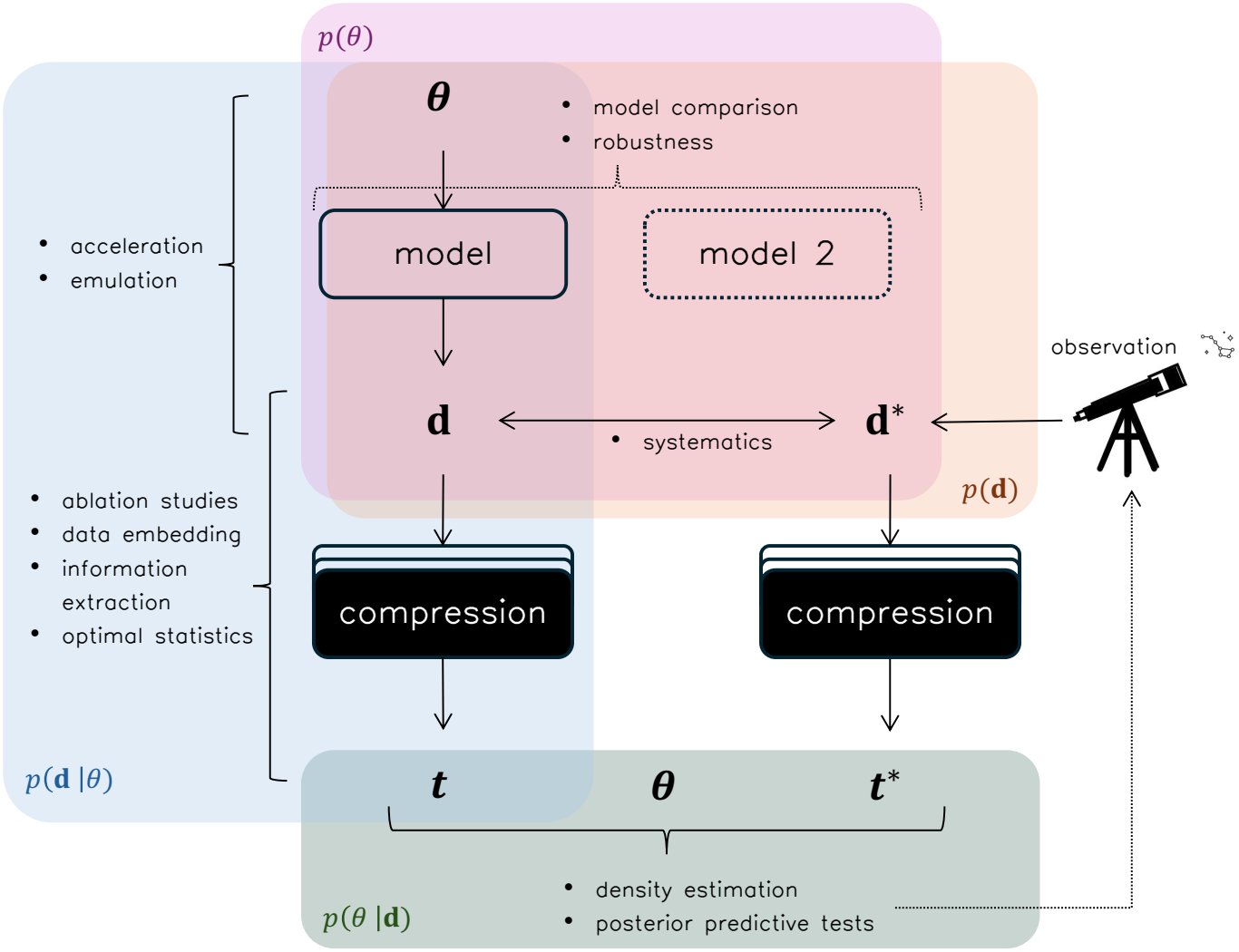


Figure 5.2: Cosmology as an optimisation problem, mapped over a joint distribution flowchart. Brackets indicate processes that can be accelerated or automated using implicit (neural) methods.

$n(z)$, however, is less obvious and has recently been tackled using generative models (Alsing et al., 2024). We have previously shown (Sec 4.7) that increasing the available number of simulations can improve concept learning. Speeding up expensive forward models like N-body simulations or systematic removal is the goal of neural *emulation* schemes (Bartlett et al., 2025; Jamieson et al., 2023; Pandey et al., 2024; Makinen et al., 2020), which can be used to augment the training set for learning summaries or estimate covariance matrices (Chartier & Wandelt, 2021).

The research presented in this thesis will largely focus on 2) finding optimal statistics from data. We will highlight the advantages of *interpretable* statistics, both at the level of the data (symmetries and ablation studies), as well as leveraging domain knowledge for more efficient neural optimisation objectives (loss functions).

Future work in implicit cosmological inference should focus on 3). Tensions in parameter predictions from early and late universe probes will be exacerbated by new, massive surveys, requiring careful handling of systematics and how they might affect existing (implicit) models. Unknown systematics could be degenerate with new pieces of a physical model. Advances such as generative modelling could help marginalise over unknown systematics, and evidence networks can automate model comparison such that the data decide which model from a suite is preferred.

CHAPTER 6

COSMIC FIELD INFORMATION EXTRACTION

We will illustrate an example optimisation problem using a non-trivial toy model for a non-Gaussian cosmological overdensity field. The parameters we choose here act as proxies for the cosmological parameters studied in subsequent chapters, mimicking signatures of Ω_m and w that might leak into large-scale structure morphology. We will utilise Information Maximising Neural Networks as our compression scheme, and we will show that our locally-optimal statistics for the problem are also *lossless* according to the definition in Chapter 4.

6.1 Probing Nongaussianity at the Field Level

IMNNs have been shown to saturate known likelihoods at the level of high-dimensional data like cosmological images. Makinen et al. (2021) showed that convolutional neural networks could capture the parameter information present in the pixel-level posterior of 2D correlated fields, such as Gaussian and lognormal cosmological mocks generated from known power spectra. In these cases the IMNN compression network outputs an *optimal* and *lossless* statistic; the output numbers are maximally informative for the architecture and data geometry chosen, and those numbers contain the same number of bits (or nats) as the full field likelihood.

It is of interest to ask whether the same information saturation is possible for fields that begin to deviate from a correlated prescription, as well as those that depend on a Bayesian hierarchy of random variables. To illustrate this, we will investigate the toy lognormal field model employed by Leclercq & Heavens (2021), which has an analytically-calculable Fisher information and whose posterior can be sampled exactly. We will illustrate our two-step neural inference procedure, and show that the chosen neural architecture produces optimal *and* lossless statistics for this more challenging problem.

6.1.1 Toy Lognormal Fields

First proposed by Coles & Jones (1991) as a way to describe matter distribution, the lognormal transformation is widely used in numerical simulations of the density field because of its computational efficiency. The starting point for generating a lognormal field is a Gaussian random field (GRF) of size $N \times N$, with $N_{\rm pix} = N^2$ the total number of pixels. Our Gaussian random field is generated from a two-point correlation function (2PCF), $\xi_G(r)$, parameterized by a single parameter, β:

$$p(f_{\rm G}|\beta) = \mathcal{N}(0, \xi_G(r)) \quad \text{with} \quad \xi_G(r) = \exp\left(-\frac{1}{4}\frac{r^2}{\beta^2}\right) \tag{6.1}$$

where r is the separation between two gridpoints, and β is in units of pixels. Large values of β correspond to larger correlation neighbourhoods (larger groups of brighter pixels), while smaller values correspond to smaller correlation lengths. The GRF is then transformed into a lognormal field via the transformation

$$f_{\rm LN} = \exp\left(\alpha f_{\rm G}\right) \tag{6.2}$$

where $\alpha > 0$ is a free parameter for "non-Gaussianity": larger values (≥ 0.2) of α yield more non-Gaussian fields, and the field becomes indistinguishably Gaussian for the limit $\alpha \to 0$. In cosmological terms, β is a proxy for the initial, Gaussian matter density perturbations, such as those measured in the CMB (Ade et al., 2016). Transformation by α then represents the nonlinear growth of the matter field for late-time cosmology. We can write this model down as a Bayesian Hierarchical Model (BHM), shown in Fig 6.1 beginning from the hyper-priors, $p(\alpha)$ and $p(\beta)$, which we take to be uniform. This plate diagram is both an experiment schematic as well as a mathematical object; random variables are denoted by white circles to be drawn from parent distributions in grey boxes. Arrows between variables represent deterministic operations, and white boxes indicate the target data and its transformations which are immutable.

For the field-level BHM solution, the hierarchical model is sampled with respect to the fixed, observed target data $\mathbf{d}^*$ at the level of each pixel in the final log-normal field $f_{\rm LN}$. To compare posterior information capture, a fixed IMNN compressor is used to generate summaries over the prior and perform ABC with respect to the identically-compressed target summaries $\mathbf{t}^*$. For this study we restrict ourselves to small $N = 20$ fields, following Leclercq & Heavens. This is to ensure tractability of the BHM computation, since the real-space r^2 vector needed to specify the pairwise multivariate covariance grows as N^4.

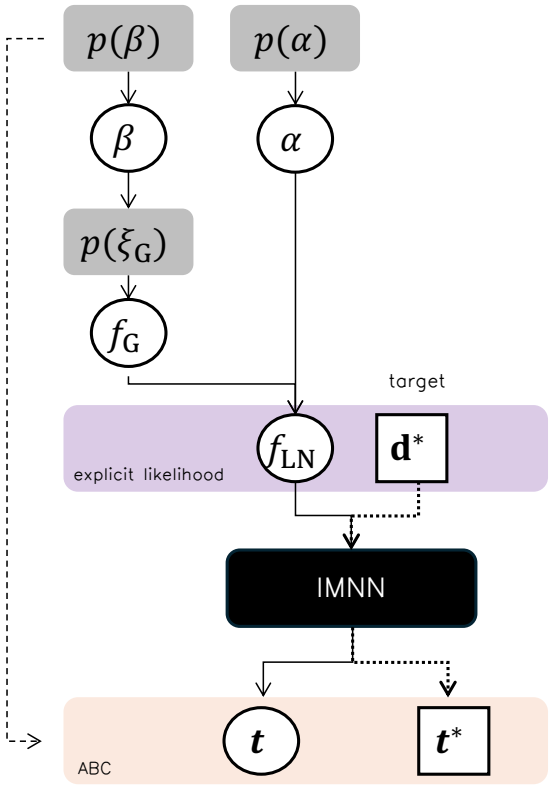


Figure 6.1: Bayesian hierarchical model for techniques for inferring parameters from toy lognormal fields. Parameters are first drawn from the prior (top) to generate a power spectrum, which is then used to generate a random Gaussian field realization, f_G, and then transformed to $f_{\rm LN}$. For field-level inference, field data and parameters are sampled using a Hamiltonian Monte Carlo (HMC) scheme. In the IMNN scheme, field data is passed through the (trained) IMNN to generate summaries for ABC inference. Grey shaded boxes indicate probability distributions, white bubbles indicate random variables, and arrows denote deterministic functions. The dashed arrow on the lefthand side shows parameter draws used for density estimation.

6.2 Fisher Information Saturation

We will first investigate whether an IMNN can saturate the known Fisher information at a chosen fiducial point.

6.2.1 Analytic Fisher Matrix

With such a simple model for our lognormal field, we can readily derive the analytic Fisher matrix and expected information content of the fields at a fiducial point. To ease notation, we denote $\mathbf{f} \equiv f_{\rm LN}$ and $\mathbf{g} \equiv f_{\rm G}$. We begin by writing the relation between a Gaussian field pixel, g_i, and lognormal field

pixel, f_i:

$$f_i = \exp(\alpha g_i) \quad \text{with} \quad \frac{df_i}{dg_j} = \alpha \exp(\alpha g_i)\delta^{\mathcal{K}}_{ij} \tag{6.3}$$

where $\delta^{\mathcal{K}}_{ij}$ is the Kronecker delta. We can then write the probability densities as

$$p(\mathbf{f})d\mathbf{f} = p(\mathbf{g})d\mathbf{g} \quad \implies p(\mathbf{f}) = p(\mathbf{g})\frac{d\mathbf{g}}{d\mathbf{f}} \tag{6.4}$$

This enables us to write

$$p(\mathbf{f}|\alpha,\beta) = \frac{1}{\sqrt{|2\pi\xi|}} \exp\left(-\frac{1}{2} g_i \xi^{-1}_{ij} g_j\right) \frac{1}{\prod_i \alpha \exp(\alpha g_i)} \tag{6.5}$$

where $\xi \equiv \xi_G$ is the Gaussian 2PCF. We are free to take the logarithm

$$\ln p(\mathbf{f}|\alpha,\beta) = \text{const.} - \frac{1}{2}\ln|\xi| - \frac{1}{2\alpha^2}\ln f_i \xi^{-1}_{ij} \ln f_j - \sum_i^{N_{\rm pix}} \ln\alpha - \sum_i^{N_{\rm pix}} \ln f_i \tag{6.6}$$

Now we can proceed to taking the derivatives of $\ln p(\mathbf{f}|\alpha,\beta)$ required by the Fisher formalism:

$$\frac{\partial \ln p}{\partial \alpha} = -\frac{N_{\rm pix}}{\alpha} + \frac{1}{\alpha^3} \ln f_i \; \xi^{-1}_{ij} \ln f_j \tag{6.7}$$

$$\frac{\partial^2 \ln p}{\partial \alpha^2} = \frac{N_{\rm pix}}{\alpha^2} - \frac{3}{\alpha^4} \ln f_i \xi^{-1}_{ij} \ln f_j \tag{6.8}$$

$$\implies F_{\alpha\alpha} = -\left\langle \frac{\partial^2 \ln p}{\partial \alpha^2} \right\rangle = \frac{N_{\rm pix}}{\alpha^2} + \frac{3N_{\rm pix}}{\alpha^2} = \frac{2N_{\rm pix}}{\alpha^2} \tag{6.9}$$

where in the last line we used $g_i = (1/\alpha)\ln f_i$ and the fact that $\langle g_i \xi^{-1}_{ij} g_j \rangle = 1$ in the expectation. Next we proceed to the off-diagonal elements, $F_{\alpha\beta} = F_{\beta\alpha}$:

$$F_{\alpha\beta} = -\left\langle \frac{\partial^2 \ln p}{\partial\alpha\partial\beta} \right\rangle = \frac{1}{\alpha}\left\langle g_i \xi^{-1} \xi_{,\beta} \xi^{-1} g_j \right\rangle = \frac{1}{\alpha} \text{tr}\left(\xi^{-1}\xi_{,\beta}\right) \tag{6.10}$$

where we used the identity $C^{-1}_{,\lambda} = -C^{-1}C_{,\lambda}C^{-1}$ and the fact that ξ is invertible. Finally, we have the derivatives with respect to β:

$$\frac{\partial \ln p}{\partial \beta} = -\frac{1}{2}\text{tr}\left(\xi^{-1}\xi,\beta\right) - \frac{1}{2}\text{tr}\left(\mathbf{g}^T\left(-\xi^{-1}\xi_{,\beta}\xi^{-1}\right)\mathbf{g}\right) \tag{6.11}$$

yielding the second derivative

$$\frac{\partial^2 \ln p}{\partial \beta^2} = -\frac{1}{2}\,\mathrm{tr}\left(-\xi^{-1}\xi_{,\beta}\xi^{-1}\xi_{,\beta}\right) - \frac{1}{2}\,\mathrm{tr}\left(\xi^{-1}\xi_{,\beta\beta}\right)$$

$$- \frac{1}{2}\,\mathrm{tr}\left(2\mathbf{g}^T\xi^{-1}\xi_{,\beta}\xi^{-1}\xi_{,\beta}\xi^{-1}\mathbf{g} - \mathbf{g}^T\xi^{-1}\xi_{,\beta\beta}\xi^{-1}\mathbf{g}\right) \quad (6.12)$$

All but the first term evaluate to zero in the expectation, yielding

$$F_{\beta\beta} = -\left\langle \frac{\partial^2 \ln p}{\partial \beta^2} \right\rangle = \frac{1}{2}\,\mathrm{tr}\left(-\xi^{-1}\xi_{,\beta}\xi^{-1}\xi_{,\beta}\right) \quad (6.13)$$

which is the same result obtained for purely Gaussian fields (see Appendix A in Makinen et al., 2021). Altogether, the Fisher matrix reads:

$$\mathbf{F} = \begin{pmatrix} \frac{2N_{\mathrm{pix}}}{\alpha^2} & \frac{1}{\alpha}\,\mathrm{tr}\left(\xi^{-1}\xi_{,\beta}\right) \\ \frac{1}{\alpha}\,\mathrm{tr}\left(\xi^{-1}\xi_{,\beta}\right) & \frac{1}{2}\,\mathrm{tr}\left(-\xi^{-1}\xi_{,\beta}\xi^{-1}\xi_{,\beta}\right) \end{pmatrix}\Bigg|_{\alpha,\beta} \quad (6.14)$$

where we note that the Fisher is to be evaluated at a given θ. At $\theta_{\mathrm{fid}} = [1.0, 0.5]$, for a field of $N_{\mathrm{pix}} = N^2 = 20^2$, $\det \mathbf{F} = 3175571.8$. Having a closed-form expression for the information content is useful for both determining the expected minimum variance of estimated parameters (Cramèr-Rao bound), as well as assessing the compression network's performance.

6.2.2 IMNN Architecture

With the analytic Fisher Information known, we can test a neural compression scheme within the IMNN framework for the N_{pix}^2 field dataset. The network takes as input an $N_{\mathrm{pix}} \times N_{\mathrm{pix}}$ field, and extracts information via a convolutional compression scheme. We adopt an inception-block CNN to be most efficient in training (e.g Szegedy et al., 2016). For each inception block, data is passed through 5^2, 3^2, 1^2 convolutions and a `maxpool` layer in parallel, with outputs concatenated and passed to the next block, shown graphically in figure 6.2. We adopt three stride-4 inception blocks, each with $n_f = 55$ filters, quartering the output spatial dimensions, each followed by a `gelu` nonlinear activation function. Once spatial dimensions are of shape $(2, 2)$, we adopt a stride-2 inception block with 1^2 kernels followed by a 1^2 convolution to $n_{\mathrm{summaries}} = n_{\mathrm{params}} = 2$ summaries.

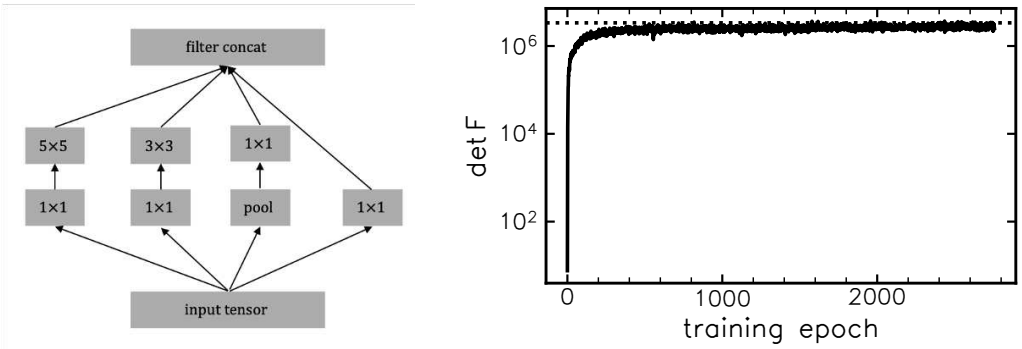


Figure 6.2: Inception block architecture (*left*). We adopt stride-4 downsampling before filter concatenation to quarter the spatial dimensions of the simulations. The IMNN saturates the analytic Fisher information at the fiducial point after 1500 training epochs (*right*).

To train the network, we generate 200 new simulation realizations on-the-fly each epoch. We make use of `Jax`'s autograd feature to compute the derivatives through network and simulation necessary for the summaries' Fisher information (Eq 4.38), obtaining both simulations and exact derivatives with respect to simulation parameters each epoch for the Fisher information computation. We train the network with the `Adam` optimizer (Kingma & Ba, 2014) until the Fisher information stops increasing for 500 epochs.

Results. Using the adjoint gradient computation through simulations and network, the IMNN optimisation is able to saturate the analytic information content within 1500 epochs of training (Fig 6.2). In the language of Chapter 4, the deterministic IMNN network is both an *optimal* and *lossless* statistic for simulations at $\theta = \theta_{\rm fid}$, in terms of the Fisher information.

6.3 Posterior Information Capture

But what does inference look like with these IMNN summaries, especially as we compress simulations away from the fiducial point ? Since we can evaluate our pixel-level likelihood exactly using a BHM framework, we have a ground truth to compare an ABC scheme using IMNN-compressed statistics.

Our target data will be generated from the same forward model, but slightly away from the fiducial point; $\theta^* = [0.9, 0.45]$, to assess information loss away from the IMNN training point.

For the ABC method, we generate $100,000$ simulations over the prior. Each field is then passed through the static IMNN to produce summaries using Eq 4.43. We can visualise the structure in the IMNN summary space with respect to the true parameters over the prior (Fig 6.3). The target

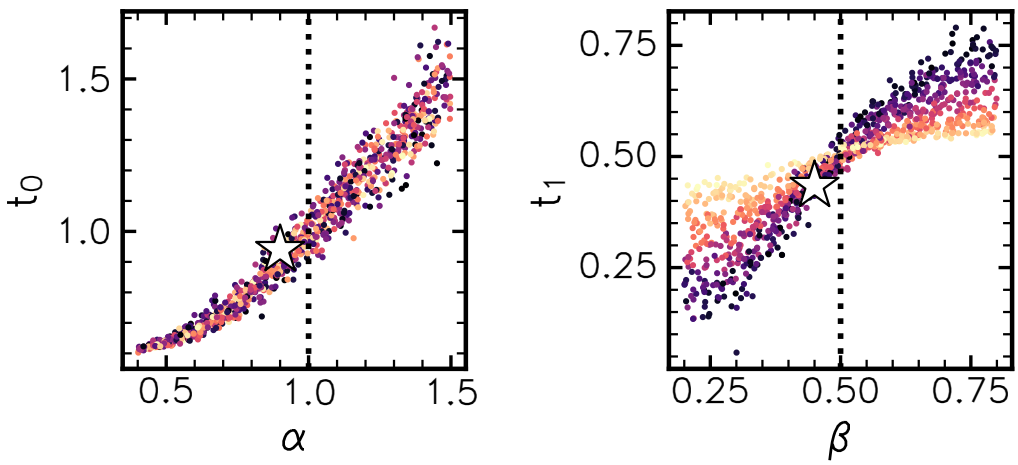


Figure 6.3: Optimised locally at the fiducial point (dashed vertical line), the IMNN summaries exhibit strong correlation with respect to parameters. Summary points are coloured with respect to the opposite parameter value, indicating a strong dependence of β information capture on the value of α. The first compressed target field summaries is shown in each plane as a white star.

simulations are also passed through the network to produce $\mathbf{t}^*$ (white star in Fig 6.3), and an ABC distance calculated with Eq. 5.1 between the prior simulations and targets. We set $\varepsilon = 0.025$ such that each target posterior is comprised of at least 200 samples. The accepted parameter values can now be interpreted as posterior samples.

6.3.1 Bayesian Hierarchical Modelling

For our "ground truth" posterior estimate for a given piece of data, we leverage a Bayesian Hierarchical Model (BHM) to sample from the prescribed uniform priors, corresponding to the orange box in figure 6.1. We can write out the posterior distribution explicitly in the noise-free case (e.g. each field pixel value is perfectly known):

$$p(\alpha, \beta, f_{\rm G}|f_{\rm LN}) \propto p(f_{\rm LN}|f_{\rm G}, \alpha, \beta)\ p(f_{\rm G}, \alpha, \beta) \tag{6.15}$$

$$= p(f_{\rm LN}|f_{\rm G}, \alpha)\ p(f_{\rm G}|\beta)\ p(\beta)\ p(\alpha) \tag{6.16}$$

$$= p(f_{\rm G}|\beta)\ p(\beta)\ p(\alpha) \tag{6.17}$$

where in the last line we note that in the noise-free case $p(f_{\rm LN}|f_{\rm G}, \alpha)$ is deterministic. Usually, BHM evaluations are high dimensional and require advanced sampling techniques such as Hamiltonian Monte Carlo (Phan et al., 2019a). Here, however, we only have two parameters, so we can evaluate our BHM on a 2D grid over the prior. For each prior point, the log-probability is com-

puted between the selected parameters and the input field values, $f_{\rm LN}$. This is done numerically in `tensorflow-probability` (Abadi et al., 2015), in which the lognormal transformation of the Gaussian field is performed via a bijector function which preserves the probability volume in the transformation. With the transformed distribution readily defined, an MCMC or grid-wise likelihood evaluation can be performed by evaluating the log-probability of the distribution given a chosen parameter vector.

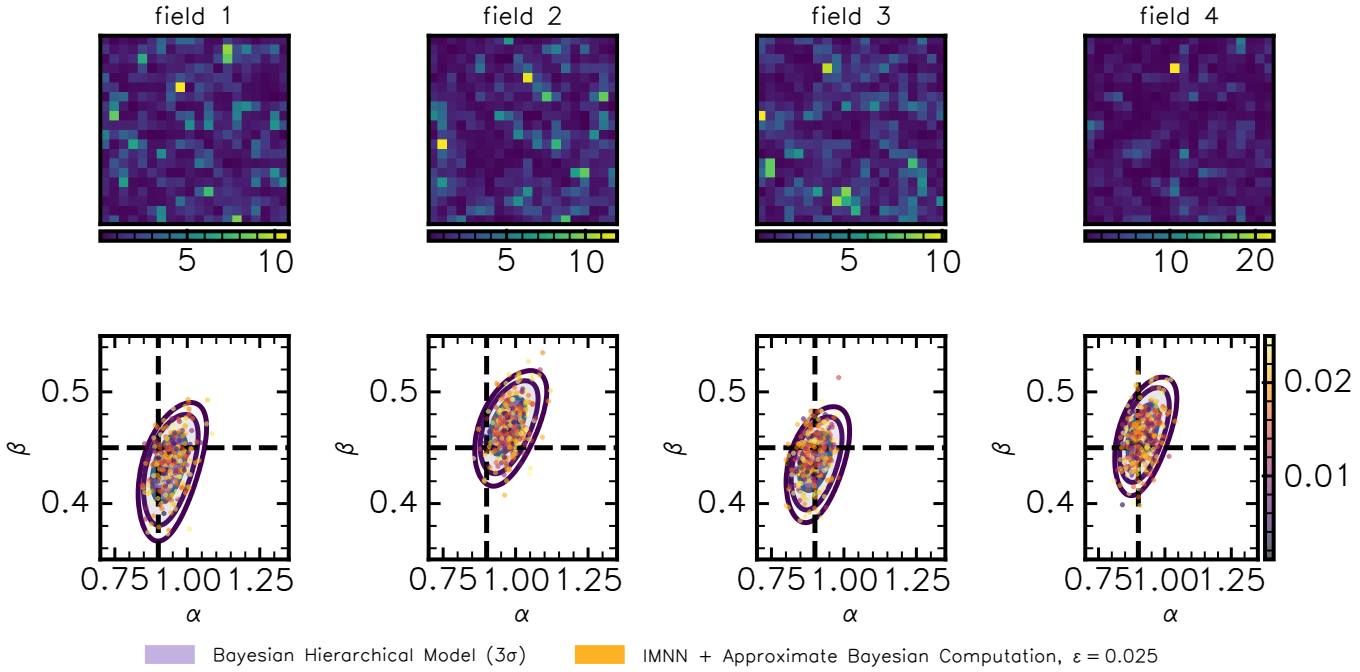


Figure 6.4: Inference comparison for four independent lognormal field realizations (top). We compare simulation-based inference using optimal IMNN summaries (orange scatter, coloured by ABC distance to target field summaries) to 3-sigma contours achieved from a BHM sampler (purple).

6.3.2 Results & Implications

We display results from the two inference methods in figure 6.4. We see that the accepted summaries around each target hug the 3σ contours computed for the exact field likelihood. This indicates that our lossless, optimal compression at $\theta = \theta_{\rm fid}$ is also *asymptotically lossless* for information capture across the prior–we can capture the posterior exactly near the fiducial point. We should note that this is rather remarkable–we trained a neural network to compress a miniature, non-Gaussian universe into two numbers that contain *the same amount of cosmological information as the full field of 400 pixels*.

But we still have unanswered questions: How does this scheme do in the presence of lots of noise and systematics ? Where does the information captured by the network come from–which features of the field are most informative ? Can this information capture be made more efficient by modifying the optimisation objective ?

6.3.3 Bonus: Leveraging Local Compression for Robustness

The local IMNN compression might seem at first inconvenient. Even though it requires fewer simulations per parameter to train a valid compression (requiring only simulations at $\theta^{\pm}$ to estimate derivatives), it can be regarded as "safe" machine learning. In the example above, we made the implicit (common) assumption that over the prior range for parameters α and β that the output fields, while random, should still vary *smoothly* with respect to the parameters. This is usually the case with physical systems, even otherwise discrete graphs (see Chapter 7). Looking at our summary scatter above, we see that the IMNN outputs summaries in a relatively smooth manifold with respect to parameters, even though it is trained at a single point. Let us now consider a "corruption" to

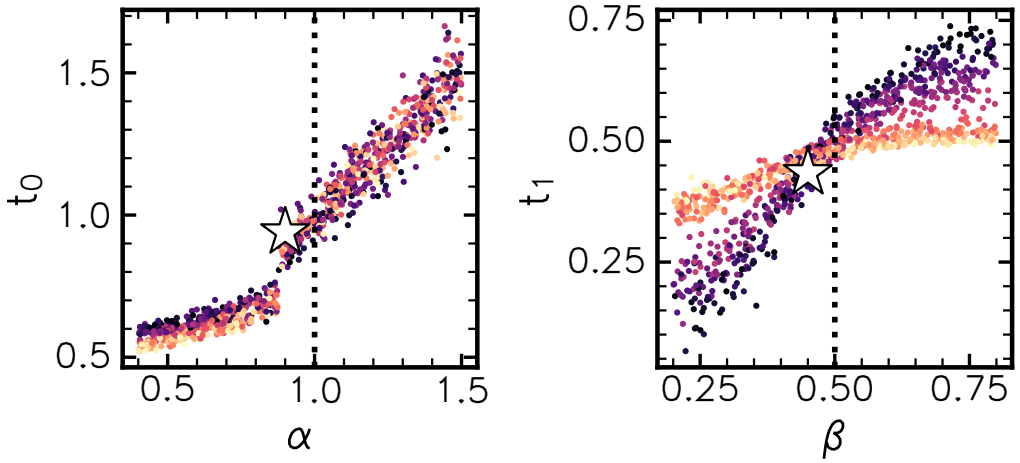


Figure 6.5: Introducing an additive, α-dependent factor for values of $\alpha < 0.87$ creates a discontinuity in the IMNN summary space.

our forward simulation. For $\alpha < 0.87$, we will add a small factor to the output field everywhere: $\mathbf{d} \leftarrow \mathbf{d} + (\alpha - 0.5)$. This is a *tiny* effect with respect to our field values, and one that doesn't change the variance of the field. But this can have an effect on the neural summarisation of the data. If we generate data using the same random seeds and prior values we did previously and feed them through the same IMNN, we obtain the summary scatter shown in Fig 6.5.

This sort of effect can happen in cosmological simulation suites due to grid misspecification or

improperly-normalised noise effects. Summary spaces do not have to be smooth, but detecting unwanted effects like these are important because they can affect the downstream inference, as displayed in Fig 6.6 In this example we corrupted the simulations passed to the neural network in a

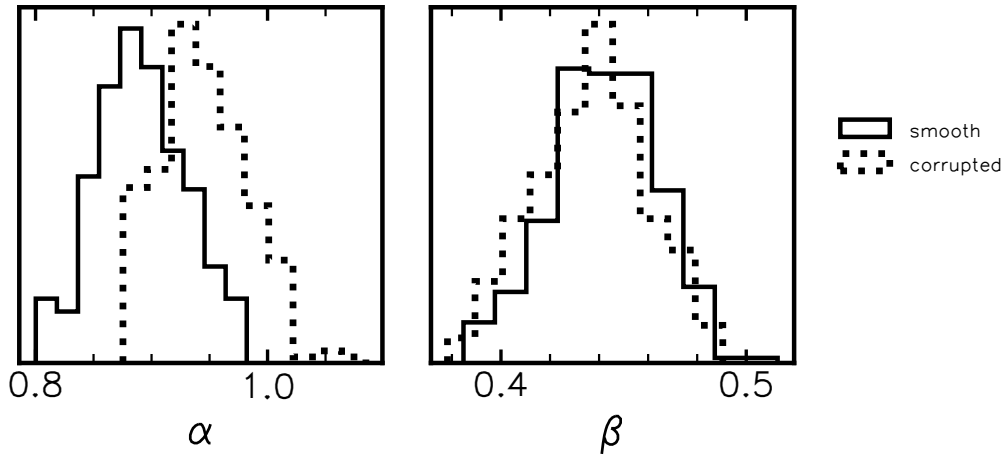


Figure 6.6: Unphysical effects in simulations can have an impact on downstream inference, shown here for field 3. Black lines show the parameter posterior marginals for the smooth simulations, and dashed lines indicate artificially tight marginals for simulations with the discrete bug in Fig 6.5.

systematic way. We could have equivalently added an "unknown" systematic to the target data and observed the IMNN summaries land "outside" of the summary cloud. Training a local compression in this way can help in robustness assessments in implicit inference.

CHAPTER 7

COSMIC GRAPHS

The Cosmic Graph: Optimal Information Extraction from Large-Scale Structure using Catalogues

T. Lucas Makinen[1,2], Tom Charnock[3], Pablo Lemos[4], Natalia Porqueres[1], Alan Heavens[1], Benjamin D. Wandelt[5,6]

[1]Imperial Centre for Inference and Cosmology (ICIC) & Astrophysics Group, Imperial College London, Blackett Laboratory, Prince Consort Road, London SW7 2AZ, United Kingdom

[2]Harvard & Smithsonian Center for Astrophysics, Observatory Building E, 60 Garden St, Cambridge, MA 02138, United States

[3]Freelance consultant in statistical modelling

[4]Department of Physics and Astronomy, University of Sussex, Brighton, BN1 9QH, UK

[5]Sorbonne Université, CNRS, UMR 7095, Institut d'Astrophysique de Paris, 98 bis boulevard Arago, 75014 Paris, France

[6]Center for Computational Astrophysics, Flatiron Institute, 162 5th Avenue, New York, NY 10010, USA

Accepted to the Open Journal of Astrophysics; Makinen et al. (2022)

Abstract

We present an implicit likelihood approach to quantifying cosmological information over discrete catalogue data, assembled as graphs. To do so, we explore cosmological parameter constraints using mock dark matter halo catalogues. We employ Information Maximising Neural Networks (IMNNs) to quantify Fisher information extraction as a function of graph representation. We a) demonstrate the high sensitivity of modular graph structure to the underlying cosmology in the noise-free limit, b) show that graph neural network summaries automatically combine mass and clustering information through comparisons to traditional statistics, c) demonstrate that networks can still extract information when catalogues are subject to noisy survey cuts, and d) illustrate how nonlinear IMNN summaries can be used as asymptotically optimal compressed statistics for Bayesian simulation-based inference. We reduce the area of joint Ω_m, σ_8 parameter constraints with small ($\sim$100 object) halo catalogues by a factor of 42 over the two-point correlation function, and

demonstrate that the networks automatically combine mass and clustering information. This work utilizes a new IMNN implementation over graph data in Jax, which can take advantage of either numerical or auto-differentiability. We also show that graph IMNNs successfully compress simulations away from the fiducial model at which the network is fitted, indicating a promising alternative to n-point statistics in catalogue simulation-based analyses.

7.1 Introduction

Modern cosmological analyses typically focus on obtaining theory and parameter constraints from compressed summary statistics obtained from field data such as the Cosmic Microwave Background or weak lensing mass-maps (Tegmark et al., 1997; Alsing & Wandelt, 2018; Jeffrey et al., 2020). Recently, field-level analyses like Porqueres et al. (2021b); Leclercq & Heavens (2021), although computationally expensive, have made it possible to sample the full field at the pixel level to ensure all survey information is accounted for in posterior construction for cosmological parameters.

However, the data collected by telescopes are often instantly compressed into discrete catalogues of sources, like galaxies and their underlying dark mater halos, or cosmic voids (Sutter et al., 2012; Kreisch et al., 2021). The typical approach taken to analyse galaxy cluster data is to "paint" identified sources onto a grid and perform luminosity peak counts in high-density regions as a tracer for underlying dark matter. Analyses of these catalogues usually focus on 2-point information, either in real or Fourier space. However, these statistics are only sufficient when the underlying field is Gaussian, which is not the case for late-time cosmic web structures. Finding a statistic with which to capture more of this information is an active area of research. Existing methods include the three-point correlation function (the bispectrum in Fourier space, e.g. Philcox & Ivanov (2022)), Minkowski functionals (Petri et al., 2013), the 1D probability distribution function (**?**), marked power spectra (Massara et al., 2022), minimum spanning trees (Barrow et al., 1985; Naidoo et al., 2019, 2022), and field-level sampling (Jasche & Wandelt, 2013; Ramanah et al., 2019; Porqueres et al., 2021b; Leclercq & Heavens, 2021; Leclercq, 2015; Jasche et al., 2015). However, truncating analyses to power or bispectra almost certainly discards information, especially for highly non-Gaussian fields, while field-level methods quickly become computationally expensive with increasing survey volume. Likewise, void cosmology constructs correlation functions from void positions and redshifts (Hamaus et al., 2015) to capture under-dense regions in structure formation. This sort of analysis usually

discards morphological features of voids, such as void ellipticity, resulting in a loss of information that could be relevant to the underlying cosmological model (Biswas et al., 2010; Lavaux & Wandelt, 2010; Xu et al., 2019b).

Graphs provide a natural way to describe the nonlinear aspects of large-scale structure (LSS). Dark matter halos and their galaxy clusters can be attributed to nodes (vertices), while filaments are traced by smaller halos and edges connecting neighbouring edges. In this representation, clustering under gravity can be translated into higher connectivity or number of edges. Higher order n-point functions can be computed efficiently for clusters, while avoiding the cost of computing extraneous connections across voids. Graph representation of LSS promises a more modular approach to information quantification, and compliments the existing body of literature. Minimum spanning trees (MSTs) have been used in cosmological analyses since Barrow et al. (1985), and subsequent studies have investigated using binned halo graph features from simulations as cosmological probes (Bhavsar & Ling, 1988; van de Weygaert et al., 1992; Krzewina & Saslaw, 1996; Ueda & Itoh, 1997; Coles et al., 1998; Adami & Mazure, 1999; Colberg, 2007; Alpaslan et al., 2014; Beuret et al., 2017; Libeskind et al., 2018; Bonnaire et al., 2020, 2022). More recently, Naidoo et al. (2022, 2019) use the minimum spanning tree (MST) computed from the *Quijote* simulations to compute the cosmological information by binning branch and shape features of the MST computed over the simulation suite. Yang & Yu (2022) illustrate graph-based approaches for modelling small-scale halo clustering in cosmological simulations.

The advent of deep learning in cosmology has made massive data generation and analysis more tractable. Many studies have investigated neural techniques for point estimate cosmological parameter extraction from cosmological fields via regression networks trained on simulation-parameter pairs (Pan et al., 2020; Ravanbakhsh et al., 2017; Kwon et al., 2020; Prelogović et al., 2021; Fluri et al., 2019, 2018; Matilla et al., 2020; Ribli et al., 2018; Gillet et al., 2019), field reconstruction (Dai & Seljak, 2022; Jamieson et al., 2023), foreground removal emulation (Makinen et al., 2020; Jeffrey et al., 2022), or cosmological parameters from graphs (Villanueva-Domingo & Villaescusa-Navarro, 2022) with squared loss. As reviewed in Villaescusa-Navarro et al. (2020a), these techniques can estimate the posterior mean of parameters (see also Jeffrey & Wandelt (2020)). This implies they require simulations drawn from a prior, specified at the time of training, not just near the parameters favored by the data. This adds to the variability that needs to be fit by the network.

We take a different approach: we consider halo catalogue graphs as our dataset and use Information

Maximising Neural Networks (IMNNs) to measure the Fisher information contained in these graphs. IMNNs are neural networks that compress data to informative nonlinear summaries, trained on simulations around a fiducial model to maximise the Fisher information (Charnock et al., 2018; Makinen et al., 2021), where, for the purposes of compression and forecasting only, the summary statistics are assumed to have a Gaussian sampling distribution. Neural networks can make use of all available data simultaneously, even saturating known field-level likelihoods (Makinen et al., 2021). This approach enables us to use asymptotically optimal nonlinear statistics (Alsing & Wandelt, 2018; Charnock et al., 2018) to then compute summaries and estimate maximum likelihood parameters and perform efficient implicit likelihood inference over a prior.

We combine this framework with a graph neural network (GNN) architecture. GNNs are well-suited to discrete and variable-length problems such as molecular classification, weather forecasting, and even physics (re)-discovery with symbolic regression (Lemos et al., 2022; Cranmer et al., 2020b), (see Battaglia et al. (2018) for a complete review).

Recent studies have made use of IMNNs for cosmology (Makinen et al., 2021; Fluri et al., 2021; Fluri et al., 2022), and highly non-Gaussian problems, such as galaxy type identification from multiband images (Livet et al., 2021). However, previous implementations relied on computing Fisher statistics for data with a *fixed input size*. Here, using GNNs, we extend the framework to a much more general class of problems. We will refer to graph IMNNs as gIMNNs.

We show how gIMNN summaries from catalogue graphs compares to traditional cosmological techniques *with respect to information extraction* using the *Quijote* halo catalogues (Villaescusa-Navarro et al., 2020b). We illustrate that by encoding physical symmetries and more descriptive graph attributes in the IMNN framework, we can extract more information from limited catalogues than traditional 2-point statistics.

The study is organised as follows: We present a graph description of large-scale structure in Section 7.2, followed by a review of the IMNN framework in the context of graph data in Section 7.3. In Section 7.4 we present our halo catalogue graph and GNN architectures and our main findings: We first investigate information as a function of increasing GNN depth and graph connectivity on both invariant and non-invariant graphs, and show that gIMNNs consistently extract more information than the 2-pt function. We next show that decorating graph nodes with mass further increases information extraction. Third, we explore the information stored in graph cardinality (the number of nodes or objects and edges connecting them) in the context of the halo mass function. Next, we

proceed to a more realistic case in which catalogue construction is subject to various levels of uncertainty in the halo mass determination. In Section 9.2 we conclude by showing how trained gIMNN summaries can be used as optimal compressors in simulation-based inference density estimation. We include supplementary descriptions of graph assembly and network generalization in Appendix 7.11.

7.2 Large-Scale Structure as a Graph

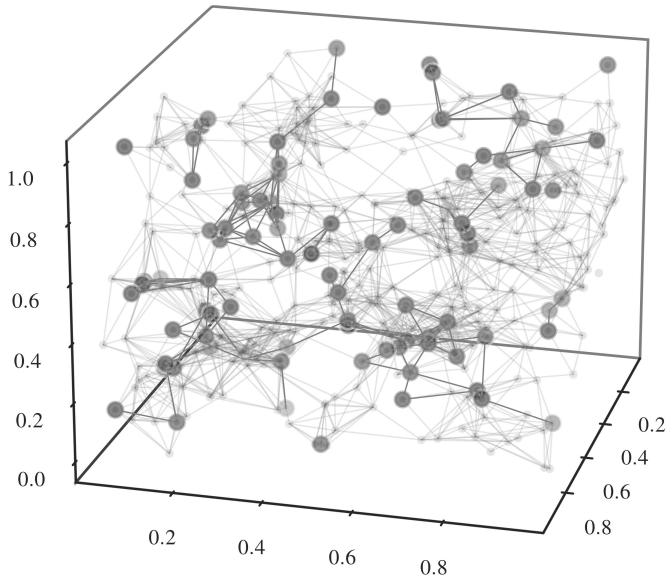


Figure 7.1: Dark matter halo graph representation of large-scale structure, constructed from a single *Quijote* simulation. The largest halos (grey) with a mass $M_i > 1.5 \times 10^{15}\ M_\odot$ trace the largest physical scales, here shown coloured by the log of the halo's mass, and connected to all neighbours within a radius of $r_{\rm connect} = 200$ Mpc. Smaller halo masses $M_i > 1.1 \times 10^{15}\ M_\odot$ (light grey) trace smaller scale clustering. The box encases a cosmological comoving volume of $(1{\rm Gpc})^3$.

Graphs provide a natural language with which to describe the cosmic web. Dark matter halos are attributed to nodes (vertices), while filaments are traced by smaller halos and edges, illustrated in Figure 7.1. In this representation, clustering under gravitational interactions can be translated into higher edge cardinality (number of edges). Higher order n-point functions can be computed efficiently for clusters, while avoiding the cost of computing extraneous connections across voids. Void catalogues (where edges would correspond to the walls separating the voids) can likewise be assembled into the dual of a halo graph. Graph construction also allows arbitrary extra information, such as the halo masses or peculiar velocities, for which the underlying sampling distribution is not known, to be combined in a nonlinear fashion in the form of node or edge labels, unlike (marked) correlation functions.

7.2.1 Graph Notation

We define a graph explicitly as a tuple $G = (\mathbf{u}, V, E)$, following the notation in Battaglia et al. (2018). The $\mathbf{u}$ is a global attribute of the graph, i.e. a label or global parameter value. $V = \{\mathbf{v}_i\}_{i=1:N^v}$ is the set of graph nodes, with cardinality N^v. The edge set $E = \{(\mathbf{e}_k, r_k, s_k)\}_{r_k=i, k=1:N^e}$, indexed by $k = 1 : N^e$, is comprised of vectors $\mathbf{e}_k$ of cardinality N^e, which may be directed, connected via receiving and sending indices between nodes, r_k and s_k. Senders and receivers can be equivalently parameterized by an adjacency matrix A_{ij} in which i and j index sender and receiver nodes, respectively. Each node, indexed by $i = 1 : N^v$, has a set of edges, $E'_i = \{(\mathbf{e}'_k, r_k, s_k)\}_{r_k=i, k=1:N^e}$, connected to it via a subset of senders and receivers. The full set of nodes is defined as $V = \{\mathbf{v}_i\}_{i=1:N^v}$, where each node $\mathbf{v}_i$ is a vector of features. In a physical system of particles, one might represent V as a set of individual particles' attributes, like mass, position, and velocity, with edges expressing interactions, such as forces, between particles. A global attribute of a graph might be a classification label, such as in molecule or cluster classification (Satorras et al., 2021; Kipf & Welling, 2017). Careful data representation on graphs can vastly simplify physical problems via inductive biases and symmetry capture (see e.g. Lemos et al., 2022; Cranmer et al., 2020b; Battaglia et al., 2018).

7.2.2 Halo Graphs

We define a dark matter halo graph $G = (\mathbf{u}, V^{\text{halo}}, E^{\text{halo}})$, constructed from a catalogue for a single realisation of the universe. We can equivalently define its dual, $H = (\mathbf{u}, V^{\text{void}}, E^{\text{void}})$, from a void catalogue. Note that if we assign global cosmological parameters to $\mathbf{u}$, G and H share this property. Hereafter we will focus on graphs from halo catalogues.

The graph framework allows the cardinality of a cosmological graph's nodes and edges to vary as a function of cosmological or survey parameters, reflecting the often strong dependence of the abundance of clusters on cosmological parameters. When assembling a graph from a halo catalogue, we choose to vary two physical parameters: a mass cut, M_{cut}, and a linking radius, r_{connect}. A halo i with a mass above M_{cut} is connected to a halo j if the absolute distance between halos i and j is less than r_{connect}, i.e. $|\mathbf{d}_{ij}| < r_{\text{connect}}$. We display the same catalogue at two mass cuts in Figure 7.1. A conservative $M_{\text{cut}} = 1.5 \times 10^{15} M_\odot$ (dark points) contains the heaviest halos and traces the largest scales, while smaller masses ($M_{\text{cut}} = 1.1 \times 10^{15} M_\odot$, light points) trace smaller scales. Each graph is connected by $r_{\text{connect}} = 200$ Mpc.

Graphs can be assembled from halo catalogues in one of two ways: as non-invariant or as invariant graphs. Non-invariant graphs have positions, $\boldsymbol{p}$, as node labels, setting $\mathbf{v}_i = \boldsymbol{p}_i$, with edges labelled as the relative distances between halos, $\mathbf{d}_{ij}$. This graph is *not* invariant under translations and rotations, as the node values are pinned to the underlying simulation grid. *Invariant* graphs have only *relative* positional information, all of which is stored in the edges. The cosmological models that we wish to constrain are invariant to rigid Euclidean group rotations and translations of the large-scale structure. In this work we include both representations for completeness.

Node features

In the invariant representation, graph nodes are 'decorated' with either an indicator $\mathbf{v}_i = v_i = 1$ in the undecorated case or the halo's scalar mass, $\mathbf{v}_i = v_i = M_i$. In the non-invariant case, nodes are also decorated with position $\mathbf{v}_i = (M_i, \mathbf{p}_i)$. We describe graph construction and padding details in Appendix 7.11.

Edge features

To construct invariant graphs, we impose translational symmetry by attributing functions of relative positions between halos on the edges. We compute the vector separations $\mathbf{d}_{ij} = \boldsymbol{p}_i - \boldsymbol{p}_j$ between all halos and do not link halos directly if $|\mathbf{d}_{ij}| > r_{\rm connect}$. For rotational invariance, we adopt Villanueva-Domingo & Villaescusa-Navarro (2022)'s notation and first compute the unit vectors $\boldsymbol{s}_{ij} = \mathbf{d}_{ij}/|\mathbf{d}_{ij}|$ and $\mathbf{n}_i = (\boldsymbol{p}_i - \bar{\boldsymbol{p}})/|\boldsymbol{p}_i - \bar{\boldsymbol{p}}|$ where $\bar{\boldsymbol{p}}$ is the centroid (or reference halo position). We then compute the direction cosines $a_{ij} = \mathbf{n}_i \cdot \mathbf{n}_j$ and $b_{ij} = \mathbf{n}_i \cdot \mathbf{s}_{ij}$. The normalized edge features for *invariant* halo graphs are then

$$\mathbf{e}_{ij} = [|\mathbf{d}_{ij}|/r_{\rm connect}, a_{ij}, b_{ij}]\,, \tag{7.1}$$

whilst for non-invariant graphs, $\mathbf{e}_{ij} = |\mathbf{d}_{ij}|/r_{\rm connect}$.

Global features

A halo graph's global features can be any quantity that describes the global properties of the system, in this case configuration or cosmology parameters. In a regression case, one might wish to label each halo graph simulation with a set of cosmological or hydrodynamical parameters, as done

in Villanueva-Domingo & Villaescusa-Navarro (2022), and fit a neural network to minimise some distance measure between the network output and these parameters. Here, global properties will be *arbitrary nonlinear summaries* of cosmology, learned in an unsupervised manner as a function of the graph's attributes using information maximising neural networks.

7.3 Information Maximising Neural Networks

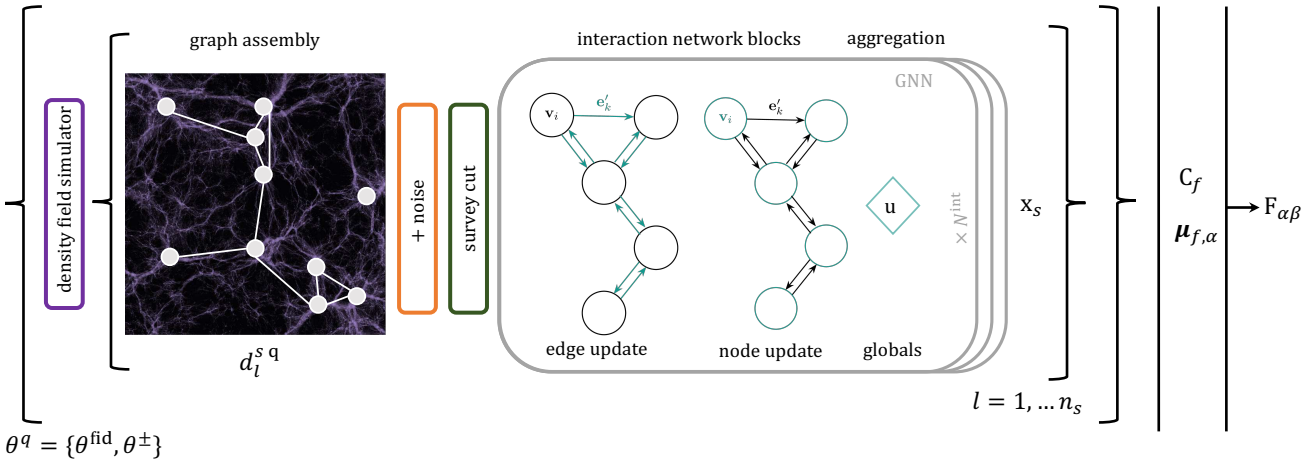


Figure 7.2: Cartoon of the graph-based information maximising neural network scheme. For each $i = 1, \ldots n_s$ dark matter field realization, halo catalogues are computed from a density field and assembled into a connected graph. A GNN block then computes edge and node updates (green) outlined in Section 7.3.1, pooling graph attributes to compute global summaries, $\mathbf{x} = \mathbf{u}$. This process is repeated for simulations at the fiducial parameter values, as well as at $\boldsymbol{\theta}^{\pm}$ for numerical derivative calculation via equation (7.10). The output of the IMNN is the Fisher information matrix, computed via equation (7.8). The network is trained via gradient back-propagation, with the $\det \mathbf{F}$ and $\mathbf{C}_f$ contributing to the scalar loss function.

The graph framework allows for a modular study of the cosmological information embedded in large-scale structure. We next review IMNNs as a tool for information extraction, as well as optimal compression for graphs assembled from cosmological surveys. The IMNN framework is presented in full in Charnock et al. (2018) with developmental updates discussed in Makinen et al. (2021), but we review the formalism here for completeness and introduce new aspects to the technique. The sharper the peak of an informative likelihood function $\mathcal{L}(\mathbf{d}|\boldsymbol{\theta})$ for some fixed data $\mathbf{d}$ with $n_{\mathbf{d}}$ data points and $n_{\boldsymbol{\theta}}$ parameters at a given value of $\boldsymbol{\theta}$, the more informative $\boldsymbol{\theta}$ is about the data. The Fisher information matrix describes how much information $\mathbf{d}$ contains about the parameters, and is given

as the second moment of the score of the likelihood

$$\mathbf{F}_{\alpha\beta} = \int d\mathbf{d}\ \mathcal{L}(\mathbf{d}|\boldsymbol{\theta}) \frac{\partial \ln \mathcal{L}(\mathbf{d}|\boldsymbol{\theta})}{\partial \theta_\alpha} \frac{\partial \ln \mathcal{L}(\mathbf{d}|\boldsymbol{\theta})}{\partial \theta_\beta}, \tag{7.2}$$

and can be written as

$$\mathbf{F}_{\alpha\beta} = -\left\langle \frac{\partial^2 \ln \mathcal{L}}{\partial \theta_\alpha \partial \theta_\beta} \right\rangle \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\rm fid}}, \tag{7.3}$$

evaluated at some fixed fiducial parameters. A large Fisher information for a set of data indicates that the data is very informative about the model parameters attributed to it. Fisher forecasting for a given model is made possible by the information inequality and the Cramér-Rao bound (Cramér, 1946; Rao, 1945), which states that the minimum variance of the value of an estimator $\boldsymbol{\theta}$ is given by

$$\langle (\theta_\alpha - \langle \theta_\alpha \rangle)(\theta_\beta - \langle \theta_\beta \rangle) \rangle \geq \mathbf{F}^{-1}_{\alpha\beta}. \tag{7.4}$$

We will write the compression as a function $f : \mathbf{d} \to \mathbf{x}$. For large datasets, data compression is essential for inference to avoid the curse of dimensionality. The MOPED formalism (Heavens et al., 2000) gives optimal score compression for cases where the likelihood and sampling distributions are exactly Gaussian.

IMNNs are neural networks that perform data compression and compute the Fisher information of a data set. Such compression is possible even if the data likelihood is unknown or intractable, simply based on having simulations of the data at a given fiducial parameter point and local information about how the parameters change the data distribution. It can be shown (see Section 4.4.1) that the optimality of the IMNN summaries holds for any unknown or intractable data likelihood even though the IMNN maximizes Fisher information assuming the parameter-independent covariance form of the Gaussian likelihood for the IMNN summaries

$$-2 \ln \mathcal{L}(\mathbf{x}|\mathbf{d}) = (\mathbf{x} - \boldsymbol{\mu}_f(\boldsymbol{\theta}))^T \mathbf{C}_f^{-1} (\mathbf{x} - \boldsymbol{\mu}_f(\boldsymbol{\theta})) \tag{7.5}$$

where

$$\boldsymbol{\mu}_f(\boldsymbol{\theta}) = \frac{1}{n_s} \sum_{i=1}^{n_s} \mathbf{x}_i^s \tag{7.6}$$

is the mean of the compressed summaries $\mathbf{x}_i^s$, with $\{\mathbf{x}_i^s | i \in [1, n_s]\}$' and we assume a parameter-independent covariance matrix. Here i indexes the random initialisation of n_s simulations, and the

superscript s denotes quantities derived from simulations, unlike quantities without the superscript s which are derived from actual observations. The summaries are obtained via simulation of data $\mathbf{d}_i^s = \mathbf{d}_i^s(\boldsymbol{\theta}, i)$ via the compression scheme $f : \mathbf{d}_i^s \to \mathbf{x}_i^s$. The covariance of the summaries is computed from the data as well:

$$(\mathbf{C}_f)_{\alpha\beta} = \frac{1}{n_s - 1}\sum_{i=1}^{n_s}(\mathbf{x}_i^s - \boldsymbol{\mu}_f)_\alpha(\mathbf{x}_i^s - \boldsymbol{\mu}_f)_\beta. \tag{7.7}$$

Note that this covariance is assumed to be independent of the parameters, which, whilst not strictly true, is enforced by regularisation during the fitting of the IMNN. A Fisher matrix can then be computed from the likelihood in equation (7.5):

$$\mathbf{F}_{\alpha\beta} = \mathrm{tr}[\boldsymbol{\mu}_{f,\alpha}^T\mathbf{C}_f^{-1}\boldsymbol{\mu}_{f,\beta}], \tag{7.8}$$

where we introduce the notation $\boldsymbol{y}_{,\alpha} \equiv \partial\boldsymbol{y}/\partial\theta_\alpha$ for partial derivatives with respect to parameters. If the compression function f is a neural network parameterized by layer weights $\mathbf{w}^\ell$ and biases $\mathbf{b}^\ell$ (with ℓ the layer index), the summaries (and respective mean and covariance) then become functions of these new parameters $\mathbf{x}(\boldsymbol{\theta}) \to \mathbf{x}(\boldsymbol{\theta}, \mathbf{w}^\ell, \mathbf{b}^\ell)$. To evaluate equation (7.8) for a neural compression, we must compute

$$\boldsymbol{\mu}_{f,\alpha} = \frac{\partial}{\partial\theta_\alpha}\frac{1}{n_s}\sum_{i=1}^{n_s}\mathbf{x}_i^{s\ \mathrm{fid}}. \tag{7.9}$$

One way of computing the derivatives of the summary means with respect to the parameters is to define a finite difference gradient dataset by altering simulation fiducial values by a small amount, yielding

$$\left(\frac{\partial\hat{\mu}_i}{\partial\theta_\alpha}\right)^{s\ \mathrm{fid}} \approx \frac{1}{n_s}\sum_{i=1}^{n_s}\frac{\mathbf{x}_i^{s\ \mathrm{fid}+} - \mathbf{x}_i^{s\ \mathrm{fid}-}}{\Delta\theta_\alpha^+ - \Delta\theta_\alpha^-}. \tag{7.10}$$

To prevent extra information being extracted from accidental correlation in limited sized data sets, reported statistics need to be computed on a validation set of simulations, which is unlikely to share the same accidental correlations as the fixed training set. An alternative explored in Makinen et al. (2021) is to calculate the adjoint gradient of the simulations as well as the derivatives of the network parameters with respect to the simulations:

$$\boldsymbol{\mu}_{f,\alpha} = \frac{1}{n_s}\sum_{i=1}^{n_s}\left(\frac{\partial\mathbf{x}}{\partial\theta_\alpha}\right)_i^{s\ \mathrm{fid}} = \frac{1}{n_s}\sum_{i=1}^{n_s}\sum_{k=1}^{n_d}\frac{\partial\mathbf{x}_i^{s\ \mathrm{fid}}}{\partial d_k}\frac{\partial\mathbf{d}_i^{s\ \mathrm{fid}}}{\partial\theta_\alpha}. \tag{7.11}$$

If the gradient of the simulations can be computed efficiently, this technique for computing the compression Fisher information eliminates the need for hyperparameter tuning of the finite difference

derivative size, $\Delta\theta_\alpha$.

The network is trained to maximise the logarithm of the determinant of the Fisher information, computed via equation (7.8). As described in Charnock et al. and Livet et al., the Fisher information is invariant to nonsingular linear transformations of the summaries. To remove this ambiguity, a term driving covariance to the identity matrix is added

$$\Lambda_C = \frac{1}{2}\left(||(\mathbf{C}_f - \mathbf{1})||^2_{\mathcal{F}} + \left|\left|(\mathbf{C}_f^{-1} - \mathbf{1})\right|\right|^2_{\mathcal{F}}\right), \tag{7.12}$$

where $||\boldsymbol{A}||_{\mathcal{F}} \equiv \sqrt{\mathrm{tr}\,\boldsymbol{A}\boldsymbol{A}^T}$ denotes the Frobenius norm. This yields the loss function

$$\Lambda = -\ln\det\mathbf{F} + r_{\Lambda_C}\Lambda_C, \tag{7.13}$$

with regularization parameter

$$r_{\Lambda_C} = \frac{\lambda\Lambda_C}{\Lambda_C + \exp(-\alpha\Lambda_C)}, \tag{7.14}$$

where λ and α are user-defined parameters. When the covariance is far from identity, the r_{Λ_C} function is large and the optimization focuses on bringing the covariance and its inverse back to identity. The network is trained until the Fisher information stops increasing for a pre-determined number of iterations. We stress that the value of $\mathbf{F}$ reported as an information metric, however, is the one computed via Eq. 7.8, computed over a validation set of simulations in the case of a finite set of data.

To summarise the IMNN algorithm we take the following steps *every training epoch* to optimise the Fisher information:

i) compress simulations at the fiducial model with different random seeds to the network. Calculate the covariance of these summaries using equation (9.13).

ii) compress simulations generated at perturbed fiducial parameter values, $\boldsymbol{\theta}^{\pm}$ to produce $\mathbf{x}^{\pm}$. Calculate the derivatives $\mathbf{x}_{f,\alpha}$ with equation (7.10).

iii) Calculate the Fisher matrix (Eq. (7.8)). Pass the Fisher and covariance matrices to the loss function. Update neural network weights using gradient descent such that $\det\mathbf{F}$ increases.

7.3.1 Graph Neural Networks

A graph neural network (GNN) block typically consists of three update functions, $\boldsymbol{\phi} = (\phi_u, \phi_v, \phi_e)$, and three aggregation functions, $\boldsymbol{\rho} = (\rho^u, \rho^v, \rho^e)$, applied sequentially to a graph tuple $G = (\mathbf{u}, V, E)$. A single graph block ℓ is comprised of several update steps to its elements:

1. *Edge update*: Each edge is parameterized by a function $\phi_e^{\ell+1}$ which takes as inputs its connected nodes, previous value, and graph global properties and yields another edge:

$$\mathbf{e}_{ij}^{\ell+1} = \phi_e^{\ell+1}(\mathbf{v}_i^\ell, \mathbf{v}_j^\ell, \mathbf{e}_{ij}^\ell \mathbf{u}^\ell), \tag{7.15}$$

   where $\mathbf{v}_i^\ell$ and $\mathbf{v}_j^\ell$ are sender and receiver nodes indexed by (s_k, r_k).

2. *Node update*: Each node is then parameterized by a function $\phi_v^{\ell+1}$ and outputs a new node:

$$\mathbf{v}_i^{\ell+1} = \phi_v^{\ell+1}\left(\rho^{e\to v}(E_i^{\ell+1}), \mathbf{v}_i^\ell, \mathbf{u}^\ell\right), \tag{7.16}$$

   Here a permutation-invariant aggregation operation $\rho^{e\to v}(E_i^{\ell+1})$ pools the neighbourhood of edges $E_i^{\ell+1}$ connected to node i into a fixed-sized vector to feed into the update function.

3. *Global update*: The global features of the graph are then updated with a function $\phi_u^{\ell+1}$:

$$\mathbf{u}^{\ell+1} = \phi^u\left(\rho^{e\to v}(E^{\ell+1}), \rho^{v\to u}(V^{\ell+1}), \mathbf{u}^\ell\right), \tag{7.17}$$

   where the graph's edge ($E^{\ell+1}$) and node ($V^{\ell+1}$) sets are pooled into fixed-sized vectors for the global update.

The order of operations of these updates is flexible, but usually applied in the order displayed above, and in the GNN block in Fig. 7.2. This framework allows $\boldsymbol{\phi}$ functions to be arbitrarily parameterized as neural networks with nonlinear activation functions. Aggregation functions $\boldsymbol{\rho}$ must be allowed to take a variable number of arguments, so are usually chosen to be permutation-invariant operators such as the mean, summation, or maximum (Bronstein et al., 2021). Stacking $\ell = 1 : N^{\text{int}}$ GNN blocks allows node information to be propagated to and from neighbours N^{int} degrees away, where int refers to *interactions*. In this work all GNN blocks operate over the entire graph. However, one could also devise surrogate GNN blocks that operate on small scales and then pass information up to

larger scales via an aggregation function $\rho^{\text{small}\to\text{large}}$, such that one GNN network is not responsible for operating on nodes of all scales in a densely-populated graph. We detail our specific implementation and architecture in Section 7.4.1.

The GNN framework is readily incorporated into the IMNN formalism, since the details of the neural network architecture only serve to better capture how the data changes with the parameters. Instead of predicting an output graph or class label, as in Battaglia et al. (2018), our final global update ϕ^u outputs IMNN summaries, $\mathbf{x} = \mathbf{u}^{\ell=N^{\text{int}}}$. This new aspect to the IMNN formalism is the ability to operate over variable-length data inputs, rendering the cardinality of input graphs, $n_{\text{data}} = N^v$ and N^e informative features of the data. A stochastic system might yield a different number of discrete particles for different parameters, meaning the *number* of data becomes a descriptor of the statistical model. This allows for a study of information $\mathcal{I} = \frac{1}{2}\ln\det F$ as a function of N^v and enables much more flexible data modelling.

7.4 Cosmological Parameter Inference with Halo catalogues

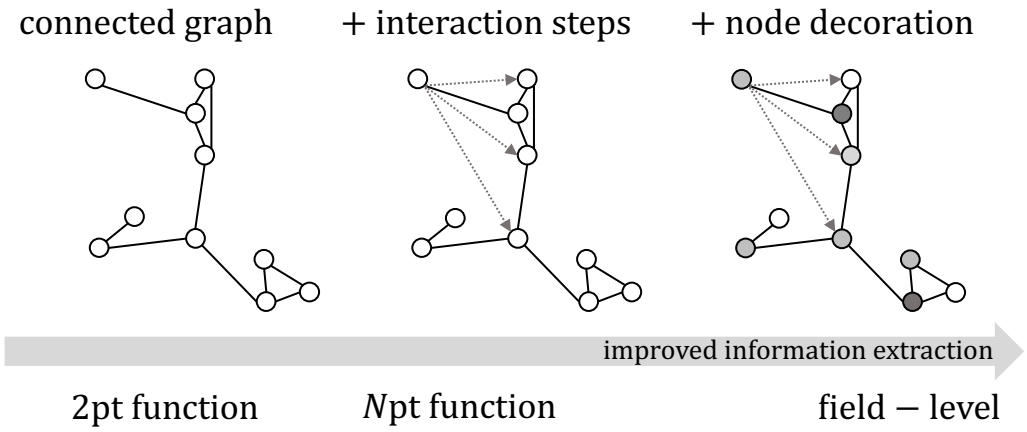


Figure 7.3: Information as a function of large-scale structure data representation. Cosmological parameter information extraction efficiency increases as information is added and propagated throughout the halo graph with increased connections and message-passing steps. The second and third graph show message propagation from the top left node after $N^{\text{int}} = 2$ update steps.

Here we consider applications of our graph IMNNs on realistic cosmological problems. Even future astronomical studies will not be able to image complete dark matter overdensity fields of large-scale structure. However, discrete galaxy and void catalogues can be assembled as tracers of structure. Different LSS realizations from stochastic initial conditions will have different numbers of halos and

voids, posing a problem for usual fixed-size neural networks, (which are themselves problematic for inference unless treated correctly). We explore information extraction as a function of graph connectivity in the context of two-parameter inference for the matter density parameter, Ω_m, as well as σ_8, the r.m.s. fluctuation of density perturbations at the 8 h^{-1} Mpc scale. Both Ω_m and σ_8 parameterize the distribution of matter in cosmological simulations, so the graph topology should be sensitive to changes in parameters.

The more descriptive a graph is, the more information one intuitively expects to extract. We demonstrate this trend by first considering *undecorated* graphs, annotated with just positions of and relative distances between halos. We show that information extraction efficiency increases as graph connectivity increases. We then show that information increases further when halo masses are included as node features.

The closest existing statistics to this representation are n-point correlation and mass functions, and we show how information increases beyond the 2-point correlation function with the same catalogue as graph connectivity is increased, as illustrated in the cartoon in Figure 7.3.

7.4.1 Halo Catalogues

Here we describe the simulated catalogues that are used for training and validation.The Quijote Halo catalogues are assembled from 3D overdensity fields at the present day ($z = 0$) using the Friends of Friends (FoF) algorithm (Davis et al., 1985). Attributes computed by the finder are halo masses M_i, positions $\boldsymbol{p}_i$, and velocities $\boldsymbol{v}_i$. Each full simulation yields a catalogue of ∼400,000 halos on average. Here we restrict our analysis to mass and clustering information in an effort to compare our method to known statistics.

Graph inputs assembly

We initially connect graphs of a manageable size by varying two hyperparameters. We first make a minimum mass cut $M_{\rm cut}$ to be considered in the catalogue. Nodes are then connected to one another within a Euclidean distance $r_{\rm connect}$. We initially explore the noise-free limit with known masses and fix $M_{\rm cut} = 1.5 \times 10^{15} M_\odot$ to assess the pure information limit of the catalogues. We add noise to the

analysis in Section 7.6. This cut yields $N^v \in [70, 140]$ halos per catalogue. We visualize two graphs in Appendix 7.11, figure 7.2.

We then assemble the truncated catalogues into non-invariant and invariant graphs, as outlined in Section 7.2.2. We initialize each graph's global property with a tuple $\mathbf{u}^{\ell=0} = (\text{arcsinh}\, N^v, \text{arcsinh}\, N^e)$ summarising the cardinality of the graphs. As described in Battaglia et al. (2018); Lemos et al. (2022); Villanueva-Domingo & Villaescusa-Navarro (2022), imposing symmetries in data representation can improve GNN training, since the network can focus on learning relevant correlations to the problem, as opposed to re-learning symmetry. We test this notion in the context of information extraction.

Graph neural network architecture

We choose to parameterize our GNN functions with simple fully-connected networks. Each $\boldsymbol{\phi}$ function is a dense network with two layers of 50 hidden neurons and `gelu` activations (Hendrycks & Gimpel, 2016). We built a custom aggregation function akin to that found in Villanueva-Domingo & Villaescusa-Navarro (2022), in which mean, max, sum, and variance are computed over node and edge attributes and then concatenated, since it is not known a priori which function is most useful for information extraction. To aggregate e.g. the set of edges E_i in a neighbourhood around node i we compute:

$$\bigoplus_{j \in E_i} \mathbf{e}_{ij} = \left[\max_{j \in E_i} \mathbf{e}_{ij}, \sum_{j \in E_i} \mathbf{e}_{ij}, \frac{\sum_{j \in E_i} \mathbf{e}_{ij}}{\sum_{j \in E_i}} \right] \tag{7.18}$$

We additionally modify these operators with a trainable arcsinh layer e.g. for edge-to-node aggregation:

$$\rho^{\ell+1}_{e \to v}(E^\ell_i) = a\, \text{arcsinh} \left(b \bigoplus_{j \in E^\ell_i} \mathbf{e}^\ell_{ij} + c \right) + d, \tag{7.19}$$

where (a, b, c, d) are scalar learnable parameters initialized as $(1, 1, 0, 0)$ to ensure numerical stability for gradient calculation. All networks are trained with an Adam optimizer with a learning rate set to 0.0001 and coupling parameters $\lambda = 10$ and $\alpha = 0.95$. We construct our graphs and GNNs using the `jraph` (Godwin et al., 2020) and `Flax` (Heek et al., 2020) libraries, which are both `Jax`-compatible.

We train our gIMNNs by splitting the *Quijote* simulations into equally-sized training and validation sets. Gradient descent is performed on training data, while reported compression statistics (Fisher information) are computed for the *validation* set using equation (7.8). Both training and validation sets comprise of $n_s = 500$ fiducial simulations at $\boldsymbol{\theta}_{\text{fid}} = (\Omega_m, \sigma_8) = (0.3175, 0.834)$ and $n_d = 250$

seed-matched derivative simulations perturbed by $\delta\boldsymbol{\theta} = (0.01, 0.015)$, yielding $n_d \times 2 \times 2 = 1000$ simulations (see Villaescusa-Navarro et al. (2020b) for details). Training on the loss defined in Eq 9.14 is performed until a patience criterion is met, in this case, when the training Fisher information stops increasing significantly for 1000 epochs.

7.4.2 Undecorated Graphs vs. n-point Statistics

We first consider an undecorated graph representation of halo catalogues *without* descriptive node features. Drawing more edges between nodes increases the connectivity of the graph, allowing information from a single node to reach more distant neighbours. Undecorated graphs of increasing connectivity are analogous to traditional n-point statistics computed for galaxy catalogues. 3-point statistics for example consider triangular groupings of galaxies, and generally offer tighter constraints from large-scale structure data than the 2PCF, as shown in (Hahn et al., 2020). We additionally explore invariant and non-invariant graph structures, outlined in Section 7.2.2.

Comparison to 2-point correlation information

As a benchmark for our analysis, we also compare the information content obtained from the 2-point correlation function (2PCF), $\xi(r)$, of our small halo catalogues, the real-space equivalent to the *Quijote* power spectrum computed in Villaescusa-Navarro et al. (2020b). For a statistic $Q = \xi(r)$, the Fisher information is given by (Tegmark et al., 1997):

$$F_{ij} = \frac{1}{2}\mathrm{tr}\left\{\boldsymbol{C}^{-1}\left[\left(\frac{\partial Q}{\partial\theta_i}\frac{\partial Q}{\partial\theta_j}^T\right) + \left(\frac{\partial Q}{\partial\theta_i}^T\frac{\partial Q}{\partial\theta_j}\right)\right]\right\}, \tag{7.20}$$

where $\boldsymbol{C}$ is estimated from simulations at the fiducial and the derivatives are approximated numerically via

$$\frac{\partial Q}{\partial\theta_i} \approx \frac{Q(\theta_i^+) - Q(\theta_i^-)}{\theta_i^+ - \theta_i^-} \tag{7.21}$$

We use the full suite of 500 derivative and 1000 fiducial halo catalogue simulations to compute Eq 7.20, and crucially *make the same conservative mass cut*. We bin distances into 10 fixed bins between 0 and $\sqrt[3]{3}$ Gpc^3, yielding a covariance matrix of size 100^2. For the 2PCF we obtain a Fisher information of 2.28×10^6, or Shannon entropy of 7.319 nats.

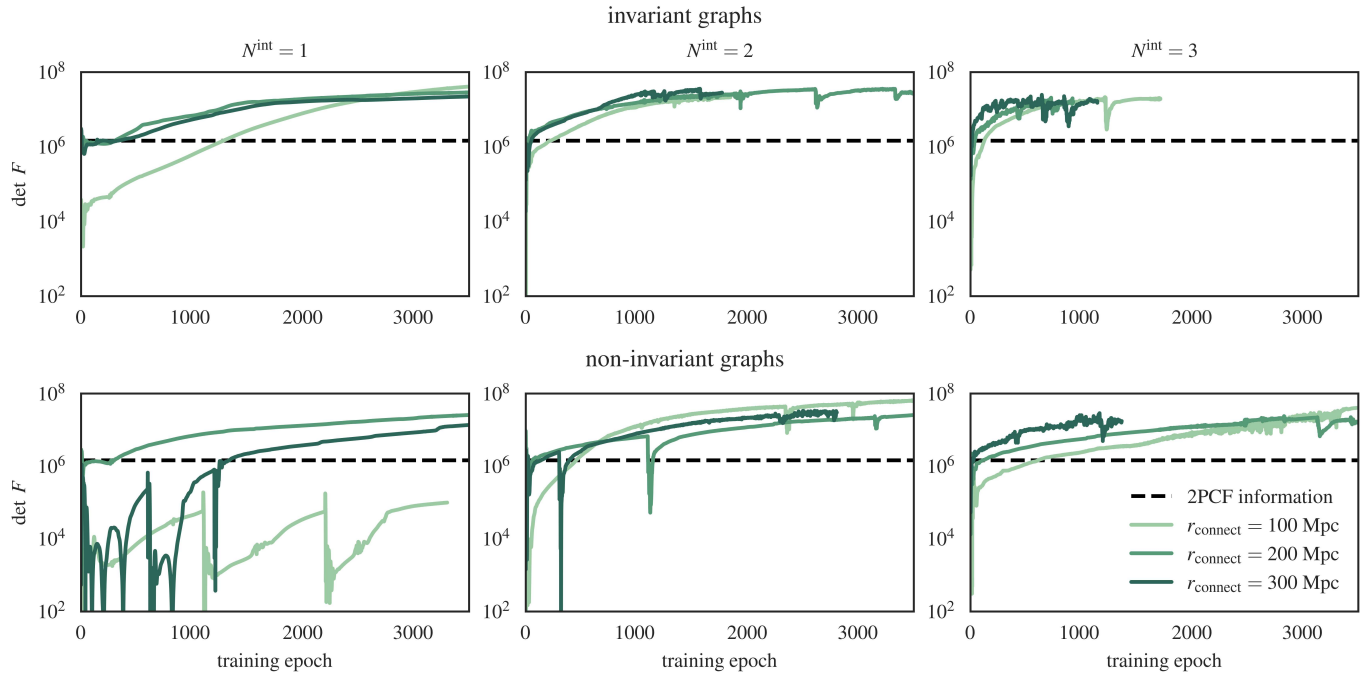


Figure 7.4: Information extraction comparison for undecorated invariant (*top row*) and non-invariant (*bottom row*) halo graphs over validation simulations. Columns indicate how many graph network update steps are performed, and colours indicate graphs assembled with different physical connection radii. The information obtained from the 2PCF computed from the same catalogues with the same mass cut is shown as a dashed line. For sufficiently descriptive networks ($N^{\rm int} > 1$), invariantly-assembled graphs train faster, but all graphs connected for $r > 100$ Mpc saturate at $\det F \approx 2.8 \times 10^7$, or ≈ 15 times higher than the 2PCF. Jagged dips in validation curves indicate points of restarting network training after patience criteria were met.

Increasing graph connectivity

Here we construct graphs of varying edge cardinality by varying a physical connection parameter, $r_{\rm connect} \in \{100, 200, 300\}$ Mpc, yielding graphs with average edge number $\langle N^e \rangle \in \{27, 150, 437\}$ respectively. We also compare information as a function of increasing GNN interaction blocks, $N^{\rm int}$, for all $r_{\rm connect}$, for both non-invariant and invariantly-structured halo graphs. Network architecture is identical for each $r_{\rm connect}$ value, and initialized by the same random seed.

Results. We display information extraction as a function of graph connectivity in Figure 7.4, computed for the validation simulation set. We also display the catalogue's 2PCF information (dashed line) as a benchmark. Invariant graphs (top row) train more smoothly than non-invariant graphs (bottom row) since the network does not have to learn relationships from position values on the nodes. A single GNN block struggles to extract information with $r_{\rm connect} = 100$ Mpc in the non-invariant representation (lower left), but plateaus at $\approx 2.8 \times 10^7$ for all other configurations. This behavior is

likely because in most cases the network is both descriptive enough and is able to capture patterns at much larger scales by attending to halos higher degrees away. The common saturation value across multiple network and connectivity combinations indicates that undecorated graphs typically contain $\det F_{\rm IMNN}/\det F_{\rm 2PCF} \approx 10-15$ times more information than the 2PCF, regardless of connectedness, provided a descriptive enough network can extract it. The graph representation improves marginal constraints in Ω_m by a factor of ≈ 2 and in σ_8 by a factor of ≈ 11, displayed in Figure 7.5.

We initially hypothesized that increasing network complexity would increase the information extraction. However, we found that we obtain essentially equivalent information for any combination of $r_{\rm connect}$ and $N^{\rm int}$. It is clear that for this small number of halos the information is easily saturated at any level, but with more halos in the catalogue, as discussed in Section 7.8, hierarchical clustering at the graph or network level might pull out more information from e.g. smaller mass scales. This exploration is reserved for a future work.

Since all sufficiently-connected representations obtain the same information, we proceed in our experiments with invariant graphs connected with $r_{\rm connect} = 200$ Mpc and $N^{\rm int} = 2$, since this combination resulted in the smoothest and fastest training (4 minutes) on a single NVIDIA-v100 GPU.

7.4.3 Decorated Graphs: Incorporating Halo Mass

We next decorate each halo node with the corresponding (noise-free) halo mass. We widen the network's hidden dimension to 64 and train both decorated and undecorated graphs with the same patience settings. Training was restarted for each three times after plateau to ensure saturation. The same network architecture is able to extract 2.3 times more information when decorated with masses, corresponding to 42 times more information than the 2PCF. We display the corresponding Fisher ellipses in Figure 7.5, along with isomass lines of the halo mass function, described in Section 7.5.1. We also decorated graphs with peculiar velocities with slight improvement in information extraction but restricted our analysis to mass and clustering for interpretability.

7.5 Mass cut information

We next investigate how much information is contained in the mass cut, $M_{\rm cut}$ which determines halo number N^v. We compare two graph assemblies: one with a *fixed* number of the most massive

halos $N^v_{\rm fixed} = 105$, i.e. the average number of halos across variable sized halo catalogues with fixed mass cut $M_{\rm cut} = 1.5 \times 10^{15} M_\odot$, and another where the cardinality is allowed to vary with $M_{\rm cut}$. For each case we compare decorated and undecorated graphs, and the network (epistemic) and data sampling (aleatoric) errors associated with each representation. To estimate network variability we

catalogue N^v	graph assembly	$\ln\det F$	vary network	**vary data**
fixed	without mass		5.03 ± 0.47	$\mathbf{5.98 \pm 1.06}$
	with mass		12.43 ± 1.44	$\mathbf{12.39 \pm 0.22}$
	2PCF	9.74		
variable	without mass		17.89 ± 0.33	$\mathbf{17.66 \pm 0.27}$
	with mass		17.40 ± 0.57	$\mathbf{17.85 \pm 0.12}$
	2PCF	14.19		

Table 7.1: Comparison of extracted information from fixed- and variable-length catalogues. Information was extracted with $r_{\rm connect} = 200$ Mpc, and networks with $N^{\rm int} = 2$ across 5 identical initialisations. We display each configuration's best Fisher information and the means and standard deviations over 5 network initializations (second-to-last column) and 5 different train/validation set splits (last column).

train five gIMNNs with $N^{\rm int} = 2$ with different initialization of network parameters, whilst fixing the training and validation sets, displaying the best Fisher obtained as well as the mean and standard deviation of $\ln\det F$ over the five runs. For data sampling uncertainty we fix the gIMNN weight values on initialization and train on five different randomised train-validation equal-sized splits of the available simulations. For both data and network error cases, the same five random seeds and network architecture is used across all data configurations.

Results. We display results in Table 7.1. Fixing catalogue size eliminates halo number as a useful feature to the network, evidenced by much lower information yields. Without mass decoration there is less information in the graph data so the data can be fit by more possible functions by the network, so the variability of the Fisher as a function of the data sampling is increased over the decorated case. However, the network has to fit a simpler compression since there are fewer relevant features without mass, so the variability in possible network weights decreases, compared to the decorated case. Fixed-length graphs do not exceed the 2PCF information until annotated with mass information.

When catalogues are allowed to vary with a physical mass cut, much more information can be extracted from both decorated and undecorated graphs. Including mass information on the nodes again increases the variability incurred across different network initializations, but decreases the aleatoric uncertainty since we better describe the likelihood with more information.

7.5.1 Comparison to the Halo Mass Function

The results of Section 7.5 indicate that catalogue information extraction is extremely sensitive to a physical mass cut. This behaviour is akin to constraints obtained using halo mass cumulative distribution functions (Reed et al., 2006; **?**; Artis et al., 2021). The halo number density function is dn/dM, defined as the number of halos of mass M per unit volume per unit interval in M, equivalently parameterized using the smoothed r.m.s. linear overdensity of the density field, $\sigma(M)$, via the halo mass function (HMF), $f(\sigma)$. The fraction of mass in collapsed halos per unit interval $\ln \sigma^{-1}$ obeys

$$\int_{-\infty}^{\infty} f(\sigma) d \ln \sigma^{-1} = 1, \tag{7.22}$$

and is related to the halo number density function via

$$\frac{dn}{dM} = \frac{\rho_o}{M} \frac{d \ln \sigma^{-1}}{dM} f(\sigma), \tag{7.23}$$

where ρ_o is the mean mass density of the universe. The form of $f(\sigma; \boldsymbol{\theta})$ can be related analytically to cosmological parameters, such as in Press & Schechter (1974), or approximated using simulations (Reed et al., 2006).

We compare gIMNN Fisher constraints to isomass contours of the integrated Press-Schechter HMF, $f_{\rm PS}(\sigma;\ \Omega_m, \sigma_8)$, integrated from a fixed $M_{\rm cut}$ as a function of cosmological parameters in Figure 7.5. We use the HMF since this quantity incorporates both halo number and mass information. See Appendix 7.10 for a detailed comparison of dn/dM and $f(\sigma)$ functions. We utilize `hmf calc` (Murray et al., 2013; Murray, 2014) for the calculation. The HMF and the corresponding halo number density at fixed $M = M_{\rm cut}$ determines a relatively narrow locus in the (Ω_m, σ_8) plane. The clustering information, traced by the 2PCF Fisher, (which is accessible by the network) serves to lift this degeneracy. The network Fishers are nearly parallel to the HMF isomass contours, but are not degenerate in the direction of the 2PCF Fisher's major axes. Decorating nodes with masses also induces a slight rotation towards the isomass contours, since the network has more detailed mass information to work with. This result indicates that *the network has automatically learned to extract information from both halo clustering and mass information.*

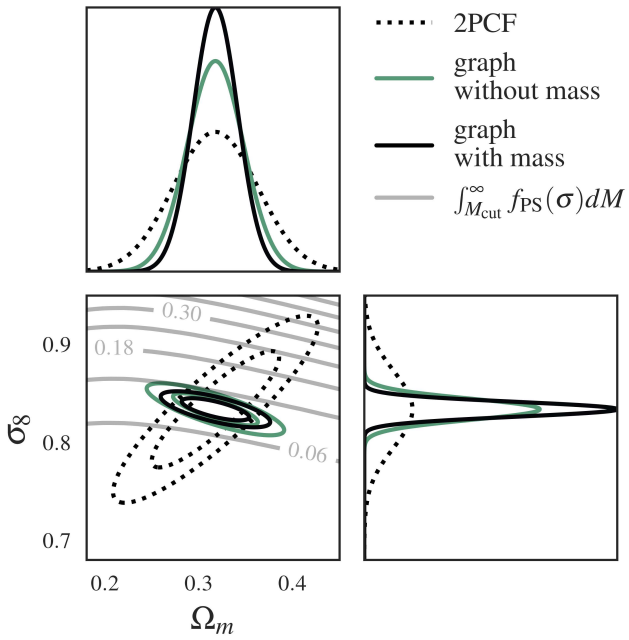


Figure 7.5: Fisher matrix comparison for decorated (black) and undecorated (green) graphs and 2PCF (dashed) computed for the same catalogue, plotted over contours of the Press-Schechter halo mass function (grey), where the numbers indicate the integrated mass density of halos from $M_{\rm cut}$ to infinity, as a function of Ω_m and σ_8. The HMF contours indicate the information provided by a fixed physical mass cut, orthogonal to the positional information in the 2PFC. The IMNN Fisher ellipses follow these contours, but are not degenerate in the direction of the 2PCF, indicating that the IMNN has learned to use both mass and clustering information. Graphs annotated with halo mass tighten constraints by a factor of 2.3 over undecorated graphs, and by a factor of 42 over 2PCF information.

7.6 Working with Noisy catalogues

We next add observational noise and catalogue cuts on-the-fly during gIMNN training to mimic survey assembly with imperfect observations. Before a graph is constructed from a halo catalogue and fed to the network in training, halo masses are subjected to white noise with fixed variance,

$$\hat{m}_i = m_i + \mathcal{N}(0, \sigma^2_{\rm noise}), \tag{7.24}$$

where $\sigma_{\rm noise} = A_{\rm noise} M_{\rm cut}$. Observed halos that fall below $M_{\rm cut}$ are then trimmed from the graph to mimic real catalogue cuts in the presence of noisy mass estimates. This noise model reflects uncertainty in the halo finder or galaxy catalogue builder. Smaller masses close to $M_{\rm cut}$ are more likely to be cut due to mass underestimation, similar to low-brightness clusters in sky surveys. We choose $M_{\rm cut} = 1.5 \times 10^{15} M_\odot$ and train identical networks with different amplitudes of on-the-fly noise; $A_{\rm noise} \in \{0.05, 0.1, 0.2\}$.

Results. We display validation curves over training epoch in Figure 7.6. Increasing catalogue noise results in higher variance per epoch in the computed Fisher statistics, as well as a slightly lower information plateau. This can be interpreted as higher noise obscuring small mass scales in the information extraction. This effect is illustrated via inflated Fisher constraints in Figure 7.7.

As the noise level increases the low-end masses have more variance when drawn on-the-fly so more halos are projected out of the catalogue because they fall below $M_{\rm cut}$, and so this information cannot be encapsulated in the compressed summaries. Increasing the noise amplitude to 20% of $M_{\rm cut}$ inflates constraints in Ω_m. In the high-noise limit, halo positions dominate $\det F$, indicated by the relatively unchanged, position-dependent σ_8 constraints. As noise decreases and masses are better known, the Fisher exhibits the same rotation seen in Section 7.5.1 along the isomass HMF lines. This effect is discussed in detail in Appendix 7.10.

Despite inflating constraints, showing the network large numbers of on-the-fly noise realizations during training can harden the network to the negative effects of limited training data and therefore provide smoother training whilst still bein g able to extract information at a similar level to noise-free catalogues. However, the model of these noisy masses must be accurate to the noise model expected for the real data otherwise the on-the-fly simulations do not provide hardening of the summaries in the correct way and may even project out informative data correlations.

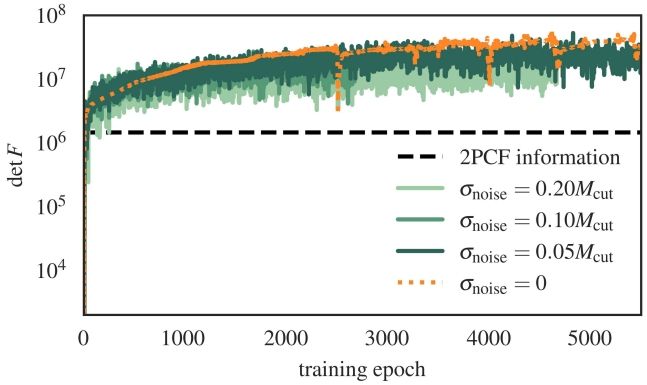


Figure 7.6: Validation curves for noisy masses. Smaller noise variance (darker curves) results in smaller per-epoch variance in $\det F$ and slightly more information extraction. Information leakage occurs with higher noise variance since smaller scales are poorly resolved and trimmed from the catalogue.

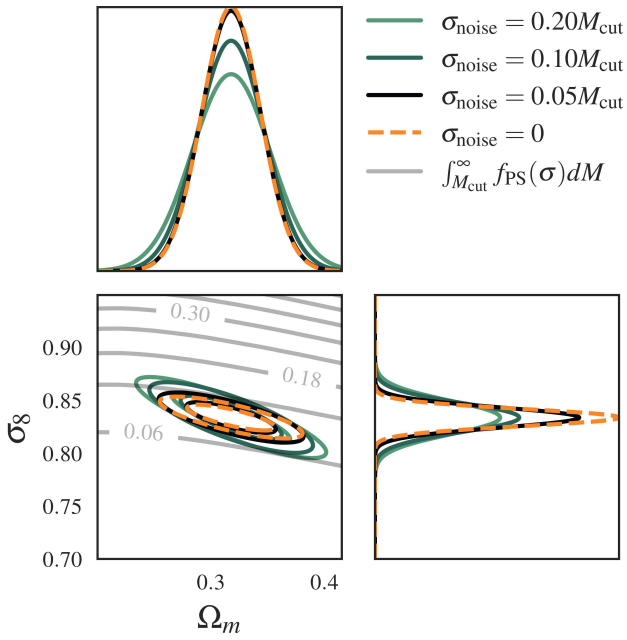


Figure 7.7: Fisher constraints for different noise models, plotted over lines of the Press-Schechter halo mass function integrated from $M_{\rm cut}$ (grey), where the numbers indicate the integrated mass density of halos from $M_{\rm cut}$ to infinity, as a function of Ω_m and σ_8. Higher $\sigma_{\rm noise}$ (lighter curves) cause low-mass halos to drop out, inflating constraints in Ω_m, but constraints in σ_8 remain relatively unchanged since this parameter is largely position-dependent.

7.7 Application to Implicit Likelihood Inference

The IMNN framework is both an information quantification scheme as well as an asymptotically optimal compression mechanism for implicit likelihood inference. The global network summaries used to compute Fisher statistics can also be used as proxies for the cosmological parameters via a score estimate (Alsing & Wandelt, 2018; Charnock et al., 2018) using the IMNN Fisher and covariance:

$$\hat{\theta}_\alpha = \mathbf{F}^{-1}_{\alpha\beta}\frac{\partial \mu_i}{\partial \theta_\beta}\mathbf{C}^{-1}_{ij}\left(x_j(\mathbf{w},\mathbf{d}) - \mu_j\right).) \tag{7.25}$$

These summaries are *not* explicit predictions for cosmological parameters, although they are pseudo-maximum likelihood estimates for the parameters in the region asymptotically close to the fiducial cosmological parameter values. Instead, we suggest using these values as informative summaries in Approximate Bayesian Computation (ABC) or density estimation schemes, as demonstrated in Makinen et al. (2021) and Charnock et al. (2018). Figure 9.5 shows a spread of these summaries over 200 test (non-seed matched with cosmic variance) $\boldsymbol{\theta}^{\pm}$ *Quijote* datasets. The network is able to distinguish halo graphs simulated at different parameter values (on the level of the parameter degeneracy), rendering these gIMNN summaries usable in accept-reject simulation-based inference (SBI) schemes. We additionally discuss network generalization to other data in Appendix 7.11.

7.8 Discussion & Conclusion

In this study, we explored cosmological information extraction from halo catalogues assembled as graphs. We first introduced graphs as a general language for describing large-scale structure formation. We showed that nonlinear summaries from sufficiently expressive graph neural networks far exceed the information contained in traditional 2-point statistics, using the ≈100 most massive halos. We illustrated that decorating graphs with mass information increases the information yield, as well as decreases training-validation set sampling variance (aleatoric error). Finally, we showed that summaries produced by the network are readily usable for simulation-based inference.

We also explored cosmological information as a function of graph construction. We showed that a significant amount of information is contained in the variable cardinality of the graph, N^v, i.e. the number of halos in a catalogue, and related this feature to the halo mass function formalism.

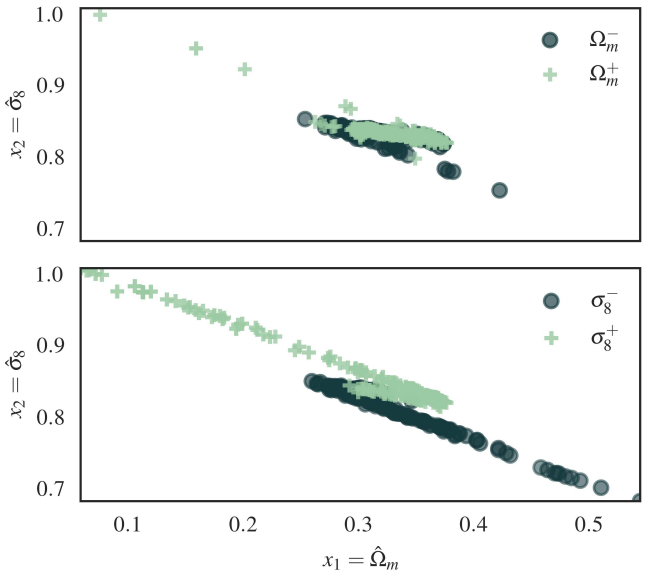


Figure 7.8: gIMNN score summaries for test derivative datasets ($\boldsymbol{\theta}^{\pm}$ with cosmic variance. A trained network can distinguish between simulations generated at different parameter values, indicated by the different distributions in summary space. These score summaries are *not* explicit predictions for cosmological parameters; they should be considered as informative summaries of the cosmological parameters that can be used for ABC or density estimation for building cosmological posteriors. In general and on average, simulations generated at lower Ω_m and σ_8 values have lower pseudo-maximum likelihood estimated values for these parameters and vice versa, however, they should not be taken as predictions of the cosmological parameters.

This test demonstrated a distinct advantage in graph representation: allowing data vector size to vary with cosmology, *combining both positional and mass information automatically into just two statistics*.

Next we demonstrated that gIMNN training can be made robust to noise with little information loss in a more realistic setting where halo masses are estimates with measurement error.

We also explored network (epistemic) and data sampling (aleatoric) error in graph representation. We showed that a combined compression of masses and positional information *decreased* data variability, meaning training and validation graphs become more descriptive of their underlying likelihood with decoration, even in a fixed-length scenario.

The results of this work hold several implications for cosmological parameter estimation and study of large-scale structure. The graph framework presented here enables further modular study of nonlinear statistics that combine attributes of mass functions and correlation functions, at a fraction of the computational cost of bispectrum or trispectrum calculation. Cosmological parameter constraints from void catalogues, here the duals to halo graphs, might elucidate complementary constraints to those obtained here (Kreisch et al., 2021).

Coulton et al. (2022) and Jung et al. (2022) recently investigated bispectrum statistics from n-body primordial nongaussianity (pNG) simulations in the *Quijote* suite to investigate propagation of pNG to late-time cosmological structure. A complementary study using gIMNNs might pick up signatures not isolated in the bispectrum framework. The gIMNN formalism could also be used to summarise and understand better the halo merger and clustering statistics as a function of scale or graph feature, such as clique number. Hierarchical aggregation schemes, as described in Section 7.3.1, could make the gIMNN scheme tractable for aggregating catalogued galaxies into subhalo graphs, and further into a global graph over the largest scales. Doing so would scale this analysis to full-sized galaxy catalogues.

Follow-up study is warranted on much larger catalogues and with more parameters, with a view to readying this framework for applications to inference from detailed physical simulations and, ultimately, realistic galaxy surveys.

7.9 Code Availability

The code used for this analysis is available at https://github.com/tlmakinen/cosmicGraphs. Full documentation for the IMNN software is available at https://www.aquila-consortium.org/doc/imnn/index.html.

T.L.M acknowledges the Imperial College London President's Scholarship fund for support of this study, as well as the great discussions with Stephon Alexander, David Spergel, and Doug Finkbeiner that inspired this work. B.D.W. acknowledges support by the ANR BIG4 project, grant ANR-16-CE23-0002 of the French Agence Nationale de la Recherche; and the Labex ILP (reference ANR-10-LABX-63) part of the Idex SUPER, and received financial state aid managed by the Agence Nationale de la Recherche, as part of the programme Investissements d'avenir under the reference ANR-11-IDEX-0004-02. This work was done within the Aquila Consortium and the Learning the Universe Collaboration. The Flatiron Institute is supported by the Simons Foundation.

Appendix

7.10 Comparing Halo Mass and Number Density Functions

Here we discuss the difference between the halo number density function and halo mass function in the context of gIMNN information extraction. The halo number density function,

$$\frac{dn}{dM} = \frac{\rho_o}{M}\frac{d\ln\sigma^{-1}(M)}{dM}f(\sigma), \tag{7.26}$$

and the halo mass function,

$$f(\sigma) = \frac{M}{\rho_o}\frac{dn(M)}{d\ln(\sigma^{-1}(M))}, \tag{7.27}$$

within the Press-Schechter formalism (Press & Schechter, 1974). Integrating the halo number density from a fixed mass $M_{\rm cut}$ yields the number of halos with a mass above this threshold:

$$N(M_i > M_{\rm cut}) = \int_{M_{\rm cut}}^{\infty}\frac{dn}{dM}dM, \tag{7.28}$$

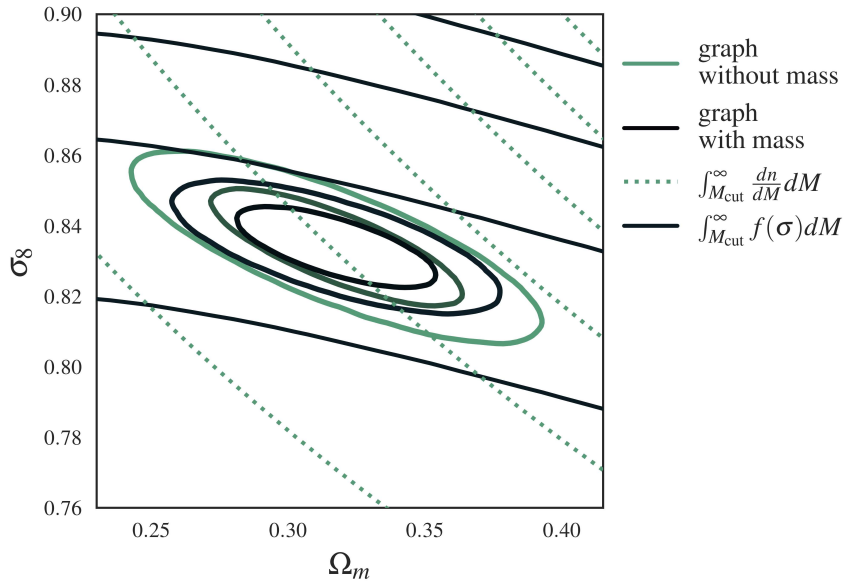


Figure 7.9: Comparison of undecorated graph (green ellipse) and decorated graph (dark ellipse) gIMNN Fishers, plotted over integrated halo number density (dashed green lines) and mass fraction (dark solid lines) functions from fixed $M_{\rm cut}$. Decorating graphs with mass information induces a slight rotation away from dn/dM and towards $f(\sigma)$ contours, since this quantity incorporates both mass and halo number information.

which in our case is the node cardinality of a halo graph, N^v. By contrast, integrating the halo mass function from $M_{\rm cut}$ yields the *fraction of total mass* residing in collapsed halos of mass above $M_{\rm cut}$:

$$F(M_i > M_{\rm cut}) = \int_{M_{\rm cut}}^{\infty} f(\sigma) dM, \tag{7.29}$$

which incorporates *both* halo number N^v and mass information above $M_{\rm cut}$.

We compare these two integrated quantities to undecorated and decorated gIMNN Fisher constraints in Figure 7.9. In the undecorated case (green ellipse), the network has explicit access to halo number and clustering information, resulting in contours more closely aligned with integrated dn/dM (dashed green lines). By contrast, when the graph nodes are annotated with mass labels (black ellipse), the network has explicit access to a combination of halo number, mass, and clustering information. This results in a slight rotation towards the integrated HMF $f(\sigma)$ contours (dark solid lines), since this quantity reflects the addition of mass fraction information. This effect is also illustrated in Figure 7.7. As discussed in Section 7.6, as noise level decreases, the network has access to sharper mass information, inducing a rotation towards the integrated HMF line.

7.11 Details of Graph Assembly in `Jax`

Here we detail `Jax`-compatible graph assembly. `Jax` is a pseudo-compiled language, meaning arrays must have a pre-determined fixed length before sent to a GPU device for operations like gradient descent. We navigate this constraint by padding graph features by pre-determined fixed values, and masking features to assess information content in a modular fashion.

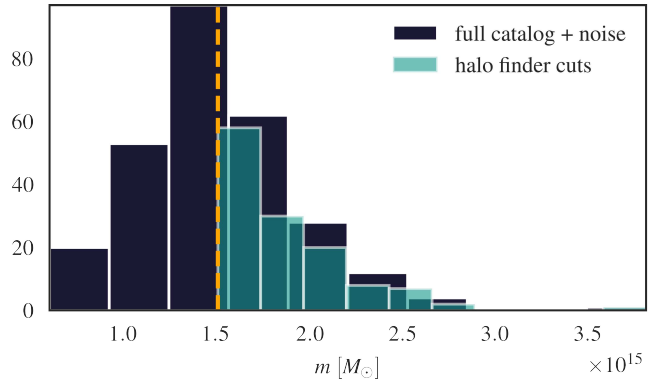


Figure 7.10: Mass distribution after added noise, $\sigma_{\rm noise} = 0.2M_{\rm cut}$ (black) and simulated halo finder cuts (teal) for a single fiducial simulation for masses larger than $1.1 \times 10^{15}\ M_\odot$. The orange dashed lines indicates the minimum mass considered by the "survey" cutoff, $M_{\rm cut} = 1.5 \times 10^{15}\ M_\odot$.

The *Quijote* catalogues are assembled into graphs by first making a mass cut on a larger selection of halos. Masses are then padded with a pre-determined padding value, generally chosen to be $\max(N^v) + 10$. These dummy halos are assigned a very large mass value and a position outside of the 1Gpc^3 box. A distance matrix is then computed for *both* halo and dummy halo nodes, along with sender-receiver indexes, (s_k, r_k). In the invariant graph case we also compute relative angles between all halos, outlined in Section 7.2.2. Connections $|\mathbf{d}_{ij}| > r_{\rm connect}$ and $\mathbf{d}_{ij} = 0$ (self-edges) are then removed and replaced by a large padding value larger than $r_{\rm connect}$. We compute N^e by the number of distances that fit these criteria. Distance values and sender-receiver arrays are then sorted smallest to largest and slotted into padded arrays of length $\max(N^e) + 10$, where padded edge values are then replaced with zeros. We encountered gradient stability issues when padding was made too large for e.g. smaller edge sets, since more padding means the network is asked to operate on extra non-informative features. This sorting arrangement is advantageous since small distances (local connections) are always included in the halo graph's edges, even if the padding container is chosen to be too small for all edges for a given $r_{\rm connect}$ value. What this means is that networks trained on a small $r_{\rm connect}$ can generalize reasonably well to datasets constructed with larger connection criteria.

Adding Noise. When constructing noisy catalogues on-the-fly, we first truncate catalogues with a smaller $M_{\rm cut} = 1.1 \times 10^{15} M_\odot$ to include more halos in the true catalog. Every training epoch, a new realisation of observational noise is then added to the masses according to Section 7.6, after which halos below $M_{\rm cut} = 1.5 \times 10^{15} M_\odot$ are discarded, illustrated in Figure 7.10. The graph edge attributes are then computed for the remaining halos as described above. For the noise scheme chosen in this work, this results in approximately equal numbers of halos being discarded above and below the mass cut line, yet increases the uncertainty in the informative HMF number count, which inflates constraints in Ω_m.

7.12 Details of the GNN Architecture

Masking Graph Features. To conduct information extraction tests with and without node decoration, the GNN architecture must remain fixed between decorated and undecorated cases. To mask a graph node or edge feature, an indicator **1** is assigned to the array in place of numerical value. These uninformative features are fed through the network but do not contribute to the information extraction since the same operation is performed across fiducial and derivative datasets, yet ensure a fair comparison of information as a function of catalogue data features since network architectures (hidden and output size dimensions) could be kept fixed with the same initialized weights.

CHAPTER 8

FISHNETS

Fishnets: Information-Optimal, Scalable Aggregation for Sets and Graphs

T. Lucas Makinen[1], Justin Alsing[2], Benjamin D. Wandelt[3,4]

[1]Imperial Centre for Inference and Cosmology (ICIC) & Astrophysics Group, Imperial College London, Blackett Laboratory, Prince Consort Road, London SW7 2AZ, United Kingdom

[2]Oskar Klein Centre for Cosmoparticle Physics, Stockholm University, Stockholm SE-106 91, Sweden

[3]Sorbonne Université, CNRS, UMR 7095, Institut d'Astrophysique de Paris, 98 bis boulevard Arago, 75014 Paris, France

[4]Center for Computational Astrophysics, Flatiron Institute, 162 5th Avenue, New York, NY 10010, USA

Submitted to the Journal of Machine Learning Research; Makinen et al. (2023)

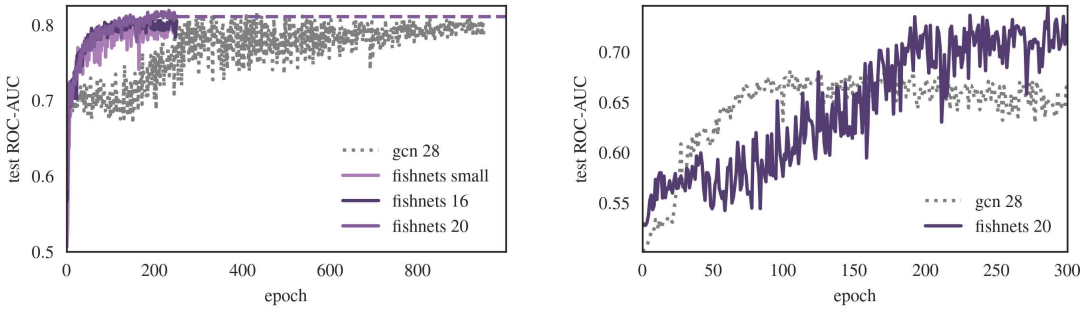
Abstract

Set-based learning is an essential component of modern deep learning and network science. Graph Neural Networks (GNNs) and their edge-free counterparts Deepsets have proven remarkably useful on ragged and topologically challenging datasets. The key to learning informative embeddings for set members is a specified aggregation function, usually a sum, max, or mean. We propose Fishnets, an aggregation strategy for learning information-optimal embeddings for sets of data for both Bayesian inference and graph aggregation. We demonstrate that i) Fishnets neural summaries can be scaled optimally to an arbitrary number of data objects, ii) Fishnets aggregations are robust to changes in data distribution, unlike standard deepsets, iii) Fishnets saturate Bayesian information content and extend to regimes where Markov Chain Monte Carlo (MCMC) techniques fail and iv) Fishnets can be used as a drop-in aggregation scheme within GNNs. We show that by adopting a Fishnets aggregation scheme for message passing, GNNs can achieve state-of-the-art performance versus architecture size on benchmark datasets over existing architectures with a fraction of learnable parameters and faster training time.

8.1 Introduction

Aggregating information from independent data in an optimal way is a fundamental problem in statistics and machine learning. On one hand, frequentist analyses need optimal estimators for data compression, while on the other Bayesian analyses need small informative summaries for simulation-based inference (SBI) schemes (Cranmer et al., 2020a). In a deep learning context graph neural networks (GNNs) rely on aggregation schemes to pool information over large data structures, where each feature might be weakly informative, but at a graph level might contribute a lot of information for predictive or regression tasks (Zhou et al., 2020).

Up until now, graph aggregation schemes have relied on simple, fixed operations such as mean, max, sum, (Corso et al., 2020b; Kipf & Welling, 2017; Hamilton et al., 2017; Xu et al., 2019a), variance, or trainable variants of these aggregators (Battaglia et al., 2018; Li et al., 2020), which are susceptible to generalisation issues in heterogeneous data aggregation, and may contribute to GNN "bottlenecking" over large aggregation neighborhoods (Alon & Yahav, 2021; Giovanni et al., 2024). We introduce a new optimal aggregation scheme grounded in information-theoretic principles. By leveraging the additive structure of the log-likelihood for independent data and underlying Fisher curvature, we can construct a learned summary space that asymptotically contains maximal information (Vaart, 1998; Coulton & Wandelt, 2023). We show that this formalism captures relevant information in both a Bayesian inference context as well as for edge aggregation in graphs.



(a) Aggregation test performance on ogbn-proteins. Fishnets saturates the patience criteria within 250 epochs (dashed line).

(b) Test performance in noisy edge setting.

Figure 8.1: Representative Test ROC-AUC curves for (a) benchmark and (b) noisy proteins datasets. Fishnets aggregation within GCNs clearly saturates information more quickly than GCNs and can also handle noisy edges and contextual information through explicit weight parameterization.

Contributions. In this work we establish Fishnets, a new, information-optimal aggregation scheme

for graph and set-based data. By explicitly learning inverse-Fisher weights in addition to neural score embeddings, we are able to achieve i) asymptotic optimality, ii) scalability, and iii) robustness to changes in the generative distribution for the data and its noise characteristics. We are able to construct optimal summary statistics for independent data for SBI applications, and using the same formalism are able to beat key benchmark GNN learning tasks with far smaller architectures in faster training time than leading networks.

Summary of Results. In Fig. 8.1 we show how incorporating our aggregation scheme improves model convergence for benchmark (8.1(a)) and realistic noisy ogb-proteins (8.1(b)). We provide other GNN benchmark and model performance specifications in Table 8.1 and demonstrate that incorporating Fishnets aggregation as a drop-in replacement in an existing GCN framework enables faster convergence and better performance with much smaller model architectures. We carefully explore the information capture of our aggregation in Section 8.4. We explain this improvement by demonstrating information saturation, robustness, and scalability in a Bayesian context for increasingly difficult problems, and highlight where existing aggregators fall short. In Section 8.5 we detail the GNN benchmarks and improved aggregator performance.

ogb dataset performance

dataset	GCN	**fishnets**
ogb-arxiv	0.7100	0.7062
ogb-molhiv	0.7600	**0.8000**
ogb-proteins	0.8425	**0.8444**

ogb-proteins comparison

model	# params	test ROC-AUC
GCN-112	1,887,144	0.8425 ± 0.0018
fishnets-8	146,596	0.8410 ± 0.0013
fishnets-16	280,740	$\mathbf{0.8444 \pm 0.0018}$

Table 8.1: (*left*) Summary of benchmark improvement within GCN framework with Fishnets aggregation. (*right*) Model size comparison for ogb-proteins benchmark. Fishnets aggregation improves performance with $\sim 15\%$ of the learnable parameters.

8.2 Method: Optimal Aggregation of independent (heterogeneous) data

8.2.1 Fisher Information and Optimality Definitions

We first define the notion of information optimality using the likelihood principle. Data $\mathbf{d}$ is related to some parameters (or quantities of interest) via a log-likelihood $\mathcal{L} = \ln p(\mathbf{d}|\boldsymbol{\theta})$. We would like to obtain a compression mapping $f : \mathbf{d} \mapsto \mathbf{t}$ from N data to n_p numbers $\mathbf{t}$ which preserves as much

information about the parameters $\boldsymbol{\theta}$ as possible. We define *the information inequality* to quantify how informative this mapping is (Lehmann & Casella, 1998):

$$\mathrm{Var}_{\boldsymbol{\theta}}\left[t_\alpha\right] \geq \left(\mathbf{A}^T\mathbf{F}^{-1}\mathbf{A}\right), \tag{8.1}$$

where $\mathbf{A} = \nabla\mathbb{E}_{\boldsymbol{\theta}}\left[\mathbf{t}^T\right]$ and the *Fisher Information matrix* is

$$\mathbf{F} = -\mathbb{E}_{\boldsymbol{\theta}}\left[\nabla\nabla^T\mathcal{L}\right] = \mathbb{E}_{\boldsymbol{\theta}}\left[\nabla\mathcal{L}\nabla^T\mathcal{L}\right], \tag{8.2}$$

where the last equality holds under mild regulatory conditions (Alsing & Wandelt, 2018). The Fisher matrix is the curvature of the log-likelihood, and is in general a function of the parameters. In the case where the compressed numbers $\mathbf{t}$ are unbiased estimators of the parameters, $\mathbb{E}_{\boldsymbol{\theta}}\left[\mathbf{t}\right] = \boldsymbol{\theta}$, $\mathbf{A} = \mathbb{I}$ and the information inequality (8.1) reduces to the Cramér-Rao bound (Cramér, 1946):

$$\mathrm{Var}_{\boldsymbol{\theta}}\left[t_\alpha\right] \geq \mathbf{F}^{-1}_{\alpha\alpha}. \tag{8.3}$$

Maximising the Fisher information over parameter space decreases the variance on the estimates of the quantities of interest $\boldsymbol{\theta}$. Alsing & Wandelt (2018) show that the score function, $\mathbf{t} = \nabla\mathcal{L}$ saturates the lower bound of (8.1) around a fiducial point, $\boldsymbol{\theta}_*$. We reproduce this proof in the general case over a space of $\boldsymbol{\theta}$ and relate information saturation to parameter estimators in Appendix 8.7.

Maximum likelihood estimators (MLEs) are the asymptotically-optimal estimators for predictive tasks. When they are available, they provide an optimally-informative embedding of the data with respect to the parameters of interest, $\boldsymbol{\theta}$ (see (Alsing & Wandelt, 2018) and Appendix 8.7).

8.2.2 Set-like Data Likelihoods

Many inference problems consist of a *set* of data vectors, $\{\mathbf{d}_i\}_{i=1}$, $\mathbf{d}_i \in \mathbb{R}^N$ which obey a global model controlled by parameters $\boldsymbol{\theta} \in \mathbb{R}^{n_p}$, and a possibly arbitrarily deep hierarchy of latent values, η. The full data likelihood for interesting parameters $\boldsymbol{\theta}$ is given by the integral over latents,

$$p(\{\mathbf{d}_i\}|\boldsymbol{\theta}) = \int p(\{\mathbf{d}_i\}|\boldsymbol{\theta},\eta)p(\eta|\boldsymbol{\theta})d\eta. \tag{8.4}$$

When the data are independently distributed, their log-likelihood takes the form

$$\ln p(\{\mathbf{d}_i\}|\boldsymbol{\theta}) = \sum_{i=1} \ln p(\mathbf{d}_i|\boldsymbol{\theta}). \tag{8.5}$$

A maximum likelihood estimator can then be formed (iteratively) by the Fisher scoring method (Alsing & Wandelt, 2018):

$$\hat{\boldsymbol{\theta}}^{\rm MLE} = \boldsymbol{\theta}_* + \mathbf{F}^{-1}\mathbf{t}, \tag{8.6}$$

which requires knowledge of a fiducial point $\boldsymbol{\theta}_*$, the score, $\mathbf{t} \in \mathbb{R}^{n_p}$, and Fisher matrix, $\mathbf{F} \in \mathbb{R}^{n_p \times n_p}$. For problems like linear regression where the analytic form of $\mathbf{F}$ and $\mathbf{t}$ are known, Eq. (8.6) gives the exact MLE for the parameters in a single iteration in the Gaussian approximation, given the dataset. In the case of independent data, both of the score and Fisher information are additive.

Taking the gradient of the log-likelihood with respect to the parameters, the score $\mathbf{t} = \boldsymbol{\nabla}_{\boldsymbol{\theta}} \ln p(\{\mathbf{d}_i\}|\boldsymbol{\theta})$ for the full dataset is the sum of the scores of the individual data points:

$$\mathbf{t} = \sum_{i=1} \boldsymbol{\nabla}_{\boldsymbol{\theta}} \ln p(\mathbf{d}_i|\boldsymbol{\theta}) = \sum_{i=1} \mathbf{t}_i(\mathbf{d}_i) \tag{8.7}$$

Taking the gradient again yields the Hessian, or Fisher information matrix (Amari, 2021; Vaart, 1998) for the dataset,

$$\mathbf{F} = \sum_{i=1} \boldsymbol{\nabla}_{\boldsymbol{\theta}} \boldsymbol{\nabla}_{\boldsymbol{\theta}}^T \ln p(\mathbf{d}_i|\boldsymbol{\theta}) = \sum_{i=1} \mathbf{F}_i(\mathbf{d}_i), \tag{8.8}$$

which is also comprised of a sum of Fisher matrices of individual data. Once the score and Fisher matrix for a dataset are known, the two can be combined to form a pseudo-maximum likelihood estimate (MLE) for the target parameters following (8.6). Therefore, constructing optimal embeddings of independent data with respect to specific quantities of interest just requires aggregating the scores and Fishers, and combining them as in (8.6). However, in general explicit forms for the likelihood (per data vector) may not be known. In this general case, as we will show in the following section, we can parameterize and learn the score and Fisher using neural networks.

8.2.3 Twin Fisher-Score Networks

For many problems, however, the analytic form of the Fisher and score are not known. Here we propose *learning these functions with neural networks*. Due to the additive structure of (8.8) and

(8.7), we can parameterize the *per-datapoint* score and Fisher with twin neural networks:

$$\hat{\mathbf{t}}_i = \mathbf{t}(\mathbf{d}_i; w_t) \in \mathbb{R}^{n_p}; \qquad \mathbf{t}_{\rm NN} = \sum_i \hat{\mathbf{t}}_i \tag{8.9}$$

$$\hat{\mathbf{F}}_i = \mathbf{F}(\mathbf{d}_i; w_F) \in \mathbb{R}^{n_p \times n_p}; \qquad \mathbf{F}_{\rm NN} = \sum_i \hat{\mathbf{F}}_i \tag{8.10}$$

where the score and Fisher network are parameterized by weights w_t and w_F, respectively. The twin networks output a score and Fisher for each datapoint (see Appendix 8.8 for formalism), which are then each summed to obtain a global score and Fisher for the dataset. We can then compute parameter estimates using these aggregated embeddings following (8.6):

$$\hat{\boldsymbol{\theta}}_{\rm NN}\left(\{\mathbf{d}_i\}; w_t, w_F\right) = \mathbf{F}_{\rm NN}^{-1}\mathbf{t}_{\rm NN} + c, \tag{8.11}$$

where the fiducial point $\boldsymbol{\theta}_* = c = 0$ can be set to an arbitrary constant. Provided the embeddings $\hat{\mathbf{t}}_i$ and $\hat{\mathbf{F}}_i$ are descriptive enough, the summation formalism can be used to obtain Fisher and score estimates under an implicit likelihood for datasets with heterogeneous structure and arbitrary size. These summaries can be regarded as sufficient statistics, since the score as a function of parameters could in principle be used to reconstruct the likelihood surface up to a constant (Alsing & Wandelt, 2018; Hoffmann & Onnela, 2023).

Loss Function. In a regression scenario, we draw data-parameter pairs from the joint distribution $\boldsymbol{\theta}, \{\mathbf{d}_i\} \curvearrowleft p(\{\mathbf{d}_i\}, \boldsymbol{\theta})$ and compute $\boldsymbol{\theta}_{\rm NN}$ from (8.11). The twin networks can then be trained jointly using a negative-log Gaussian loss:

$$\mathcal{L}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_{\rm NN};\ w_t, w_F) = \frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{\rm NN})^T \mathbf{F}_{\rm NN}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{\rm NN}) - \frac{1}{2}\ln\det \mathbf{F}_{\rm NN}. \tag{8.12}$$

where $\boldsymbol{w} = (w_t, w_F)$ Minimizing this loss with respect to the neural network weights ensures that information is saturated via maximising the aggregated Fisher, and forces the distance between embedding MLE and parameters to be minimized with respect to the Cramér-Rao bound ((9.3)) as a function of the data. This loss can also be interpreted as a maximum likelihood (MLE) loss for the quantities of interest $\boldsymbol{\theta}$, as opposed to typical mean-square error (MSE) regression losses (see Appendix 8.11 for deepsets details).

8.3 Related Work

Deepsets Mean Aggregation. A comparable method for learning over sets of data is regression using the Deepsets (DS) formalism Zaheer et al. (2018). Here an embedding $f(\mathbf{d}_i; w_1)$ is learned for each datum, and then aggregated with a fixed permutation-invariant scheme and fed to a global function g; $\hat{\boldsymbol{\theta}} = g\left(\bigoplus_{i=1} f(\mathbf{d}_i; w_1);\ \ w_2\right)$. The networks are optimised minimising a squared loss against the true parameters, $\text{MSE}(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta})$. When the aggregation is chosen to be the mean, the deepsets formalism is scalable to arbitrary data and becomes equivalent to the Fishnets aggregation formalism *with flat weights across the aggregated data* (see Appendix 8.11 for in-depth treatment).

Learned Softmax Aggregation. Li et al. present a learnable softmax counterpart to the DS aggregation scheme in the context of edge aggregation in GNNs. Using the above notation, their aggregation scheme reads:

$$\text{SoftmaxAgg}(\cdot) = \sum_{i=1} \frac{\exp\left(\beta f(\mathbf{d}_i; w_1)\right)}{\sum_l \exp\left(\beta f(\mathbf{d}_l; w_1)\right)} \cdot f(\mathbf{d}_i; w_1) \tag{8.13}$$

where β is a learned scalar temperature parameter and $f(\cdot; w_1)$ is some embedding layer. They show that adopting this aggregation scheme allows more graph convolution (GCN) layers to be stacked efficiently to deepen GNN models. Many other aggregation frameworks have been studied, including Graph Attention (Veličković et al., 2018), LSTM units (Hamilton et al., 2017), Recurrent aggregations (Soelch et al., 2019), and scaled multiple aggregators (Corso et al., 2020a).

8.4 Experiments: Bayesian Information Saturation

Bayesian Simulation Based Inference (SBI) provides a framework in which to perform inference with intractable likelihood. There have been massive developments in SBI, such as neural ratio estimation (Miller et al., 2021a) and density estimation (Alsing et al., 2019; Papamakarios et al., 2019a). Key to all of these methods is compressing a large number of data down to small summaries–typically one informative summary per parameter of interest to preserve information (Alsing & Wandelt, 2018; Charnock et al., 2018; Makinen et al., 2021). ML methods like regression (Jeffrey & Wandelt, 2020) and information-maximising neural networks (Charnock, 2019; Makinen et al., 2022, 2021) are very good at learning embeddings for highly structured data like images, and can do so losslessly (Makinen

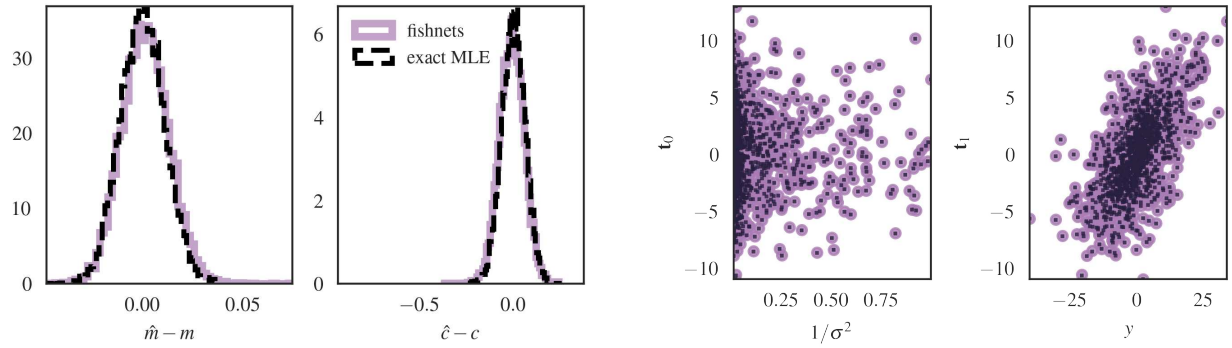
et al., 2021). For unstructured datasets comprised of many independent data, the task of constructing optimal summaries amounts to an aggregation task (Zaheer et al., 2018; Hoffmann & Onnela, 2023; Wagstaff et al., 2019). The Fishnets formalism is an optimal version of this aggregation. What deepsets and "learned" aggregation functions are missing is explicitly constructing the inverse-Fisher weights per datapoint, as well as being able to construct the total Fisher information, which is required to turn summaries into unbiased estimators (Alsing & Wandelt, 2018). Explicitly learning the $\mathbf{F}^{-1}$ weights in addition to the score allows us to achieve 1) asymptotic optimality 2) scalability, and 3) robustness to changes in information content among the data.

In this section we demonstrate the 1) information saturation, 2) robustness and 3) scalability of the Fishnets aggregation through two examples in the context of SBI, and highlight the shortcomings of existing aggregators. We first investigate a linear regression scaling problem and introduce a robustness test in which Fishnets outperforms deepset and learned softmax aggregation on test data. We then extend Fishnets to an inference problem with nuisance (latent) parameters and censorship to demonstrate the applicability of network scaling to a regime where MCMC becomes intractable.

8.4.1 Validation Case: Linear Regression

We use a toy linear regression model to validate our method and demonstrate network scalability. We consider the form $y = mx + b + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma)$, where the parameters of interest are the slope and intercept $\boldsymbol{\theta} = (m, b)$. This likelihood has an analytically-calculable score and Fisher matrix (see Appendix 8.9.1), which can be used to calculate exact MLE estimates for the parameters $\boldsymbol{\theta} = (m, b)$ via (8.6). We choose wide Gaussian priors for θ, and uniform distributions for $x \in [0, 10]$ and $\sigma \in [1, 10]$. For network training, we simulate 10^4 datasets of size = 500 datapoints. For testing, we generate an additional 10^4 datasets of size = 10^4 datapoints to demonstrate scalability. See Appendix 8.9.2 for neural architecture details.

Results. We display a comparison of test set performance to the true MLE solution in Figure 8.2(a), and slices of the true and predicted score vectors as a function of input data. The networks are able to recover the exact score and Fisher information matrices (see Figure 8.2(b)), even when scaled up 20-fold. *This test demonstrates that Fishnets can (1) saturate information on small training sets to enable scalable predictions on far larger aggregations of data (2).*



(a) Fishnets saturate information for datasets 20 times larger than the training set.

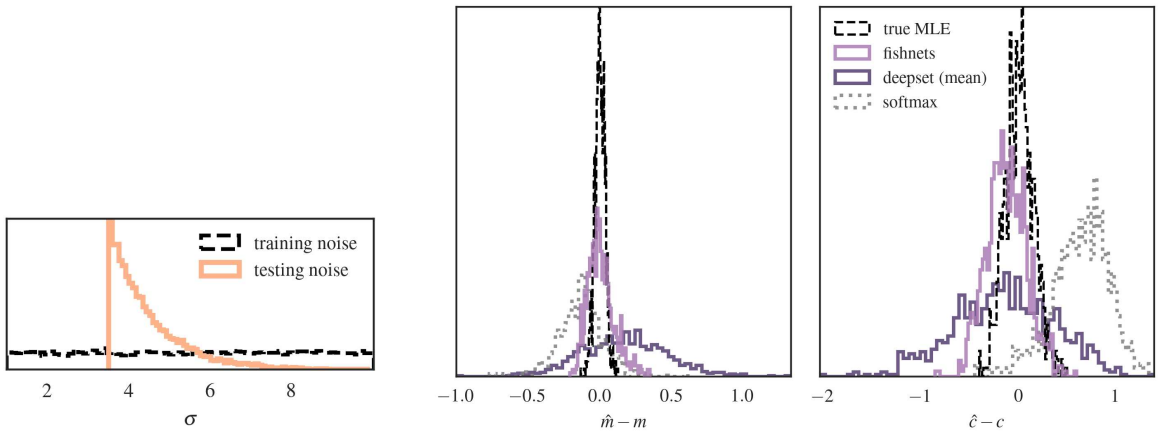
(b) Fishnets achieve the exact form of the score as a function of input data in the linear regression case.

Figure 8.2: (a) Residual maximum likelihood estimates for slope (*left*) and intercept (*right*) scatter about the truth for linear regression test datasets of size $= 10^4$. The solid pink line is obtained from a weighted average of an ensemble of Fishnets networks, *which were trained on datasets of size* $= 500$. (b) Slices of true (dark) and network predicted (pink) score vector components as a function of data inputs for the $= 10^4$ test set.

8.4.2 Robustness to changes in the underlying data distributions

In real-world applications, actual data processed by a network might follow a different distribution than that seen in training. Here we compare three different network formalisms on changing shapes of target data distributions.

We train three networks on the same $= 500$ datasets as before: a sum-aggregated Fishnets network, a mean-aggregated deepset, and a learned softmax-aggregated deepset with mean-square error loss. To demonstrate the improvement in aggregation Fishnets offers, we adopt smaller networks for the regression task (see Table 8.2 and Appendix 8.9.2 for architecture details). We apply our trained networks to test data $= 850$ with noise variances and x values drawn from different distributions to the training data: $\sigma \curvearrowleft \text{Exp}(\lambda = 1.0)$ centred at $\sigma = 3.5$, truncated at $\sigma = 10.0$, and $x \curvearrowleft \mathcal{U}(0, 3)$. The noise and covariate distributions have the same support as the training data, but have different expectation values and distributions, which can pose a problem for the mean-aggregation used in the standard deepsets formalism. We display results in Figure 8.3(b). The heterogeneous Fishnets aggregation allows the network to correctly embed the noisy data drawn from the different distributions, while a significant loss in information can be seen for flat mean aggregation. The learned softmax aggregation improves the width of the residual distribution, but is still significantly wider than the Fishnets solution. We quote numeric results in Table 8.2.



(a) Changing noise distribution. (b) Robustness testing different aggregators

Figure 8.3: (b) Fishnets (pink) are robust to different noise distributions in test data (8.3(a)). Deepsets (grey) can return biased results for some parameters (left) and lossy estimates for others (right). Learned softmax aggregation appears to provide lossier and biased parameter estimates.

These robustness tests show that Fishnets successfully learns *per-object* embeddings (score) and weights (Fisher) within sets, *while being robust to changing shapes of the training distributions of these quantities (3).* This test also shows that even in a very simple prediction scenario, *common and learned aggregators can suffer from robustness issues.*

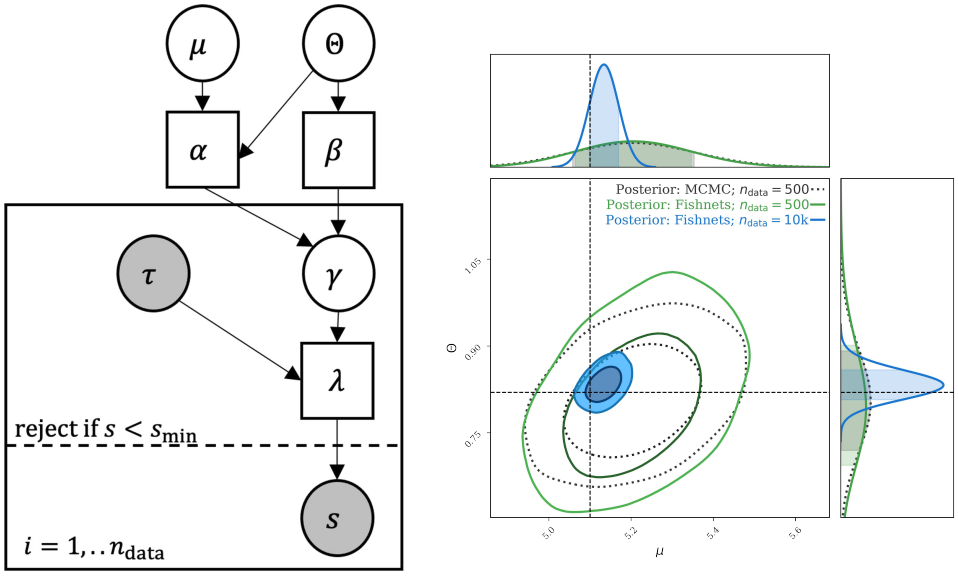
	network	# **params**	**MSE**$(\hat{\boldsymbol{m}}, \boldsymbol{m}_{\text{true}})$	**MSE**$(\hat{\boldsymbol{c}}, \boldsymbol{c}_{\text{true}})$
robustness test	**fishnets**	$10,855$	$\mathbf{0.007 \pm 0.017}$	$\mathbf{0.046 \pm 0.078}$
	deepset	$87,810$	0.120 ± 0.178	0.285 ± 0.406
	softmax	$87,811$	0.042 ± 0.069	0.482 ± 0.347

Table 8.2: Summary of robustness testing for different set-based networks. Fishnets' Fisher aggregation has an advantage over mean- and learned softmax deepsets aggregation when test data follows a different distribution than the training suite, and does so with an eigth of the number of learnable parameters.

8.4.3 Scalable Inference With Censorship and Nuisance Parameters

As a non-trivial information saturation example we consider a censorship inference problem with latent parameters inspired by epidemiological studies. Consider a serum which, when injected into a patient, decays over time, and the (heterogeneous) decay rate among people is not well known. A population of patients are injected with the serum and then asked to come back to the lab within $t_{\text{max}} = 10$ days for a measurement of the remaining serum-levels in their blood, s. We can cast this problem as a Bayesian hierarchical model visualised in Figure 8.4(a) (see Appendix 8.9.3 for details)

where the goal is to infer the mean μ and scale Θ of the decay rate Gamma distribution from the data, $\{\tau_i, s_i\}$. In the censored case, measurements are rejected if $s_i < s_{\rm min}$, and collected until $n_{\rm data}$ valid samples are collected. As a ground-truth comparison for the uncensored version of this problem, we sample the above hierarchical model using Hamiltonian Monte-Carlo (HMC). For comparison, we utilize the same Fishnets architecture and small-data training setup as before to predict (μ, Θ) from data inputs $[\tau_i, s_i]^T$. Once trained, we generated a new suite of $= 500$ simulations and pass the data through the network to learn a neural posterior from $(\hat{\theta}_{\rm NN}, \boldsymbol{\theta})$ pairs. We then evaluated both HMC and neural posteriors at the same target data. Finally, using the same network we perform the same procedure, this time with simulations of size $= 10^4$, *where the HMC becomes computationally prohibitive.*



(a) Gamma population hierarchical Bayesian model diagram.

(b) Information Saturation against HMC (MCMC).

Figure 8.4: (a) Gamma population plate diagram. Circles represent random variables, boxes are deterministic quantities, and shaded variables are observed as data. The dashed line represents a possible censorship in measurement. Measurements of data $(t, s)_i$ are conducted until $n_{\rm data}$ samples are drawn. (b) The same Fishnets network can be used for inference on datasets much larger than those used in training. The twin Fishnet architecture was trained on $= 500$. We then compress a target dataset and perform density estimation (green) and compare to an MCMC sampler as our true posterior (black dashed). Fishnets nearly saturates the information. We then *use the same network to compress simulations of* $= 10^4$ to obtain the blue contours.

Results. We display inference results in Figure 8.4(b). The summaries obtained from Fishnet compression of the small data (green) result in posteriors that hug the "true" MCMC contours (black), *indicating information saturation.* Extending the same network on the larger data results in intuitively smaller contours (blue). It should be emphasized that $= 10^4$ *is a regime where the MCMC*

inference is no longer tractable on standard devices. Fishnets here allows for 1) much faster posterior calculation and 2) allows for optimal inference on larger data set sizes without any retraining.

As a final demonstration we solve the same problem, this time subject to censorship. In the censored case, the target joint posterior defined by the hierarchical model requires computing an integral for the selection probability as a function of the model hyper-parameters; in general, these selection terms make Bayesian problems with censorship computationally challenging, or intractable in some cases (Qi et al., 2022; Dickey et al., 1987).

We train Fishnets on the small data size, subject to censorship below $s_{\min}$. We obtain posteriors of the same shape of the censored case, but for a consistency check perform a probability-integral transform (PIT) test for the neural posterior. For each parameter we want the marginal PIT test to yield a uniform distribution to show that the learned posterior behaves as a continuous distribution. We display these results in Figure 8.5. We obtain a Kolmogorov-Smirnov test (Massey Jr., 1951) p-value of 0.628 and 0.233 for parameters μ and Θ, respectively, indicating that our posterior is well-parameterized and robust.

8.5 Graph Neural Network Aggregation

Graphs can be thought of as tuples of sets within connected neighborhoods. Graph neural networks (GNNs) operate by message-passing along edges between nodes. For predicting node- and graph-level properties, an aggregation of these sets of edges $\{\mathbf{e}_{ij}\}$ or nodes $\{\mathbf{v}_i\}$ is required to reduce features to fixed-size feature vectors. Whereas in the SBI setting, we are interested in finding optimal estimators for specific parameters of interest, in the GNN aggregation setting we are implicitly trying to find a compact latent (embedding) representation of the aggregated neighborhood data, and optimally estimate and propagate those latent features through the GNN architecture.

Here we compare the Fishnets aggregation scheme as a drop-in replacement for learned softmax aggregators within the graph convolutional network (GCN) scheme presented by Li et al.. We can

rewrite our aggregations to occur within neighborhoods of nodes:

$$\text{SoftmaxAgg}(\cdot) = \sum_{i \in \mathcal{N}(v)} \frac{\exp\left(\beta \mathbf{e}_{iv}\right)}{\sum_{l \in \mathcal{N}} \exp\left(\beta \mathbf{e}_{li}\right)} \cdot \mathbf{e}_{iv}, \tag{8.14}$$

$$\text{FishnetsAgg}(\cdot) = \left(\sum_{i \in \mathcal{N}(v)} \mathbf{F}(\mathbf{e}_{iv}) \right)^{-1} \left(\sum_{i \in \mathcal{N}(v)} \mathbf{t}(\mathbf{e}_{iv}) \right), \tag{8.15}$$

where the aggregation occurs in a neighborhood $\mathcal{N}$ of a node v. The Fishnets aggregation requires a bottleneck hyperparameter, n_p, which controls the size of the score embedding $\mathbf{t}(\mathbf{e}_{iv}) \in \mathbb{R}^{n_p}$ and Fisher Cholesky factors $\mathbf{F}_{\text{chol}} \in \mathbb{R}^{n_p(n_p+1)/2}$. We use a single linear layer before aggregation to obtain score and Fisher components from hidden layer embeddings.

8.5.1 Drop-in replacement for Graph Benchmark Datasets

Here we replace the learned softmax aggregation with Fishnets aggregation in Li et al.'s publicly-available best-performing models. We change four hyperparameters in testing our new architectures: number of layers, n_p, dropout, and learning rate. We study several graph datasets from the Open Graph Benchmark (OGB) (Hu et al., 2020, 2021), which require substantial aggregation steps to predict either node or graph-level properties. The object of this study is to investigate *how well fishnets aggregation can perform within an existing architecture*, with fewer layers and minimal hyperoptimisation.

Results. We display benchmark results in Table 8.1, and refer the reader to Appendix 8.10 for architecture and dataset details. This small drop-in study shows that incorporating the more information-efficient Fishnets Aggregation, we can achieve better than or similar results to SOTA GCNs *with a fraction of the trainable parameters and training epochs.*

8.5.2 Focus Study on *ogbn-proteins* Benchmark

In this section we study the proteins dataset in detail and highlight a scenario where the heterogeneous Fishnets aggregation drastically improves performance. Here we expect different node neighbourhoods to have a heterogeneous edge weighting "association score" structure across protein categories, making the Fishnets aggregation ideal for applicability beyond the training set, as in the linear re-

gression case. The association scores can be stochastically modelled with added measurement noise, increasing the difficulty of the classification problem. We adopt a stripped-down version of the training routine presented in Li et al. (2020) (no subgraph and edge preprocessing) to make modifications to the raw data by adding noise. We first benchmark our training routine with smaller GCN and Fishnets aggregation on the noise-free data, and then proceed to adding noise to the edges.

Noisefree Results. We display representative test ROC-AUC curves over training in Figure 8.1(a), and in Table 8.3. Fishnets-16 and Fishnets-20 clearly saturate information within 250 epochs to 79.63% and 81.10% accuracy respectively.

Modelling Uncertain Protein Associations.

Here we incorporate uncertainties on the protein interaction strengths (edges), in order to demonstrate the robustness of the Fishnets approach to changes in the underlying data (noise) distribution on the graph features. We model noisy "measurements" of the protein graph edge associations using a simple Binomial model: taking the dataset edges $\mathbf{p}_{ij} = \mathbf{e}_{ij} \in [\mathbf{0}, \mathbf{1})$ as the"true" association strengths, we can simulate a noisy measurement of those quantities as N weighted coin tosses per edge, where N varies between measurements:

$$N \curvearrowleft \mathcal{U}(20, 200) \tag{8.16}$$

$$\mathbf{n}_{\text{success}} \curvearrowleft \text{Binomial}\left(n = N, p = \mathbf{p}_{ij}\right) \tag{8.17}$$

$$\mathbf{e}_{ij} \leftarrow \left[\hat{\mathbf{p}}_{ij} = \mathbf{n}_{\text{success}}/N, N\right]. \tag{8.18}$$

Note that in the last step the new graph edge now contains the (noisy) measured associations, as well as N (which provides a measure of uncertainty on those estimated interaction strengths). The GNN task is now to learn to re-weight predictions conditioned on the provided N coin toss information, much like feeding in σ in the linear regression case. We train a 28-layer GCN and 20-layer Fishnets. For the test dataset, we alter the distribution for N to be $\mathcal{U}(20, 50) + \mathcal{U}(170, 200)$ such that we sample the extremes of the training distribution support.

Noisy Results. We display test ROC-AUC curves for both networks in Figure 8.1(b), subject to a patience setting of 250 epochs on the validation set. The GCN framework exhibits an early plateau at 64.71% accuracy, while Fishnets saturates to 71.98% accuracy. This stark difference in behaviour can be explained by the difference in formalism: The Fishnets aggregation explicitly learns a weighting

test	network	# params	test ROC-AUC
noisefree	**fishnets-20**	442,372	$\mathbf{0.8110 \pm 0.0021}$
	GCN-28	477,964	0.7951 ± 0.0059
noisy-edges	**fishnets-20**	442,500	$\mathbf{0.7198 \pm 0.0109}$
	GCN-28	478,092	0.6471 ± 0.0090

Table 8.3: Summary of performance on benchmark and noisy edge variants of the proteins dataset. Errorbars denote standard deviation of test ROC-AUC in the last ten epochs of training.

scheme as a function of measured edge probabilities *and* the conditional information N, much like the linear regression case where σ was passed as an input. This scheme helps to learn how to deal with edge-case noise artefacts like the noisy edge test case. Explicitly specifying the inverse-Fisher weighting formalism as an inductive bias (Battaglia et al., 2018) during aggregation can help explain the fast information saturation exhibited in both graph test settings.

8.6 Discussion & Future Work

In this paper we built up an information-theoretic approach to optimal aggregation in the form of Fishnets. Through progressively non-trivial examples, we demonstrated that explicitly parameterizing the score and inverse-Fisher weights of set members results in an aggregation scheme that saturates Bayesian information in non-trivial problems, and also serves as an optimal aggregator for sets and graph neural networks in heterogeneous data scenarios.

The stark improvement in information saturation on the proteins test dataset relative to architecture size and training efficiency indicates that the Fishnets aggregation acts as an information-level inductive bias for GNN aggregation. Follow-up study is warranted on optimizing hyperparameter choices for graph neural network architectures using Fishnets. We chose to demonstrate improved information capture by using an ablation study of smaller models, but careful (and potentially bigger) network design would almost certainly improve results here and potentially achieve SOTA accuracy on common benchmarks.

Appendix

8.7 Saturating the Information Inequality over Parameter space

Here we show that knowing the score function $\mathbf{t} = \nabla\mathcal{L}$ saturates the information inequality and provides a natural data compression function. We first consider the Taylor expansion of the log-likelihood around a fixed fiducial point in parameter space, $\boldsymbol{\theta}_*$, (where $g_* \equiv g(\boldsymbol{\theta} = \boldsymbol{\theta}_*)$):

$$\mathcal{L} = \mathcal{L}_* + \delta\boldsymbol{\theta}^T\nabla\mathcal{L}_* - \frac{1}{2}\delta\boldsymbol{\theta}^T\mathbf{J}_*\delta\boldsymbol{\theta} \tag{8.19}$$

where $\mathbf{J} = -\nabla\nabla^T\mathcal{L}$ is the observed information matrix. To linear order in $\boldsymbol{\theta}$, the data $\mathbf{d}$ couples to the parameters through the score function $\mathbf{t} \in \mathbb{R}^{n_p}$. We can show that $\mathbf{t}$ saturates the information inequality via

$$\mathrm{Cov}_{\boldsymbol{\theta}}\left[\mathbf{t},\, \mathbf{t}\right] = \mathbb{E}_{\boldsymbol{\theta}}\left[\nabla\mathcal{L}_*\nabla_*^T\right] = \mathbf{F}_*, \tag{8.20}$$

where we have used the fact that $\mathbb{E}_{\boldsymbol{\theta}}\left[\nabla\mathcal{L}_*\right] = 0$. From this we observe that the covariance of the score function is the Fisher matrix. Using the fact that

$$\mathbf{A} = \nabla\mathbb{E}_{\boldsymbol{\theta}}\left[\nabla^T\mathcal{L}\right] = \mathbb{E}_{\boldsymbol{\theta}}\left[\nabla\nabla^T\mathcal{L}\right] = -\mathbf{F}_*, \tag{8.21}$$

the right-hand side of the information inequality becomes $\mathbf{A}_*^T\mathbf{F}_*^{-1}\mathbf{A}_* = \mathbf{F}_*$, which shows that the score statistics $\mathbf{t}$ saturate the information inequality. Within this formalism, no statistics can provide more (Fisher) information about the parameters $\boldsymbol{\theta}$.

We can relate this information saturation to an optimal, quasi maximum-likelihood estimator whose covariance is equal to the inverse Fisher information from the above derivation. Maximising the Taylor expansion (8.19) with respect to the parameters yields

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_* + \mathbf{J}_*^{-1}\nabla\mathcal{L}_* \tag{8.22}$$

where both the score $\mathbf{t}_* = \nabla\mathcal{L}_*$ and the observed information $\mathbf{J}_*^{-1}$ depend on the observed data. In practice, we can exchange $\mathbf{J}$ with its expectation value, the Fisher information: $\mathbf{F}_* \equiv \mathbb{E}_{\boldsymbol{\theta}}\left[\mathbf{J}_*\right]$, which

yields

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_* + \mathbf{F}_*^{-1} \nabla \mathcal{L}_*. \tag{8.23}$$

Making this replacement means the MLE estimator only depends on the data through the score function statistics $\mathbf{t} = \nabla \mathcal{L}_*$. The covariance of the MLE estimator (at the expansion point $\boldsymbol{\theta}_*$) is then:

$$\mathrm{Cov}_{\boldsymbol{\theta}_*} \left[\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\theta}} \right] = \mathbf{F}_*^{-1} \mathbb{E}_{\boldsymbol{\theta}_*} \left[\nabla \mathcal{L}_* \nabla^T \mathcal{L}_* \right] \mathbf{F}_*^{-1} = \mathbf{F}_*^{-1}, \tag{8.24}$$

where $\mathbb{E}_{\boldsymbol{\theta}_*} \left[\nabla \mathcal{L}_* \nabla^T \mathcal{L}_* \right] \equiv \mathbf{F}_*$. Hence the covariance of the MLE is equal to the Fisher information matrix at $\boldsymbol{\theta}_*$ and the Cramér-Rao bound is saturated.

The above proof of information saturation calculated the information saturation around a fixed fiducial point. In general, however, by parameterising the score and Fisher functions with neural networks under the loss (8.12) we are learning an embedding $\mathbf{t}(\mathbf{d}_i; w_t)$ and weighting neighborhood $\mathbf{F}(\mathbf{d}_i; w_F)$ as a function of data (and implicitly parameters).

8.8 Calculating the Fisher Matrix from Network Outputs

To ensure that our Fisher matrix is positive-definite, our Fisher-score networks output $n_{\text{params}} + n_{\text{params}} \frac{(n_{\text{params}}+1)}{2}$ numbers as lower triangular entries in a Cholesky decomposition of the Fisher matrix, $\mathbf{L}$. To ensure that the lower triangular entries remain positive-definite, we add a softplus activation to the diagonal entries of $\mathbf{L}$:

$$\mathrm{diag}(\mathbf{L}) \leftarrow \mathrm{softplus}(\mathrm{diag}(\mathbf{L})) \tag{8.25}$$

We then compute the Fisher via:

$$\mathbf{F} = \mathbf{L}\mathbf{L}^T \tag{8.26}$$

The negative-log likelihood loss in Equation 8.12 allows for explicit interrogation of the resulting Fisher matrix at the level of the predicted quantities (parameters), and ensures that the summary space in $\hat{\boldsymbol{\theta}}$ is convex. In the GNN regression formalism, Fishnets does not *explicitly* maximise the Fisher information as a part of the loss, rather the Fisher matrix weights are optimized as an inductive bias as a hidden layer in the GNN scheme.

8.9 Bayesian Information Experiment Details

8.9.1 Scalable Linear Regression

We use a toy linear regression model to validate our method and demonstrate network scalability. We consider the form $y = mx + b + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma)$, where the parameters of interest are the slope and intercept $\boldsymbol{\theta} = (m, b)$. This likelihood has an analytically-calculable score and Fisher matrix,

$$\mathbf{t} = \sum_{i=1} \frac{1}{\sigma_i^2} \begin{bmatrix} x_i(y_i - (m_{\text{fid}} x_i + b_{\text{fid}})) \\ y_i - (m_{\text{fid}} x_i + b_{\text{fid}}))] \end{bmatrix} + \mathbf{t}_0, \tag{8.27}$$

$$\mathbf{F} = \sum_{i=1} \frac{1}{\sigma_i^2} \begin{bmatrix} x_i^2 & x_i \\ x_i & 1 \end{bmatrix} + \mathbf{C}_{\text{p}}^{-1}, \tag{8.28}$$

where $\mathbf{t}_0 = \mathbf{C}_{\text{p}}^{-1}(\theta_{\text{fid}} - \mu_{\text{p}})$, with $\mu_{\text{p}} = \mathbf{0}$ is the mean of the prior on the score, and $\mathbf{C}_{\text{p}}^{-1} = \mathbf{I}$ is added to the Fisher matrix as a prior on the inverse-covariance of the spread of the summaries. With these two expressions we can calculate exact MLE estimates for the parameters $\boldsymbol{\theta} = (m, b)$ via Eq. (8.6). We choose wide Gaussian priors for θ, and uniform priors for $x \in [0, 10]$ and $\sigma \in [1, 10]$. For network training, we simulate 10^4 datasets of size = 500 datapoints. For testing, we generate an additional 10^4 datasets of size = 10^4 datapoints to demonstrate scalability. We use fully-connected MLPs of size [256, 256, 256] with `ELU` activations (Clevert et al., 2015) for both score and Fisher networks. Both networks receive the input data $[y_i, x_i, \sigma_i^2]^T$. We train networks for 2500 epochs with an `adam` optimizer using a step learning rate decay schedule. We train an ensemble of 10 networks in parallel on the same training data with different initializations.

8.9.2 Robustness Network Architecture Comparison

We train three networks on the same = 500 datasets as before: a sum-aggregated Fishnets network, a mean-aggregated deepset, and a learned softmax-aggregated deepset (no Fisher output and standard MSE loss against true parameters MSE($\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}$)). Here we initialise Fishnets with [50,50,50] hidden units for score and Fisher networks, and two embeddings of [128,128,128] hidden units for both deepset networks, all with `swish` (Ramachandran et al., 2017) nonlinearities for the data embedding

(see Table 8.2). All networks are initialised with the same seed.

8.9.3 Gamma Population Model

Consider a serum which increases patients' red blood cell counts, whose decay rate, τ, is not known. A population of patients are injected with the serum and then asked to come back to the lab within $t_{\max} = 10$ days for a measurement of their blood cell count, s. We can cast this problem using the following hierarchical model

$$\begin{aligned}
\mu &\curvearrowleft \mathcal{U}(0.5, 10) \\
\Theta &\curvearrowleft \mathcal{U}(0.1, 1.5) \\
\gamma_i &\curvearrowleft \text{Gamma}(\alpha = \mu/\Theta, \beta = 1/\Theta) \\
\tau_i &\curvearrowleft \mathcal{U}(0, 10) \\
\lambda_i &= A \exp(-\tau_i/\gamma_i) \\
s_i &\curvearrowleft \text{Pois}(\lambda_i),
\end{aligned}$$

where the goal is to infer the mean μ and scale Θ of the decay rate Gamma distribution from the data, $\{\tau_i, s_i\}$. In the censored case, measurements are rejected if $s_i < s_{\min}$, and collected until n_{data} samples are accepted. The model is visualised in a plate diagram in Figure 8.4(a). In the uncensored case, the posterior estimation for this problem is readily solved using a high-dimensional Hamiltonian Monte-Carlo (HMC) sampler. We implement this model in `Numpyro` (Phan et al., 2019b) as a baseline MCMC comparison for our algorithm. For the Fishnets implementation, we generate 10^4 simulations of size $n_{\text{data}} = 500$ over a uniform prior for μ and Θ. We then train the same Fishnets architecture used for the linear regression case with data inputs $[\tau_i, s_i]^T$. Once the networks were trained, we pass a suite of $= 5000$ simulations through the network to generate neural summaries with which to train a density estimation network. Following Alsing et al. (2019), we use Mixture Density Networks to learn an amortized posterior for $p(\hat{\theta}_{\text{NN}}|\theta)$ with three hidden layers of size $[50, 50, 50]$. We then evaluate this posterior at the same target data used for the HMC, shown in green in Figure 8.4(b). The Fishnets compression results in slightly inflated contours, indicating a small leakage of information. To demonstrate scaling, we addidionally generate another simulation at $= 10^4$ using the same random seed. We train another amortised posterior using 5000 simulations at $= 10^4$ and pass the data through the same trained Fishnet architecture. The resulting posterior

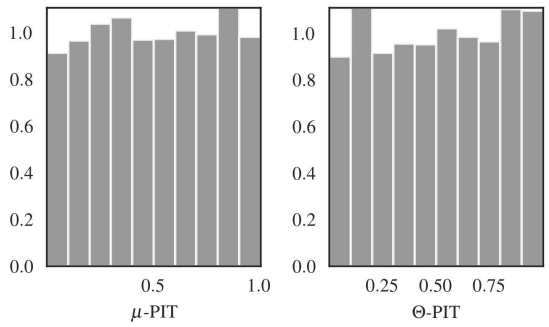


Figure 8.5: Density estimation posteriors obtained from parameter-Fishnets summary pairs are robust over training data. Each parameter's PIT test is close to uniform, which shows that the Fishnets summary posterior has successfully captured the underlying Bayesian information from the data.

is shown in blue for comparison.

8.10 Graph Prediction Benchmark Experiment Details

All GNN models are implemented in PyTorch Geometric (Fey & Lenssen, 2019), and all experiments are performed on a single NVIDIA V100 32GB with the same random seed for initialisation and training. We first describe graph- and node-level prediction tasks, followed by the experimental details on the three benchmark datasets.

Node Property Prediction. This task consists of aggregating edge and node information within neighborhoods to predict properties at the node level. The *ogbn-arxiv* dataset is a directed citation graph of $169,343$ papers summarised as 128-dimensional vectors (nodes) and $1,166,243$ citations (edges), where the direction indicates the citation direction. The task is to predict which of 40 classes each paper belongs to. The *ogbn-proteins* dataset consists of $132,534$ proteins encoded as 8-dimensional one-hot features indicating protein species (nodes) and $39,561,252$ undirected weighted edges indicating association scores between proteins. The task a 112-class classification from aggregated subgraph edges and nodes using an ROC-AUC metric.

Graph Property Prediction. This task requires the aggregation of edges and nodes to global features of a graph. We consider the *ogbn-molhiv* dataset, which is comprised of $41,127$ molecules with atoms arranged as nodes and bonds as edges. The prediction task is binary classification.

ogb-molhiv. This dataset does not provide a node feature for each protein. We initialize the node features via a sum aggregation, e.g. $x_i = \sum_{j \in \mathcal{N}} e_{ij}$, where x_i denotes the initialized node features and

e_{ij} denotes the input edge features. We train a 7-layer DyResGEN model with softmax aggregator with learnable β parameter. A batch normalization is used for each layer. We set the hidden channel size as 256. A dropout with a rate of 0.5 is used for each layer. An Adam optimizer with a learning rate of 0.0001 are used to train the model for 150 epochs. For the Fishnets comparison we train a 3-layer network with Fishnets Aggregation with bottleneck $n_p = 8$ in place of the softmax aggregation, and a learning rate of 0.00002.

ogb-arxiv. We train Li et al. (2020)'s 28-layer ResGEN model with softmax aggregation where β is fixed as 0.1. Full batch training and test are applied. A batch normalization is used for each layer. The hidden channel size is 128. We apply a dropout with a rate of 0.5 for each layer. An Adam optimizer with a learning rate of 0.01 is used to train the model for 500 epochs. For the Fishnets comparison we train a 3-layer version of the same ResGEN network with Fishnets Aggregation with bottleneck $n_p = 9$ in place of the softmax aggregation.

ogb-proteins. This dataset does not provide a node feature for each protein. We initialize the node features via a sum aggregation, e.g. $x_i = \sum_{j \in \mathcal{N}} e_{ij}$, where x_i denotes the initialized node features and e_{ij} denotes the input edge features. We train Li et al. (2020)'s 112-layer DyResGEN with softmax aggregator. A hidden channel size of 64 is used. A layer normalization and a dropout with a rate of 0.1 are used for each layer. We train the model for 1000 epochs with an Adam optimizer with a learning rate of 0.01. For the Fishnets comparison we train an 8-layer and 16-layer version of the same DyResGEN network with Fishnets Aggregation with bottleneck $n_p = 8$ in place of the softmax aggregation. Here we temper the learning rate to 0.005 and decrease the dropout rate to 0.25.

8.10.1 Noisy Proteins Focus Study

Here we again initialize the node features via a sum aggregation.

We test five model architectures using the vanilla *ogbn-proteins* dataset (no subgraph and edge preprocessing as performed by Li et al. (2020)). This change allowed us to flexibly incorporate the added edge feature in the noisy edge setting. To benchmark our training routine we adopt a 28-layer DyResGEN network with learned softmax aggregations and hidden size of 64, and a smaller version of this model with hidden size 14 and 28 layers. We construct two, shallower Fishnets GNNs, with 16 and 20 layers, each with 64 hidden units, and one small model with 14 hidden units and 14 layers. For each graph convolution aggregation, we adopt a "score" bottleneck of $n_p = 10$ for the large Fishnets

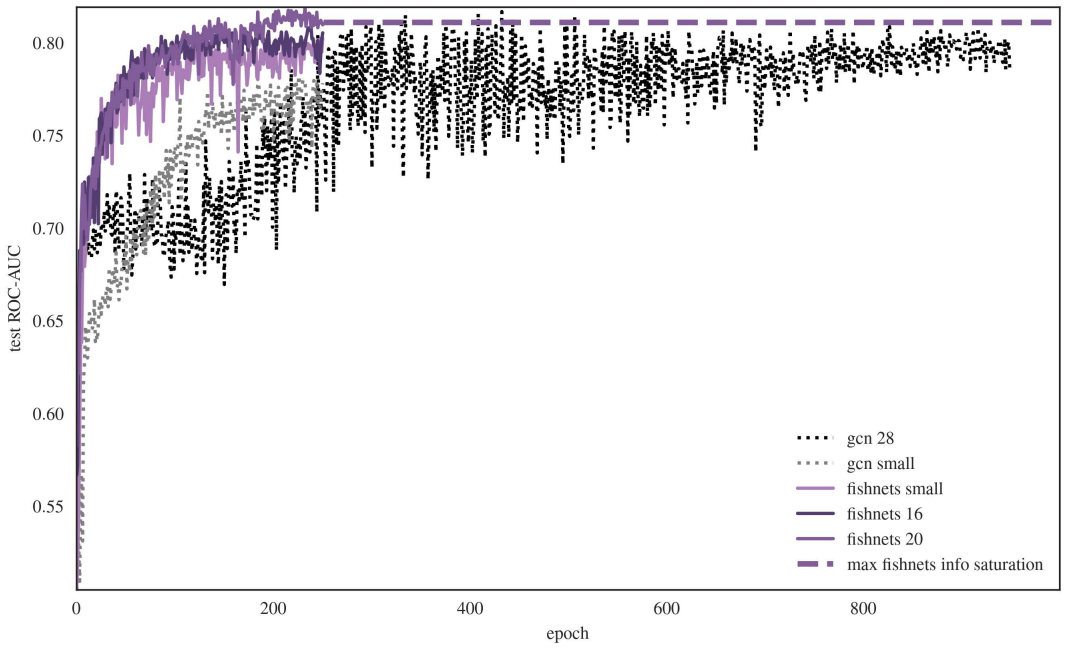


Figure 8.6: Zoomed-in test ROC-AUC training trajectories for models considered in benchmark ablation study on ogbn-proteins.

test	network	# params	test ROC-AUC
noisefree	**fishnets-20**	442,372	$\mathbf{0.8110 \pm 0.0021}$
	fishnets-16	355,584	0.7963 ± 0.0059
	fishnets small	30,360	0.7929 ± 0.0045
	GCN-28	477,964	0.7951 ± 0.0059
	GCN small	33,580	0.7731 ± 0.0052
noisy edges	**fishnets-20**	442,500	$\mathbf{0.7198 \pm 0.0109}$
	GCN-28	478,092	0.6471 ± 0.0090

Table 8.4: Full summary of performance on benchmark and noisy variants of the proteins dataset. Errorbars denote standard deviation of test ROC-AUC in the last ten epochs of training.

models and $n_p = 8$ for the small model. We train all networks with a cross-entropy loss over the same dataset and fixed random seed using an Adam optimizer with fixed learning rate 0.001. We incorporate an early stopping criterion conditioned on the validation data, which dictates an end to training (saturation) when the validation ROC-AUC metric stops increasing for `patience` = 250 epochs.

In the noisy proteins setting we again control for stochasticity in training set loading and added edge noise by fixing the initial random seed before each training run.

8.11 Deepsets Formalism

Summary. The deepsets method presented by Zaheer et al. shows in Theorem 9 that any function over a *countable* set can be decomposed in the form $f(X) = \rho\left(\sum_{x \in X} \phi(x)\right)$. They then extend this to the universality of deepsets since ρ and ϕ can be parameterized as neural networks, which can be universal function approximators. The deepsets formalism allows point-estimates for regression parameters to be obtained following an aggregation of features in a potentially variably-sized set of data. Incorporating our formalism, each set member $\mathbf{d}_i$ is first passed to a neural network $\phi(\mathbf{d}_i; w_1)$, and subsequently aggregated using some permutation-invariant scheme, $\bigoplus_i$.

$$\hat{\boldsymbol{\theta}} = \rho\left(\bigoplus_{i=1} \phi(\mathbf{d}_i; w_1); \quad w_2\right), \tag{8.29}$$

where ϕ is the embedding network parameterised by weights w_1 and ρ is the "global" network with weights w_2 that maps aggregated features to predicted parameters. When the aggregation is chosen to be the mean, the deepsets formalism is scalable to arbitrary data and becomes equivalent to the Fishnets aggregation formalism *with flat weights across the aggregated data*. The loss takes the form of a convex squared loss, e.g. the mean square error

$$\mathcal{L} = \frac{1}{n_{\text{batch}}} \sum_{i}^{n_{\text{batch}}} (\hat{\boldsymbol{\theta}}_i - \boldsymbol{\theta}_i)^2 \tag{8.30}$$

where n_{batch} is a batch of full simulations, each of size .

Training and Generalization. In practice, Deepsets requires a *fixed* aggregation scheme from which to learn its global function. Most often this is a summary of embedding layers $\phi(x)$. For networks to scale to arbitrary dataset cardinality, aggregations like max, mean, and variance need to be used. In a scenario where the training data distribution $p(x, \theta)$ follows a different distribution from the training data, these aggregations might pose an issue. Concretely, consider $x \sim \mathcal{N}(\mu, 1)$, with the target quantity $\theta = \mu \sim \mathcal{U}(0, 2)$. Next consider a deepset with the identity embedding layer $\phi(x) = x$ and mean-aggregation:

$$\hat{\mu} = \rho\left(\frac{1}{n}\sum_i x_i\right) \tag{8.31}$$

If test data x_i were drawn from the same distribution as the test data, ρ would act on the mean value of the set of data, in this case $\rho(\mathbb{E}_{p(x,\theta)}[x_i])$, and would converge to a *learned function of the joint*

prior-data distribution $p(x, \theta)$. However, if a test set of data were drawn from a different distribution, e.g. $\mu_{\text{test}} \sim \mathcal{N}(0.5, 0.1)$, then the expectation $\mathbb{E}_{p(x,\theta)}$ would take on a different value, and ρ would return an incorrect result for the deterministic aggregation. Here it is important to emphasize that $p^{\text{test}}(x, \theta)$ and $p(x, \theta)$ overlap along the same support, meaning the network *will have seen examples of data drawn from this prior* in the limit of an infinite training set. However, the fixed aggregation makes use of a training-data distribution-dependent quantity for its mapping, which can be skewed under covariate shift or different noise settings.

'Don't play what's there, play what's not there'

Miles Davis

HYBRID STATISTICS

The previous research chapters have demonstrated how new statistical techniques paired with neural networks can produce both optimal and, in some cases, lossless statistics. As seen in Chapter 7, the *true* cosmological parameter information content contained in dark matter simulations is unknown, but by exploiting symmetries and new, sparse data manifolds, improved, combined nonlinear statistics can be obtained from the structure itself. Chapter 8 sought to improve generic information capture in sets and graphs. There we demonstrated that learned aggregations could indeed be made *lossless* with respect to known set-based likelihoods, and improve information capture in arbitrary graph datasets with smaller networks. These studies exploit symmetries in the data-network connection to find useful statistics efficiently.

We now turn to an investigation concerning the *information objective*–can the loss function that the compression network is optimised with be made more efficient by leveraging something that we already know about the data ? By specifying such an existing function of the data, is there an information-theoretic way that we can tell the new network where to look, to *harmonise* with the existing information ? Moreover, this approach might shed some light onto the persistent question surrounding neural networks: "Where is the information coming from ?"

The next three chapters will introduce the concept of *hybrid summary statistics*, and tested in the context of cosmological parameter estimation.

Part I (Makinen et al., 2025) introduces the concept by prying open Fisher information maximisation as an "information update" objective with respect to an existing statistic. The hybrid objective improves information capture in noisy weak lensing map compression, especially as simulations become increasingly non-Gaussian.

Part II (incorporating Makinen et al., 2024; Makinen et al., in prep.c) extends the hybrid statistics formalism beyond local Fisher compression to a learned compression over the joint distribution

$p(\mathbf{d}, \theta)$, using Mutual Information maximisation to express the optimisation objective. The hybrid objective improves network information extraction beyond the two-point function, especially in simulation-limited regimes. The formalism also extends to compression of multiple data products under a common objective, such as statistically-independent patches of a large sky survey.

Part III (Makinen et al., in prep.c) applies the hybrid formalism to weak lensing and Dark Energy equation of state measurement with the Dark Energy Survey (DES) Y3 dataset. The scheme uses the same setup as Jeffrey et al. (2024), but with a hierarchical hybrid statistic loss. This change in objective induces remarkable improvement in w measurement from mock datasets. It is forecast to produce a state-of-the art Dark Energy measurement on the real data.

CHAPTER 9

HYBRID STATISTICS (PART I)

Hybrid summary statistics: neural weak lensing inference beyond the power spectrum

T. Lucas Makinen[1], Natalia Porqueres[2], Alan Heavens[1], Tom Charnock[3], Axel Lapel[4,5], Benjamin D. Wandelt[4,6]

[1]Imperial Centre for Inference and Cosmology (ICIC) & Astrophysics Group, Imperial College London, Blackett Laboratory, Prince Consort Road, London SW7 2AZ, United Kingdom

[2]Department of Physics, University of Oxford, Denys Wilkinson Building, Keble Road, Oxford OX1 3RH, United Kingdom

[3]Freelance consultant in statistical modelling

[4]Sorbonne Université, CNRS, UMR 7095, Institut d'Astrophysique de Paris, 98 bis boulevard Arago, 75014 Paris, France

[5]Sorbonne Université, Université Paris Diderot, Sorbonne Paris Cité, CNRS, Laboratoire de Physique Nucléaire et de Hautes Energies (LPNHE). 4 place Jussieu, F-75252, Paris Cedex 5, France

[6]Center for Computational Astrophysics, Flatiron Institute, 162 5th Avenue, New York, NY 10010, USA

Accepted to the Journal of Cosmology and Astroparticle Physics; Makinen et al. (2025)

Abstract

Cosmological inference relies on compressed forms of the raw data for analysis, with traditional methods exploiting physics knowledge to define summary statistics, such as power spectra, that are known to capture much of the information. An alternative approach is to ask a neural network to find a set of informative summary statistics from data, which can then be analysed either by likelihood- or simulation-based inference. This has the advantage that for non-Gaussian fields, they may capture more information than two-point statistics. However, a disadvantage is that the network almost certainly relearns that two-point statistics are informative. In this paper, we introduce a new hybrid method, which combines the best of both: we use our domain knowledge to define informative physics-based summary statistics, and explicitly ask the network to augment the set with extra statistics that capture information that is not already in the existing summaries. This

yields a new, general loss formalism that reduces both the number of simulations and network size needed to extract useful non-Gaussian information from cosmological fields, and guarantees that the resulting summary statistics are at least as informative as the power spectrum. In combination, they can then act as powerful inputs to implicit inference of model parameters. We use a generalisation of Information Maximising Neural Networks (IMNNs) to obtain the extra summaries, and obtain parameter constraints from simulated tomographic weak gravitational lensing convergence maps. We study several dark matter simulation resolutions in low- and high-noise regimes. We show that i) the information-update formalism extracts at least 3× and up to 8× as much information as the angular power spectrum in all noise regimes, ii) the network summaries are highly complementary to existing 2-point summaries, and iii) our formalism allows for networks with extremely lightweight architectures to match much larger regression networks with far fewer simulations needed to obtain asymptotically optimal inference.

9.1 Introduction

Weak gravitational lensing alters the trajectories of distant photons as they pass through the large-scale structure of visible and dark matter to our detectors. These deflections distort the observed shapes of galaxies, whose patterns can be used to trace the matter distribution in between, and are sensitive to parameters that describe the expansion history and structure formation of the Universe. The inference of these parameters from cosmological weak lensing surveys is usually performed using two-point statistics of the lensing images, such as shear correlation functions or power spectra. Recent analyses include the Dark Energy Survey (Amon et al., 2022; Secco et al., 2022), the Kilo-Degree Survey (KiDS, Li et al., 2023; Asgari et al., 2021) and the Hyper Suprime-Cam survey (Dalal et al., 2023). However, two-point statistics do not fully describe the rich non-Gaussian features present in large-scale structure, where more cosmological information might be found.

Implicit inference (also known as simulation-based inference or likelihood-free inference) has made it possible to utilise higher-order statistics derived from simulations (see e.g. Cranmer et al. (2020a) for a review), and circumvent the need for an explicit likelihood function, which can be challenging to compute via Bayesian Hierarchical Models (Alsing et al., 2016; Porqueres et al., 2021a,b, 2023; Loureiro et al., 2023; Sellentin et al., 2023). In weak gravitational lensing for example, even (incorrectly) assuming that the underlying cosmological density is Gaussian, the two-point statistics that

describe this field can themselves have significantly non-Gaussian sampling distributions (Sellentin & Heavens, 2017; Alsing et al., 2016; von Wietersheim-Kramsta et al., 2024). The question arises as to which additional statistics contain significant extra information about the parameters, beyond that which is present in the two-point functions. Higher-order statistics are an obvious choice, but they suffer from a lack of knowledge of their sampling distributions, and the very large number of statistics make them cumbersome for implicit inference. However, advances in deep learning have made it possible to learn highly-informative neural compressions for massive simulations automatically, and in some cases losslessly (Makinen et al., 2021; Charnock et al., 2018; Makinen et al., 2022). These compressions yield radically smaller summary spaces, which are ideal for implicit inference, and which can be used for Bayesian posterior estimation via accept-reject or density estimation strategies (Alsing et al., 2019).

These new advances have made simulation-based studies for weak lensing a popular avenue of research in recent years. This includes studies of peak counts (Peel et al., 2017; Zürcher et al., 2022; Kratochvil et al., 2010; Lanzieri et al., 2023), higher-order statistics (Euclid Collaboration et al., 2023) such as wavelet scattering transformations (Cheng et al., 2020, 2024), Fourier-space normalising flows (Dai & Seljak, 2024), and field-based convolutional neural networks (Ribli et al., 2019; Fluri et al., 2018, 2019; Sharma et al., 2024). The ultimate goal is to obtain statistics that exhaust the information content of the observed weak lensing field. This can be done explicitly (assuming a likelihood for shear or convergence voxels) via field-level sampling (Alsing et al., 2016; Ramanah et al., 2019; Porqueres et al., 2021b,a, 2023; Leclercq, 2015; Jasche et al., 2015; Boruah & Rozo, 2023; Loureiro et al., 2023; Tsaprazi et al., 2023).

Impicit inference approaches have matured enough for real-data analysis, beginning with Fluri et al. (2022), who analysed map-level KiDS data with an assumed Gaussian summary likelihood, and more recently von Wietersheim-Kramsta et al. (2024), who reproduced C_ℓ constraints with an implicit likelihood. Jeffrey et al. (2024) presented a Dark Energy Survey data analysis using a convolutional neural network compression of the full mass map and demonstrated marked improvement over existing power spectrum and peak count constraints on the same dataset.

This work seeks to demonstrate an improved optimisation strategy and add a new layer of interpretability to this growing body of literature. A common question for deep learning and implicit inference practitioners is what features are being learned from the data by neural approaches. Makinen et al. (2022) showed explicitly that neural networks trained on halo catalogues identified features

that could be linked to explicitly-understood cosmological distributions such as halo mass and correlation functions. Here we respond to this question by modifying our optimisation criterion such that a network only outputs statistics obtained from the data that work alongside to an existing statistic, in this case the weak lensing angular power spectrum. To be explicit, we train the network to maximise the extra Fisher information that is not already present in the power spectrum. We term these neural summaries "hybrid" statistics since they combine new and existing functions of the data. We make our constraint comparison within a completely simulation-based setup to interrogate the information content of the respective summaries in a sampling distribution-agnostic way.

This paper is organised as follows: In Sections 9.2 and 9.3 we present our general formalism for finding optimal new summaries from simulated data given some existing descriptive statistics, and describe how the strategy can be implemented automatically with Information Maximising Neural Networks (IMNNs, Charnock et al., 2018; Makinen et al., 2021). In Section 9.4, we describe our mock weak lensing formalism and present the simulation suite details. In Section 9.5, we present our angular C_ℓ compression scheme as the existing statistic in the information-update formalism. We then present our physics-inspired, lightweight neural network architecture designed to find optimal additional summaries. In Section 9.6.1, we make comparisons of information gain over the power spectrum as a function of resolution and increased shape noise, and show that our optimisation scheme captures physical features in realistic noise regimes.

9.2 Implicit Inference

The goal of most science experiments is to obtain data $\mathbf{d}$ with which to test models that describe the data generation process. In cosmology, this often boils down to obtaining a posterior distribution for some model parameters $\boldsymbol{\theta}$: $p(\boldsymbol{\theta}|\mathbf{d}) \propto p(\mathbf{d}|\boldsymbol{\theta})p(\boldsymbol{\theta})$, which requires knowledge of the likelihood $p(\mathbf{d}|\boldsymbol{\theta})$, and the assumption of a suitable prior $p(\boldsymbol{\theta})$. This data-generating distribution is often too complicated to evaluate for inference, or too complex to write down analytically.

9.2.1 Density Estimation

Simulation-based inference circumvents the need for a tractable likelihood $p(\mathbf{d}|\boldsymbol{\theta})$, and instead seeks to parameterise the underlying, implicit likelihood or posterior present in forward simulations of the

data. Neural density estimators (NDEs; e.g. Bishop, 1994) use neural networks that give some estimate $q(\boldsymbol{\theta}, \mathbf{d}; \varphi)$ of the desired posterior (or likelihood) by varying weights and biases (parameterised as φ) to minimize the loss

$$U(\varphi) = -\sum_{i=1}^{N} \ln q(\boldsymbol{\theta}_i | \mathbf{d}_i; \varphi), \tag{9.1}$$

over batches of parameter-data samples drawn from the joint distribution $(\boldsymbol{\theta}_i, \mathbf{d}_i) \curvearrowleft p(\boldsymbol{\theta}, \mathbf{d}_i)$. This loss is equivalent to minimising the Kullback-Leibler divergence between the target distribution and q (Kullback & Leibler, 1951). In this work we employ Masked Autoregressive Flows (MAF; Papamakarios et al., 2017) to model the posterior distribution directly. We detail our implementation in Section 9.5.3. Density estimation also makes posterior predictive and coverage tests far easier to perform. We show examples of these tests in Appendix 9.10.

9.2.2 Data Compression

Accept-reject and density estimation schemes become more difficult to compare to a target, observed data vector $\mathbf{d}_{\rm obs}$ the larger the dimensionality $\dim(\mathbf{d})$, which posits the need for data compression to some smaller summary space $\mathbf{x}$. We would like a function $f : \mathbf{d} \mapsto \mathbf{x}$ that is ideally maximally informative about the parameters $\boldsymbol{\theta}$. Under certain conditions, $f(\mathbf{d})$ can yield a sufficient statistic, for which the dimension $\mathbf{x}$ is equal to the number of parameters, e.g. $\dim(\mathbf{x}) = \dim(\boldsymbol{\theta})$. Heavens et al. (2000) introduced Massively Optimised Parameter Estimation and Data (MOPED) compression, which gives optimal score compression for cases where the likelihood and sampling distributions are Gaussian, and this was generalised to other forms of score compression by Alsing & Wandelt (2018); Carron & Szapudi (2013); Hoffmann & Onnela (2023). Neural compression is a popular scheme for learning mappings agnostic to sampling distributions, for which several optimisation schemes have been proposed. Regression-style approaches learn a compression $f(\mathbf{d}; w)$ parameterised by (neural) weights w via a loss, for example quadratic, over parameter-data pairs over a prior, using variants of the mean square error (MSE), or in some cases learning f and the neural posterior (via Eq. 11.14) simultaneously, dubbed Variational Mutual Information Maximisation (VMIM; Jeffrey et al., 2020). Sharma et al. (2024) recently compared these losses paired with convolutional networks for a separate weak lensing simulation suite.

This work builds upon the Information Maximising Neural Network (IMNN) approach (Charnock et al., 2018; Makinen et al., 2021), which prescribes a neural compression that maximises the determi-

nant of the Fisher matrix of summaries around a *local* point in parameter space. A compression is i) learned at a fiducial point from a set of dedicated fiducial and derivative simulations and then ii) applied to simulations over a prior for posterior construction. This approach has numerous advantages, namely:

1. An asymptotically-optimal compression can be learned from simulations around a single point in parameter space.
2. The compression automatically and simultaneously gives Fisher posterior constraint forecasts.
3. Priors used for density estimation are decoupled from the compression step and can be chosen after learning the compression.
4. Adding additional parameters of interest to the compression learning scheme only requires relatively small numbers of extra simulations for the new derivatives; e.g. the distribution $p(\boldsymbol{\theta}, \mathbf{d})$ need not be re-simulated.

In the following section we will extend this approach to find new (neural) data compressions that only increase information about parameters above what is already present in a set of existing statistics such as the power spectrum.

9.3 How to Choose an Optimal New Summary

Consider some data $\mathbf{d} \in \mathbb{R}^N$ created from parameters $\boldsymbol{\theta}$ that can be summarised in a compressed summary vector via a function $h : \mathbf{d} \mapsto \mathbf{t}$ with $\mathbf{t} \in \mathbb{R}^{n_t}$ where $n_t < N$. We can estimate the covariance matrix of the summaries $\mathbf{C}_t$, and the mean $\boldsymbol{\mu}_t$ from simulations, along with derivatives with respect to parameters of the mean $\boldsymbol{\mu}_{,\theta_i}$. Assuming for now that the summaries have a Gaussian sampling distribution (this assumption is temporary and is dropped in the inference phase), we can compute the Fisher information of the observables via

$$[\mathbf{F}_t]_{ij} = \boldsymbol{\mu}^T_{,\theta_i} \mathbf{C}_t^{-1} \boldsymbol{\mu}_{,\theta_j} \tag{9.2}$$

where we introduce the notation $\boldsymbol{y}_{,\theta_i} \equiv \partial \boldsymbol{y} / \partial \theta_i$ for partial derivatives with respect to parameters. The Fisher information matrix here describes how much information $h(\mathbf{d})$ contains about the model

parameters, and is given as the second moment of the score of the likelihood with respect to h, assuming a parameter-independent, Gaussian covariance of the statistic $\mathbf{t}$.[1] A large Fisher information for a function of the data indicates that the mapping to $\mathbf{t} = h(\mathbf{d})$ is very informative about the model parameters used to generate the realisation of data $\mathbf{d}$. Fisher forecasting for a given model is made possible by the information inequality and the Cramér-Rao bound (Cramér, 1946; Rao, 1945), which states that the minimum variance of the value of an estimator $\boldsymbol{\theta}$ is given by

$$\langle(\theta_i - \langle\theta_i\rangle)^2\rangle \geq (\mathbf{F}^{-1})_{ii}, \tag{9.3}$$

with no summation over i.

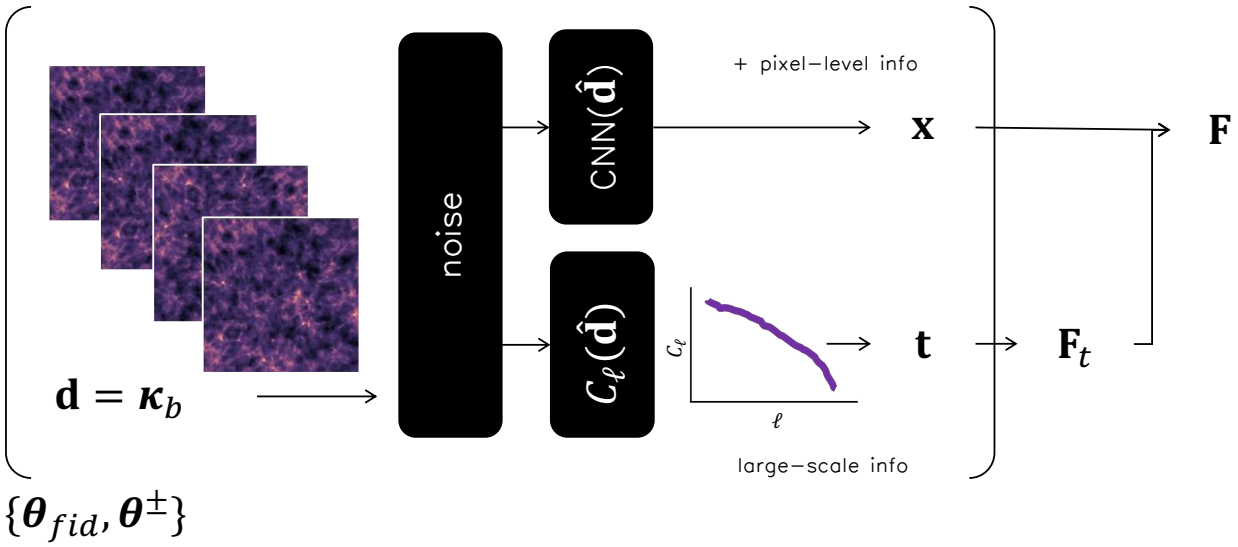


Figure 9.1: Hybrid summary network schematic, illustrated for weak gravitational lensing. Noisy data (weak lensing $\boldsymbol{\kappa}_b$ with shape noise) are passed in parallel to an existing summary function (tomographic C_ℓ with optional MOPED compression) to produce summaries $\mathbf{t}$, and a network (CNN) to output an additional set of summaries $\mathbf{x}$, described in Section 9.5 and illustrated in Figure 9.2. To train the network the Fisher information is first calculated for $\mathbf{t}$ and then updated via Equation 9.11 to yield $\mathbf{F}$, for the loss Eq. 9.14.

We would next like to add new summary statistics to increase the information over what is present in $\mathbf{t}$. For clarity, we begin by adding a single summary x, before generalising to multiple additional summaries. We do this via some function $f : \mathbf{d} \mapsto x$. This new number has variance σ_x^2 and when concatenated to the old observables gives the mean vector

$$\boldsymbol{\mu} = [\boldsymbol{\mu}_t, \langle x\rangle]^T. \tag{9.4}$$

[1]Note that the Gaussian assumption is used here only to define a compression; once the summaries are defined, SBI no longer assumes Gaussianity. If the compressed summaries are not Gaussian-distributed, the compression will be suboptimal but the downstream SBI analysis will implicitly determine and use their true sampling distribution.

For mean-subtracted quantities Δt and Δx, the covariance vectors between old and new observables can be computed (e.g. from simulations) as $[\mathbf{u}]_i = \langle \Delta t_i \Delta x \rangle$, which yields the full covariance matrix

$$\mathbf{C} = \begin{pmatrix} \mathbf{C}_t & \mathbf{u} \\ \mathbf{u}^T & \sigma_x^2 \end{pmatrix}. \tag{9.5}$$

Notice here that the smaller the values of $\mathbf{u}$, the less correlated x is with $\mathbf{t}$. The full updated Fisher matrix is then

$$F_{ij} = \boldsymbol{\mu}_{,\theta_i}^T \mathbf{C}^{-1} \boldsymbol{\mu}_{,\theta_j}. \tag{9.6}$$

With some rearrangement, we obtain the fast Information-Update Formula (IUF):

$$\mathbf{F} = \mathbf{F}_t + \frac{1}{s} \mathbf{v}\mathbf{v}^T, \tag{9.7}$$

with $[\mathbf{v}]_i = \langle x \rangle_{,\theta_i} - \boldsymbol{\mu}_{,\theta_i}^T \mathbf{C}_t^{-1} \mathbf{u}$ and $s = \sigma_x^2 - \mathbf{u}^T \mathbf{C}_t^{-1} \mathbf{u}$. This calculation is only $\mathcal{O}(n_{\text{params}}^2 + n_{\text{params}} d + d^2)$ operations where $d = \dim(\mathbf{t})$, which is asymptotically d times faster than Eq. 9.6 when $d >> n_{\text{params}}$. This formalism also yields a fast update for the determinant of the new Fisher matrix:

$$\ln \det \mathbf{F} = \ln \det \mathbf{F}_t + \ln \left(1 + \frac{1}{s} \mathbf{v}^T \mathbf{F}_t^{-1} \mathbf{v} \right). \tag{9.8}$$

Interpretation. The updated Fisher information in Equation 9.7 clearly separates the information contribution from the existing observables in the first Fisher term and the new observables in the second term. An optimal, "complementary" new observable x adds a lot of information if it has highly correlated measurement error with the existing summaries $\mathbf{t}$, but changes with respect to parameters in a way that is as distinguishable as possible from how x and $\mathbf{t}$ change together.

Multiple New Observables. The IUF can be naturally extended to a vector of new summaries $\mathbf{x}$. We promote $\mathbf{v}$ to a matrix

$$[\boldsymbol{V}]_{ij} = \langle \mathbf{x}_j \rangle_{,\theta_i} - \boldsymbol{\mu}_{,\theta_i}^T \mathbf{C}_t^{-1} \mathbf{u}_j, \tag{9.9}$$

and the scalar s generalises to the matrix

$$\boldsymbol{\Sigma} = \mathbf{C}_x - \mathbf{U}^T \mathbf{C}_t^{-1} \mathbf{U} \tag{9.10}$$

where $[\mathbf{U}]_{ij} = [\mathbf{u}_j]_i$ and $\mathbf{C}_x$ is the covariance of network outputs $\mathbf{x}$. Altogether the updated Fisher matrix for a vector of extended summaries is

$$\mathbf{F} = \mathbf{F}_t + \boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{V}^T. \tag{9.11}$$

9.3.1 Finding a New Summary With a Neural Network

We can find a new observable $\mathbf{x}$ by optimising the IUF equation (9.7) with a neural network $f : \mathbf{d} \mapsto \mathbf{x}$ that operates on the data. This formalism folds neatly into the IMNN formalism (Charnock et al., 2018; Makinen et al., 2021, 2022). We illustrate this procedure for weak lensing data in a schematic in Figure 9.1. We can choose to optimise Equation 9.6 directly, but Equation 9.11 is less computationally expensive for large covariance matrices and more than one additional summary. The ingredients needed to compute the components of the loss are a suite of n_s simulations at a fiducial value of parameters $\{\mathbf{d}\}_{i=1}^{n_s}|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\rm fid}}$ and a set of n_d seed-matched simulations at perturbed values of each parameter $\boldsymbol{\theta}_i^{\pm} = \boldsymbol{\theta}_i \pm \Delta\boldsymbol{\theta}_i$ holding all other parameters fixed at their fiducial values. Using this finite difference gradient dataset the partial derivatives of a data summary function $Q(\mathbf{d})$ with respect to parameters is

$$\left(\frac{\partial\hat{\mu}_i}{\partial\theta_\alpha}\right) \approx \frac{1}{n_d}\sum_{i=1}^{n_d}\frac{Q(\mathbf{d}_i^+) - Q(\mathbf{d}_i^-)}{\theta_\alpha^+ - \theta_\alpha^-}. \tag{9.12}$$

For $n_{\rm params}$ summaries, this method requires $n_d \times n_{\rm params} \times 2$ simulations with n_d unique random seeds alongside the n_s simulations at the fiducial point required for the covariance. This is done for the mean of both existing and new summary statistics, consolidated as $\mathbf{y} = [\mathbf{t}, \mathbf{x}]$. The covariance of the (existing and new) summaries is estimated from the data as well, using n_s simulations at $\boldsymbol{\theta}_{\rm fid}$:

$$\hat{\mathbf{C}}_{\alpha\beta} = \frac{1}{n_s - 1}\sum_{i=1}^{n_s}(\mathbf{y}_i - \bar{\mathbf{y}})_\alpha(\mathbf{y}_i - \bar{\mathbf{y}})_\beta, \tag{9.13}$$

where $\bar{\mathbf{y}}$ is the average over the simulations at the fiducial point. The full covariance can be broken down into (or estimated separately by) its components $\mathbf{C}_t, \mathbf{C}_x$, and $\mathbf{u}$ according to Eq. 9.5. Note that this covariance is assumed to be independent of the parameters, which, whilst not strictly true, is enforced by regularisation during the fitting of a network. If it does not hold, it simply makes the compression suboptimal elsewhere in the parameter space. Crucially, both old and new summaries and their statistics must be computed on the same (noisy) simulations to correctly distinguish between noise fluctuations and newly-informative features of the data during optimisation.

Optimising a neural function $\mathbf{x} = f(\mathbf{d}; w)$ to maximise the determinant of $\mathbf{F}$ from Eq. 9.11 forces the new summaries $\mathbf{x}$ to add complementary information to the existing summaries' Fisher contributions. As described in Charnock et al. (2018) and Livet et al. (2021), the Fisher information is invariant to nonsingular linear transformations of the summaries. To remove this ambiguity, a term penalising the network summary covariance $\mathbf{C}_x$ is added. This gives the loss function

$$\Lambda = -\ln\det\mathbf{F} + \lambda\frac{1}{2}\operatorname{tr}\mathbf{C}_x \tag{9.14}$$

where λ is a regularising coefficient. This scalar loss function can be optimised via gradient descent to update weights w for the network's contributions to the combined Fisher information. The network can do no worse at summarising the data than the existing summary, since the loss only optimises the second term of Eq. 9.11. With the updated Fisher information we can also compute quasi-maximum likelihood estimates (MLE) for the parameters for a given mean-subtracted summary vector $\Delta = [\Delta\mathbf{t}, \Delta\mathbf{x}]$ (Alsing & Wandelt, 2018):

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_{\rm fid} + \mathbf{F}^{-1}\boldsymbol{\mu}_{,\theta_i}\mathbf{C}^{-1}\Delta^T. \tag{9.15}$$

We then use these hybrid statistics (as they are functions of the data) as our highly-informative and extremely compressed data set, ideal for simulation-based or implicit inference. The resulting compression is *asymptotically* optimal at the fiducial point in parameter space, but for smoothly-varying data manifolds results in a smooth summary space that can be exploited for neural posterior estimation away from the fiducial point as described in Makinen et al. (2021); Charnock et al. (2018). A useful aspect of learning this local compression is that a prior for posterior estimation can be specified after the compression network is trained, unlike regression networks.

9.4 Weak Gravitational Lensing

9.4.1 Formalism

The effect of weak lensing (WL) on a source field is defined by its shear, γ, which captures the distortions in the shapes of observed galaxies. In the flat-sky limit in Fourier space, this observable

can be related to the convergence field κ observable, which describes variation in angular size:

$$\tilde{\gamma}(\boldsymbol{\ell}) = \frac{(\ell_1 + i\ell_2)^2}{\ell^2}\tilde{\kappa}(\boldsymbol{\ell}) \tag{9.16}$$

where $\boldsymbol{\ell} = (\ell_1, \ell_2)$ is the complex wavevector. The convergence field can be connected to the underlying dark matter field by integrating the fractional overdensity along the line-of-sight to give (Kilbinger, 2015):

$$\kappa(\boldsymbol{\vartheta}) = \frac{3H_0^2\Omega_m}{2c^2}\int_0^{r_{\rm lim}} \frac{rdr}{a(r)} g(r)\delta^f(r\boldsymbol{\vartheta}, r), \tag{9.17}$$

where $\boldsymbol{\vartheta}$ denotes the coordinate on the sky, r is the comoving distance, $r_{\rm lim}$ is the galaxy survey's maximum comoving distance, δ^f is the dark matter overdensity field at scale factor a, and, for a flat Universe

$$g(r) = \int_r^{r_{\rm lim}} dr' n(r') \frac{r - r'}{r}, \tag{9.18}$$

is the integration of the redshift distribution $n(r)$ in the given comoving shell. In real-data analyses the data will be the cosmic shear, but here we restrict our analysis to noisy convergence maps. The forward model to generate $\boldsymbol{\kappa}$ consists of a cosmological parameter draw, $\boldsymbol{\theta}$, which is used to generate primordial fluctuations, $\delta^{\rm ic}$. Here the initial conditions are a Gaussian random field governed by the Eisenstein & Hu (1999) cosmological power spectrum $P(k;\theta)$, which includes baryonic acoustic oscillations (BAO). The cosmic initial conditions are then evolved forward via a specified non-linear gravity model $G(\delta^{\rm ic})$, which describes the growth of the large-scale structure (LSS). The evolved dark matter field δ^f is then used to generate the convergence field. Using the Born Approximation, we implement a discrete version of Equation 9.17 using a summation over voxels to approximate the radial line-of-sight integrals:

$$\kappa^b_{mn} = \frac{3H_0^2\Omega_m}{2c^2}\sum_{j=0}^{N}\delta^f_{mnj}\left[\sum_{s=j}^{N}\frac{(r_s - r_j)}{r_s}n^b(r_s)\Delta r_s\right]\frac{r_j\Delta r_j}{a_j}, \tag{9.19}$$

where b indexes the tomographic bin, and m, n index the spatial pixels on the sky. The index j indicates the voxel along the line-of-sight at the comoving distance r_j. The total number of voxels along the line-of-sight, N, is obtained from a ray tracer. Δr_j is the length of the line segment inside voxel j, and δ^f_{mnj} is the discretized dark matter overdensity field. The comoving radial distance r_s is the distance to the source. Each tomographic bin has a source redshift distribution $n^b(z_s)$. Once κ^b_{mn} is computed, the convergence field $\mathbf{d} = \{\hat{\kappa}^b_{mn}\}$ is obtained by adding uncertainties equivalent to the shape noise (and measurement error) in the shear field. This is captured by zero-centred Gaussian

white noise added pixel-wise with variance

$$\sigma_n^2 = \sigma_\epsilon^2 \frac{N_{\rm tomo}}{n_{\rm gal} A^b_{\rm pixel}}, \tag{9.20}$$

where σ_ϵ^2 is the total galaxy intrinsic ellipticity dispersion, $n_{\rm gal}$ is the source galaxy density on the sky, and $A^b_{\rm pixel}$ is the angular size of the pixel in each tomographic bin. For Stage-IV weak lensing surveys like Euclid $n_{\rm gal}$ will be ~ 30 arcmin^{-2} and $\sigma_\epsilon \simeq 0.3$ (Euclid Collaboration, 2022). For network training purposes we introduce an amplitude scaling parameter $\sigma_n' = A\sigma_n$ that we report in terms of effective source galaxy density.

9.4.2 Simulation Details

We analyse several simulation suites at different resolutions to conduct our experiments. In all cases our physical box size is kept fixed at $L_x = L_y = 250$ Mpc h^{-1} and $L_z = 4000$ Mpc h^{-1} in comoving units, in a pixel grid of shape $(N_x, N_y, N_z) = (N, N, 512)$, where we vary N to probe changing gravity solver scales. We utilise `pmwd` particle mesh (PM) simulations (Li et al., 2022) integrated for 63 timesteps to generate the nonlinear dark matter overdensity field for $N = [64, 128]$ resolution and 100 timesteps for $N = 192$ resolution. This controls the particle spacings L/N which probe increasingly nonlinear scales described by the PM simulations. We compute the line-of-sight integral in comoving units before binning the L_z dimension in redshift bins converted to comoving units via the cosmology-dependent change of variable. For this analysis we do not include lightcone effects. We choose our four tomographic redshift bins to be Gaussian, centred at $z = [0.5, 0.75, 1.0, 1.25]$ with width $\sigma_z = 0.14$, following Porqueres et al. (2021a). The resulting convergence fields span a 3.58×3.58 deg^2 field of view. Shape noise is added to the noise-free simulations before computing two-point or network statistics as described below. We generate two distinct datasets to i) construct a locally optimal compression and ii) perform posterior density estimation.

Compression Simulations. For a given resolution we generate two equally-sized datasets for training and validation of our network compression. To calculate network and two-point covariances described below we simulate $n_s = 1500$ simulations at a fiducial cosmology $\boldsymbol{\theta}_{\rm fid} = (\Omega_m, S_8) = (0.3, 0.8)$. For finite-difference derivatives we simulate $n_d \times 2 \times n_p = 375 \times 2 \times 2$ seed-matched simulations at a perturbed parameter set $\boldsymbol{\theta} \pm \Delta\boldsymbol{\theta}_{\rm fid} = \boldsymbol{\theta}_{\rm fid} \pm (0.0115, 0.01)$. All other cosmological parameters were held fixed at Planck 2018 parameters (Planck Collaboration et al., 2020). The total number of

simulations used for optimal compression is thus 4500.

Density Estimation Simulations. Because our compression from the convergence field data is learned locally, we are free to choose our prior guided by the compression method's Fisher forecast. We simulate 5000 simulations over a uniform prior in (Ω_m, S_8), whose width is chosen according to the strategy described in Section 9.5.3.

9.5 Finding Hybrid Weak Lensing Statistics

The information-update formula is perfectly suited to improving weak lensing $\Omega_m - \sigma_8$ parameter constraints with neural summaries. We would like to know if more cosmological parameter information can be extracted from the convergence field beyond a simulation-based tomographic angular C_ℓ statistic analysis, and in what resolution regimes. We present the MOPED scheme for angular C_ℓ compression and information-update neural network architecture.

9.5.1 MOPED Angular C_ℓ Compression

Here our existing summaries are either binned angular C_ℓ or MOPED-compressed vectors $\mathbf{t}$ (without the optional Gram-Schmidt orthogonalisation employed in Heavens et al. (2000)). We outline our setup below and display a schematic of the architecture in Figure 9.1.

We compute empirical auto- and cross-spectra C_ℓ across the four noisy tomographic bins, resulting in 10 C_ℓ vectors. To test scale-dependent information, we optionally apply a maximum $\ell_{\rm cut}$ to each vector to mimic existing survey analyses. To reduce the number of simulations needed to estimate the covariance matrix, we bin each spectrum into six evenly-spaced ℓ bins weighted by C_ℓ value. We can estimate the covariance of the C_ℓ vector using the n_s fiducial simulations, and the finite difference derivatives with respect to each parameter via Eq. 9.12. Together, the Fisher matrix for these summaries is computed with Eq. 9.2. With these ingredients we can then perform the MOPED compression from mean-subtracted vectors evaluated at a fiducial set of parameters $\Delta = \left(\hat{C}_\ell - \langle C_\ell \rangle_{\rm fid}\right)$ down to score summaries $\mathbf{t}$:

$$\mathbf{t} = \boldsymbol{\theta}_{\rm fid} + \boldsymbol{\mu}_{,\theta_i} \mathbf{C}_t^{-1} \Delta^T. \tag{9.21}$$

which can then be rescaled by the Fisher matrix to obtain an MLE of the parameters (Alsing & Wandelt, 2018):

$$\hat{\boldsymbol{\theta}}_{\rm MOPED} = \boldsymbol{\theta}_{\rm fid} + \mathbf{F}_t^{-1}\mathbf{t}. \tag{9.22}$$

In practice, we replace $\mathbf{t}$ with $\hat{\boldsymbol{\theta}}_{\rm MOPED}$ as our existing MOPED statistics, which has covariance $\mathbf{C}_t = \mathbf{F}_t^{-1}$. These compressed summaries are the default C_ℓ-based summaries that we feed into the normalising flow posterior estimation scheme (Section 9.5.3), as normalising flows are not guaranteed to work well with large inputs e.g. the 60-dimensional binned C_ℓ vector (Cranmer et al., 2020a). For network optimisation however, the longer, binned power spectrum vector can be used to find $\hat{\boldsymbol{\theta}}_{\rm network}$. Changes in the ℓ bins with respect to noise and parameters increases the number of cross-correlations ($\mathbf{u}$) with network summaries and encourage improved information extraction. We explore this option for training in noisy settings in Section 9.6.2. Both choices of statistics fit neatly into the existing statistic formalism described in Section 9.3.

9.5.2 Physically-Informed Neural Network Architecture

We design a novel, lightweight 2D convolutional neural network that is constrained by physics knowledge, namely that a) physical laws are translation-invariant and direction-invariant, and b) the non-Gaussian cosmic shear information is found in clustering patterning on small scales. This motivates both the CNN approach, and the use of a convolution kernel whose complexity is truncated to a multipole expansion around a circularly-symmetric kernel. We present this network layer-by-layer and display a schematic in Figure 9.2.

The inputs to the network are the log-transformation of the convergence maps at the four specified tomographic bins, adapted from Simpson et al. (2015); Seo et al. (2010); Joachimi et al. (2011):

$$\boldsymbol{\kappa}^b = \kappa_o \ln\left[1 + \boldsymbol{\kappa}^b/\kappa_o\right] \tag{9.23}$$

where $\kappa_o = |\kappa^b_{\rm min}| + 0.01$, where $\kappa^b_{\rm min}$ is the minimum convergence value for the tomographic bin b.

Multipole Kernel Embedding. For convergence maps we can target clustering information by learning convolution functions of the data with certain, enforced symmetries. Kodi Ramanah et al. (2020) showed via neural emulation of dark matter simulations that learned convolutional weights

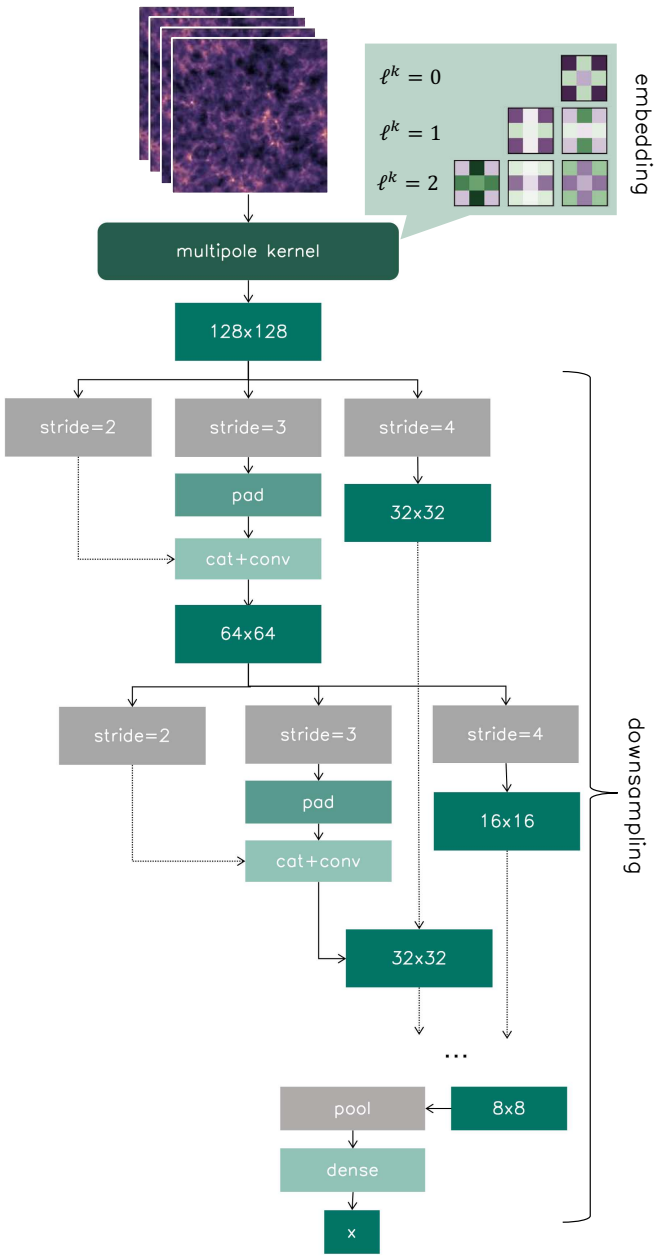


Figure 9.2: We use a small convolutional neural network that exploits the data symmetries to compress κ fields down to additional summaries. Input data (here of shape $(128, 128, 4)$) are passed through a residual multipole kernel layer (shared colour indicates shared weights) and then subsequently passed to convolutional blocks with varying strides with small 2×2 kernels to capture fluctuations on different scales. All linear layers are followed by a nonlinear activation function. Dashed arrows indicate feature concatenation at the same spatial resolution. This downsampling continues until the spatial resolution of the data reaches 8×8, after which the output tensor is mean-pooled along the spatial axes and passed to three dense layers. The output from the network is a pair of numbers.

tend to be distributed in spherically-symmetric and close-to-spherically-symmetric ways. Although the physics laws suggest restriction to circularly-symmetric kernels in 2D, as employed by Charnock et al. (2020) and Ding et al. (2024) for modelling halo bias corrections, Kodi Ramanah et al. (2020) found that networks were able to extract more information by mild breaking of this symmetry, perhaps simply associated with the grid pixelisation of the kernel. This motivates ordering kernel complexity by increasingly breaking from rotational symmetry. Here we expand on Ding et al. (2024)'s implementation and explicitly encode low-order multipole expansion symmetries in CNN kernels. Convolutional kernel weights are shared for kernel pixels equidistant from the centre of a 3D or 2D kernel, associated to the spherical harmonic coefficients $Y_m^{\ell_k}(\theta, \phi)$. Here we make use of these multipole kernels (MPK) in a 2D setting for information capture, embedding the convergence field using a smaller number of neural weights. This choice of embedding is also likely to improve performance in the presence of (white) noise, as noise artefacts are not distributed with the same rotational symmetry as convergence clustering features.

We first embed each log-transformed tomographic bin into six filters corresponding to the $\ell_k = [0, 1, 2]$ multipole moments for a 7×7 kernel per tomographic bin, which for e.g. $N = 128$ corresponds to a 0.19 deg^2 receptive field. We found empirically that including higher ℓ_k did not improve information capture. We illustrate a cartoon example of these symmetric kernels for a 3×3 kernel in Figure 9.2. This output is then passed to a nonlinearity and then to another multipole kernel for each input filter, which are then summed along the filter axis at each multipole kernel to yield six output channels. We learn the residual from the first embedding layer l to the next, e.g. $x^{l+1} = \texttt{mpk\_layer}(x^l + \texttt{mpk\_layer}(x^l))$. This choice of data embedding layer drastically reduces the number of learnable weights and forces the network to learn physically-symmetric functions of the data in its first layer. The largest model considered here contains just $6,904$ trainable parameters, which is 0.08% the footprint of the ResNet18 employed e.g. by Sharma et al. (2024); Lanzieri et al. (2024).

Incept-Stride Tree Network. The embedded data are then passed to an inception-style network (Szegedy et al., 2016) with one important difference: instead of varying kernel sizes, we keep the kernel shape fixed to 2×2 and vary the *stride* that each layer takes in parallel passes over the data. The objective of this section of the network is to downsample the embedded data by combining information from different scales so that the only features on informative scales are strongly activated and pushed

through the learned network to the output summaries using the fewest independent kernel weights possible.

The data is passed to stride-2, stride-3, and stride-4 downsampling layers followed by a nonlinear activation function and a subsequent stride-1 convolution. The outputs of the stride-3 block are padded periodically in the spatial dimension and concatenated to the output of the stride-2 block, and then passed to another stride-1 convolution. The stride-4 outputs are kept aside until the data has been passed to the next inception block and the data has reached the same spatial resolution. We continue downsampling in this tree-like fashion until the data reach a spatial resolution of 8×8. We then mean-pool the features along the spatial axes and pass the resulting flattened filter axis to a final linear layer that outputs the desired additional summaries. Every layer is followed by a new, bijective `smooth_leaky` activation function, which we found empirically extracted information most consistently across our experiments:

$$\texttt{smooth\_leaky}(x) = \begin{cases} x, & x \leq -1 \\ -|x|^3/3, & -1 \leq x < 1 \\ 3x & x > 1. \end{cases} \tag{9.24}$$

The intuition here is that unlike natural image data, lensing shear maps are relatively smooth functions, so are best linked to smoother activation functions following convolutions, in contrast to natural images with sharp features like feature borders, for which typical disjoint activations like the `leaky_ReLU` were developed (Xu et al., 2015).

Training Setup. To train the network we split our dataset into equally-sized validation and training sets, with the same n_s number of fiducial and n_d seed-matched derivative simulations. Every epoch a new noise realisation is added onto the noisefree convergence maps and a random rotation is performed. These transformations are seed-matched for each derivative simulation index. We use the `adam` optimiser with a fixed learning rate of 0.0005 with gradient clipping at a value of 1.0, and a weight decay penalty of 0.0005 added to the loss function. These two modifications to the optimisation routine "smooths" the loss landscape and prevents the network from overfitting to the training data, respectively. Training is halted when the validation loss stops decreasing significantly for a `patience` number of epochs.

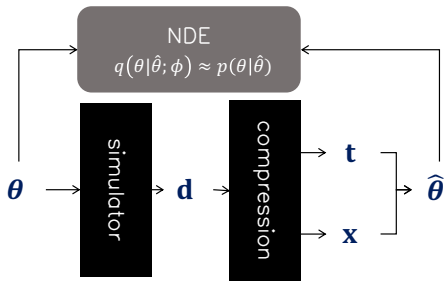


Figure 9.3: Cartoon of density estimation scheme with fixed compression (network or MOPED). Parameters $\boldsymbol{\theta}$ are drawn from a prior and MLE estimates $\hat{\boldsymbol{\theta}}$ are produced from data $\mathbf{d}$ for either (fixed) compression method using Eq. 9.15 and fed to a MAF neural density estimator for the amortised posterior distribution, trained under the loss in Eq. 11.14.

Noise Hardening. All networks are first trained at a low noise level, $A_{\rm noise} = 0.125$, after which the noise level is increased in increments of $\Delta A_{\rm noise} = 0.05$ for a minimum of 100 epochs subject to a patience setting of 75 epochs at each setting. This can be thought of as "domain-transfer" learning on-the-fly. Slowly increasing the noise allows physical features (e.g. convergence patterns) to be embedded early in training, such that the network outputs are already concentrated on the informative distribution of the data when the shape noise increases.

9.5.3 Neural Density Estimation

To measure the information capture in both MOPED and network summaries we employ a neural posterior estimation scheme to parameterise the amortised summary-parameter posterior $p(\boldsymbol{\theta}|\mathbf{y})$, where $\mathbf{y}(\mathbf{d})$ is either the MOPED summary or updated summary set of MLE parameter estimates using Eq. 9.15. We employ an identical ensemble of masked autoregressive flows (MAFs; Papamakarios et al., 2017; Alsing et al., 2019) for each set of summaries using the LtU-ILI codebase (Ho et al., 2024). We opt for networks with 50 hidden units and 12 transformations. We chose this high level of complexity such that the posterior density parameterisations in all cases were sufficiently descriptive. A unique aspect of our network training scheme is that a joint parameter-data prior distribution can be chosen after learning the network and MOPED compression, displayed as a cartoon in Figure 9.3. We generated 5000 simulations for each of two wide uniform priors in the S_8 formalism: $p^{(1)}(\Omega_m, S_8) = \mathcal{U}[[0.15, 0.35] \times [0.7, 1.52]]$ and $p^{(2)}(\Omega_m, S_8) = \mathcal{U}[[0.15, 0.35] \times [0.5, 1.0]]$. We opt for the smaller prior in cases where the 3σ Fisher posterior estimate for the observable considered falls within the support of $p^{(2)}$.

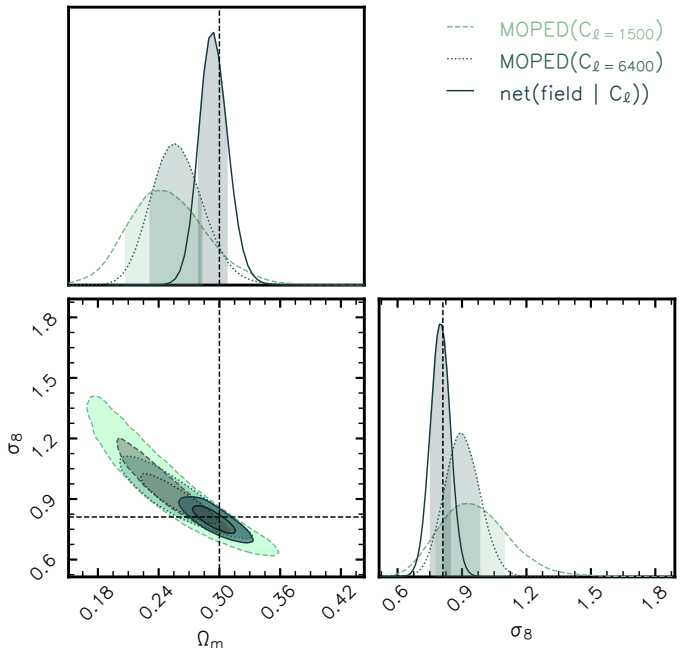


Figure 9.4: Using information-update network summaries (green) drastically improves $\Omega_m - \sigma_8$ constraints beyond MOPED C_ℓ summaries in a low-noise setting. We compare the posteriors obtained from a KiDS-like survey truncation at $\ell_{\rm cut} = 1500$ (blue) to the constraints from all available modes $\ell_{\rm cut} = 6400$ at the given resolution (green). The network's additional summaries (dark green) is able to improve information extraction by a factor of 5 beyond the $\ell_{\rm cut} = 6400$ and a factor of 8 above $\ell_{\rm cut} = 1500$.

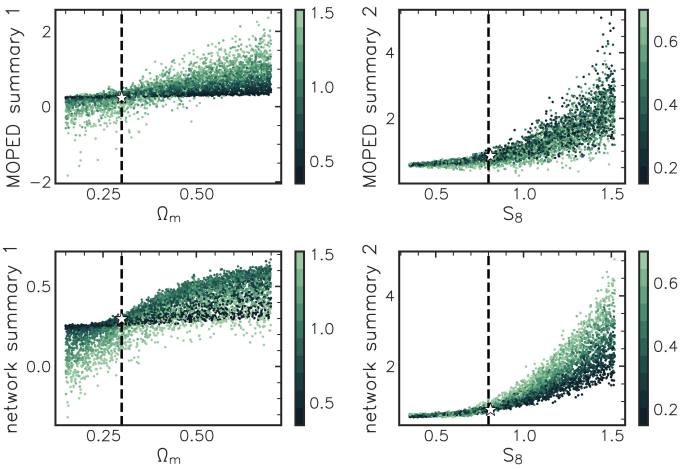


Figure 9.5: Information-update network (bottom) makes simulations more distinguishable in summary space than C_ℓ compression (top). Points in parameter-summary space are coloured by the opposite parameter's value. The network finds patterns that separate these summaries in a complementary fashion even away from the fiducial point $(\Omega_m, S_8) = (0.3, 0.8)$. We display a 3D view of this four-dimensional space in Figure 9.11.

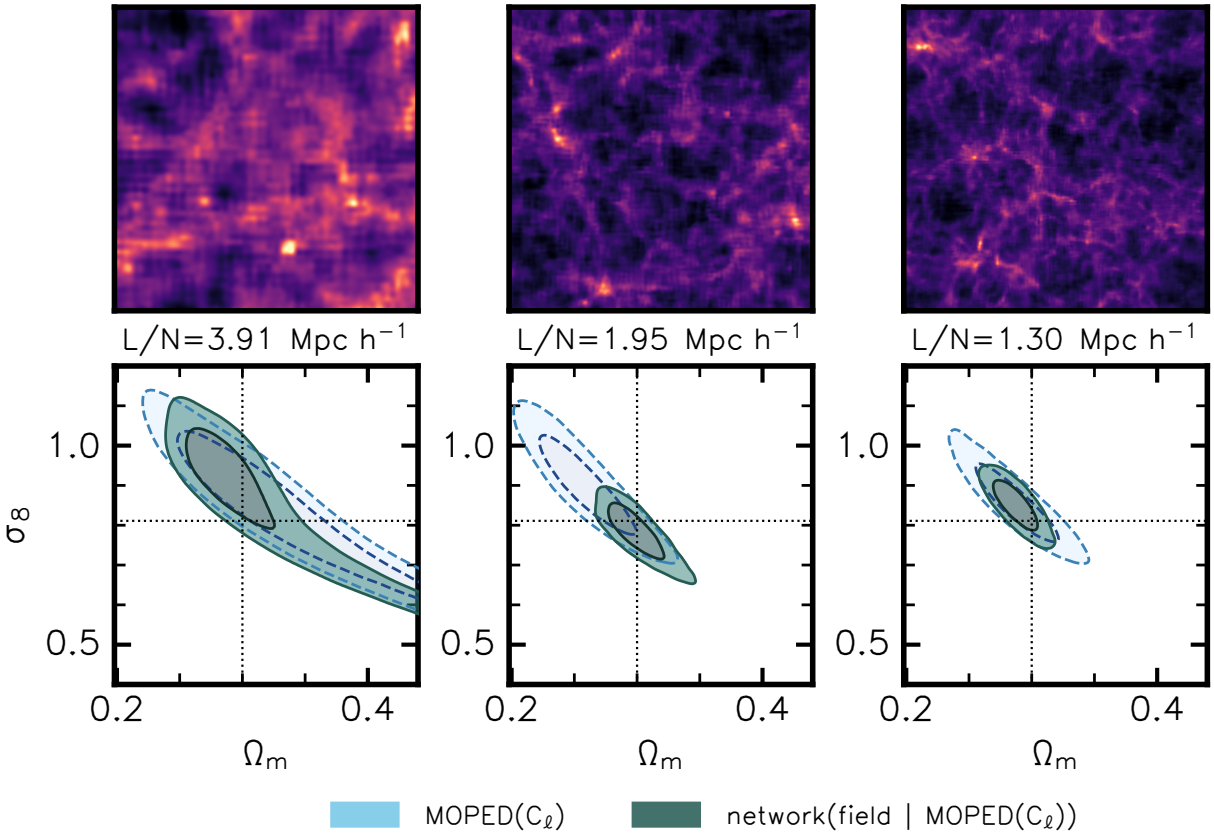


Figure 9.6: Computing additional complementary summaries from the convergence field improves parameter constraints (green) over the two-point information (blue) as the field becomes more nonlinear in the low-noise regime. For $N = 64$ fields the information gain above the C_ℓ constraints is modest, but improves as more nonlinear scales are included at the level of the field as resolution increases.

9.6 Results

9.6.1 Low-noise Regime

We first investigate the information extraction as a function of dark matter simulation resolution with a small amount of shape noise, and compare information extraction to two-point statistics at different scales. We construct particle mesh simulations with varying numbers of pixels $N_x = N_y \in [64, 128, 192]$. Intuitively we expect more information beyond the two-point statistic to be found at higher resolutions, which can be interpreted as the descriptiveness of the underlying gravity model.

Scale Cutoff. We first explored the effect of a scale cutoff at resolution $N = 128$ for the C_ℓ summaries with a low noise setting. We construct MOPED summaries from C_ℓs subject to a Stage-III survey-like cut at $\ell_{\rm cut} = 1500$, as well as summaries from all available ℓ modes at the given resolution. The highly-compressed MOPED summaries give almost identical posteriors to using the full set of C_ℓ values as the data vector. The network is tasked with finding complementary summaries in the $\ell_{\rm cut} = \ell_{\rm max}$ case. We display the constraints obtained on the same target simulation in Figure 9.4 and in Table 9.1. The network extracts up to 5 times more information than the two-point function in a low-noise setting with all modes and 8.3 times more information in high-noise settings, as measured by the determinant of the Fisher matrix.

	resolution	$H(C_\ell)$	H(net)	ratio
low noise	$N = 64$	6.9	7.4	**2.9**
	$N = 128$	6.9	7.7	**5.0**
	$N = 192$	7.6	8.5	**5.1**
high noise	$N = 128$	5.3	6.0	**4.5**
	$N = 192$	5.2	6.3	**8.3**

Table 9.1: Summary of parameter Shannon information ($\mathrm{H} = \frac{1}{2}\ln\det F$) from MOPED and information-update networks for low noise ($n_{\rm gal} = 1900$) and high noise ($n_{\rm gal} = 83$) scenarios. The ratios of Fisher determinants are shown in the last column. For the noisy $N = 192$ case we optimise networks against the binned C_ℓ vectors as opposed to the MOPED summaries.

Summary Scatter. The information-update loss scheme asks the network to use data features such that output summaries complement existing C_ℓ-based summaries. We can interpret the statistics learned by the network by visualising a summary scatter over the suite of prior simulations. Figure 9.5 shows the network and MOPED outputs versus true parameter, coloured by the opposite parameter's value for summaries used to generate the network and $\ell_{\rm max} = 6400$ posteriors in Figure 9.4. Remarkably, even though the network is only trained at the fiducial cosmology (dashed vertical line in each plane), the information-update loss allows the network to find useful features with which to distinguish parameters in a smoother summary space (less scatter) than the MOPED compression. This increased structure is then harnessed by the density estimation scheme to provide tighter parameter constraints than MOPED. The complementary nature of the information from the network-updated statistics to the original statistics decreases away from the fiducial point in both dimensions, but does so smoothly, i.e. the information about the parameters coming from the four statistics is mixed away from the fiducial point.

Resolution Dependence. We next compare constraints as a function of PM simulation resolution, which effectively controls the nonlinearity of the dark matter gravity solver. Here we wish to measure the parameter information gain that using a nonlinear network to probe nonlinear scales adds to the power spectrum. We generate three suites of fiducial and finite-difference convergence maps to learn the compression with $A_{\rm noise} = 0.125$. Figure 9.6 shows constraints at each resolution. More non-Gaussian information is extracted for the higher resolution simulations, indicated by the increase in network constraining power over the C_ℓs, since these simulations probe smaller, more nonlinear scales accessed by the network. We report our network and MOPED Fisher constraints in Table 9.1. In this low-noise setting we observe an information increase of a factor of 2.9 for $N = 64$ and a factors of 4-5 for $N = [128, 192]$ high-resolution simulations, aligning with our intuition.

9.6.2 High-noise Regime

The information-update formalism displays promising results in the presence of increased systematics such as galaxy shape noise. Here we start with a network trained on the lowest noise setting and slowly increase the noise amplitude (equivalently decreasing the galaxy density). The network is able to increase its relative performance against the two-point statistic as we increase resolution (Fig. 9.7) and shape noise (Fig. 9.8). Figure 9.7 shows that with increased simulation resolution the network

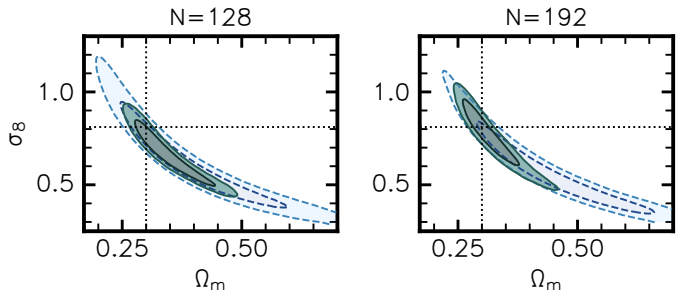


Figure 9.7: Additional neural summaries (green) are robust to shape noise ($n_{\rm gal} = 120$) at different resolutions. Constraints from C_ℓ-only summaries (blue) suffer from the increased noise due to shot-noise contributions to the high-ℓ bins.

has access to more nonlinear scales and can compensate for the shot noise that dominates the C_ℓ calculation at high ℓ values. For a fixed resolution ($N = 192$), the additional summaries found by the network appear robust to noise; keeping the parameter constraints consistent as increased shape noise pushes the C_ℓ constraints towards the prior edge in Ω_m. This is especially promising for higher noise cases for galaxy density $n_{\rm gal} < 50$ arcmin^{-2}, as this falls between the capabilities of Euclid (Euclid Collaboration, 2022) and Roman (Spergel et al., 2015) telescopes.

Optimising With the Binned Power Spectrum. For our high-resolution simulations we also explored optimising the information-update formalism (Eq. 9.11) with respect to the full binned C_ℓ vector. This "stretches" the off-diagonals $\mathbf{u}$ of the full summary covariance (Eq. 9.5), such that the IUF forces the network to respond to explicit fluctuations in these C_ℓ bins with respect to noise. Here we posit that the network will be able to find summaries that complement fluctuations at different ℓ scales more efficiently. We find that indeed this choice of optimisation allows the network to extract 8.3 times more information than the two point function in the noisiest ($n_{\rm gal} = 30$) setting, compared to a 5.6 times improvement when optimising against the MOPED-compressed summaries. We visualise the joint covariance of learned and binned C_ℓ summaries (Eq. 9.5) in Appendix 9.11.

9.7 Discussion & Conclusions

In this paper, we present an implicit inference technique to extract neural summary statistics from field-level data, specifically weak lensing maps, that are designed to match or increase automatically the Fisher information about the cosmological parameters over a set of pre-defined summaries, typically traditional two-point statistics. We apply this method to find summary statistics from

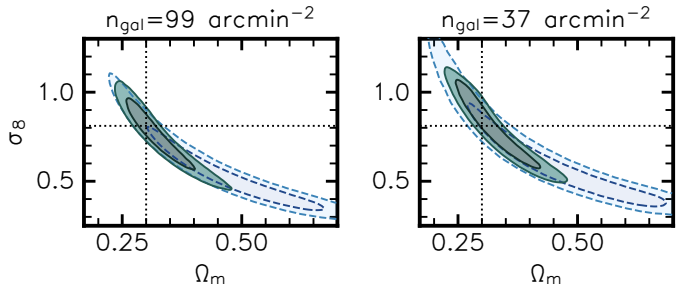


Figure 9.8: Neural summaries (green) are robust to increased shape noise, controlled by the galaxy density parameter $n_{\rm gal}$. Increased shot noise at small scales inhibits angular C_ℓ constraints (blue). Here we display an inference on the same $N = 192$ resolution simulation subject to increased shape noise. We optimise the network using the binned C_ℓ vectors.

tomographic convergence maps that explicitly complement the angular power spectrum estimates. This powerful hybrid mixture of physics-based and neural network derived summary statistics is guaranteed to improve the two-point parameter constraints and allows for networks with small physics-informed architecture to achieve similar results to larger regression networks. We demonstrated that this approach extracts between a factor 3 and 8 more information than the angular power spectrum, as measured by the determinant of the Fisher matrix. For weak lensing, the main gain is a substantial reduction in the credible region for Ω_m, with a smaller improvement in the S_8 error.

Other studies have previously combined power spectrum and network summaries in weak lensing analyses. Jeffrey et al. (2024) for example feed in both sets of independently-obtained summaries into an NDE for posterior estimation to obtain a $\sim 2\times$ improvement in information extraction in $\Omega_m - S_8$. Here we show that coordinating the field-level network optimisation with an existing summary can give us even more efficient extraction.

The hybrid summary formalism presented in this work is not limited to weak lensing data, and it can be generalised to any dataset to identify the features from which the information captured by large neural networks comes. This technique might also reduce the need for large convolutional networks to learn the large-scale correlations in larger dark matter and galaxy simulations (Lemos et al., 2023b). In future work, we will apply this formalism to find the summary that complements the information from more than one pre-determined summary statistic, such as angular power spectrum and peak count summaries. This has the potential to improve the cosmology constraints from implicit likelihood analyses of weak lensing such as Fluri et al. (2022) and Jeffrey et al. (2024).

It is worth making a final comment on the black-box nature of the neural network. Although the

additional summary statistics are obtained by the network based on information theoretic considerations, the process might still be regarded as a black box. For the purposes of Bayesian inference, however, it does not matter how the network found the summaries; it only matters that they are informative. The (a priori unknown) sampling distribution of the summaries is not used in simulation-based inference, which can be applied in a fully Bayesian way. However, our new multipole network architecture does allow the user to probe this information capture as a function of kernel complexity. Here we found empirically that e.g. truncating kernels to $\ell_k = [0, 1]$ captured less information than the $\ell_k = [0, 1, 2]$ kernels used in the presented analysis. We leave a thorough comparison to a future work.

9.8 Code Availability

The code for this analysis will be made available at `https://github.com/tlmakinen/hybridStatsWL`. All custom networks and simulation tools were written in `Jax` (Bradbury et al., 2018) and `flax` (Heek et al., 2020) and were run on a single NVIDIA v100 32Gb GPU. Posterior density estimation was performed locally on a laptop CPU using the LtU-ILI code (Ho et al., 2024).

9.9 Acknowledgements

TLM acknowledges the Imperial College London President's Scholarship fund for support of this study, and thanks Niall Jeffrey, Justin Alsing, David Spergel, and Maximilian von Wietersheim-Kramsta for insightful discussions. NP is supported by the Beecroft Trust. BDW. acknowledges support by the ANR BIG4 project, grant ANR-16-CE23-0002 of the French Agence Nationale de la Recherche; and the Labex ILP (reference ANR-10-LABX-63) part of the Idex SUPER, and received financial state aid managed by the Agence Nationale de la Recherche, as part of the programme Investissements d'avenir under the reference ANR-11-IDEX-0004-02. TLM acknowledges helpful conversations facilitated by the Learning the Universe Collaboration. This work was conducted within the Aquila Consortium. The Flatiron Institute is supported by the Simons Foundation.

Appendix

9.10 Posterior Coverage Tests

One of the distinct advantages of SBI neural density estimation is the immediate availability of coverage tests. In this work we trained an estimator for the posterior distribution given some point-estimates for the parameters via MOPED or the hybrid-summary network: $p(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}})$. This density estimator is an *amortised* posterior, meaning the posterior density for any given summaries $\hat{\boldsymbol{\theta}}$ is immediately available without having to do MCMC sampling with a likelihood. We can then do repeated mock data parameter inference over the prior using this posterior density, and calculate how many true parameter vlaues from the credible intervals match the expected fraction, forming a posterior "coverage" test. We display one such test in Figure 9.9 making use of the `TARP` coverage test framework presented in Lemos et al. (2023a).

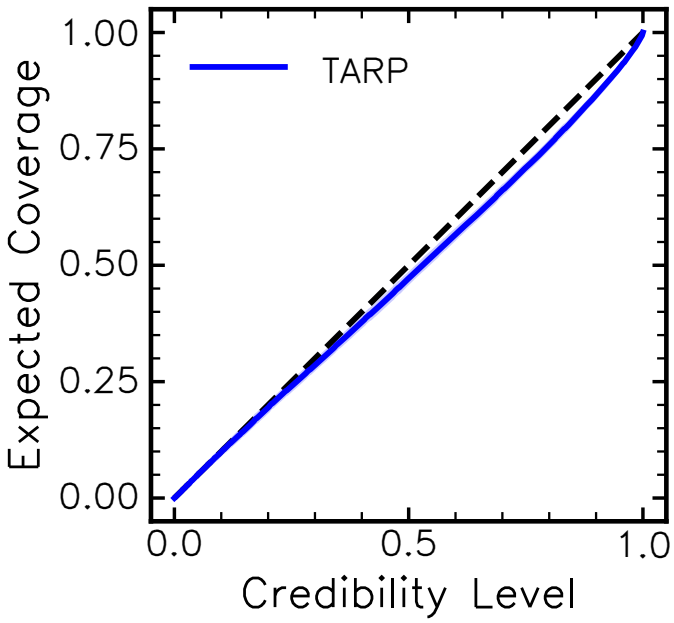


Figure 9.9: Example coverage test result for low-noise inference with $N = 192$ resolution (rightmost panel in Fig. 9.6) using `TARP` (Lemos et al., 2023a). Using our amortised parameter-summary posterior $p(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$, we can do repeated mock data parameter inference over the prior, and measure which fraction of true values from the appropriate credible intervals matches the expected fraction. The blue line traces 100 "distances to random points" (DRP), which is accelerated using the `TARP` framework within LtU-ILI (Ho et al., 2024). The DRP line (blue) traces the truth line (dashed), indicating a successful test.

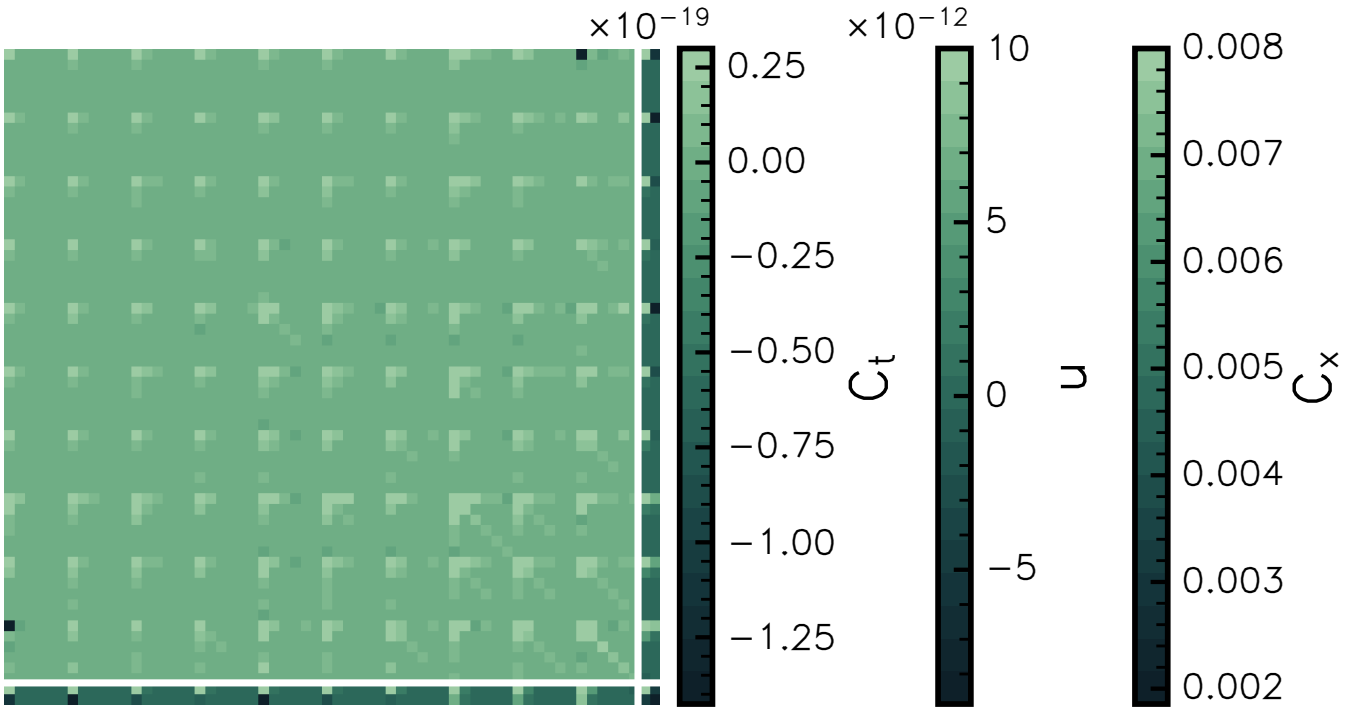


Figure 9.10: Example joint C_ℓ-network summary covariance visualisation (Eq. 9.25) for a network optimised against the binned power spectrum. We separate the 60×60 C_ℓ covariance structure (upper left corner) from network summaries (lower right corner) with the set of intersecting white lines. The learned network summaries exhibit a non-zero correlation structure with the ℓ bins, illustrated by the $\mathbf{u}$ off-diagonal matrix vectors on the bottom and right-hand edges. Here it is obvious that only one of the two network summaries is highly correlated with the binned power spectrum modes.

9.11 Learned Covariance Matrix Visualisation

In Figure 9.10 we illustrate the full joint covariance,

$$\mathbf{C} = \begin{pmatrix} \mathbf{C}_t & \mathbf{u} \\ \mathbf{u}^T & \mathbf{C}_x \end{pmatrix}, \tag{9.25}$$

of the binned tomographic C_ℓ statistic and two learned network summaries, clearly separated by the white intersecting lines. We plot each component of this structure with a separate colourbar. The cross-correlation row-matrices $\mathbf{u}$ indicate that the learned summaries exhibit non-trivial correlation structure with the binned ℓ-modes, which contributes to information capture according to the hybrid statistics formalism.

9.12 Summary Scatter

Here we display a three-dimensional view of the four-dimensional joint distribution of compressed summaries and parameters $p(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$ obtained from MOPED and network compressions in a low-noise setting. The information update formalism tells the convolutional network during optimisation to make use of the nonlinear information on the smaller (pixel-level) scales that it has access to in a way that is complementary to the power spectrum. Although optimised at a fiducial point, the mapping learned is smooth as a function of data $\mathbf{d}(\boldsymbol{\theta})$ away from the training point, resulting in more structure in the four-dimensional joint distribution space $p(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$ than MOPED, allowing the summaries (z-axis and colour) to respond more smoothly and rapidly as a function of parameters $(x, y) = (\Omega_m, S_8)$. This smoother joint distribution surface can then be harnessed by the NDE scheme to produce tighter posteriors in an amortised fashion.

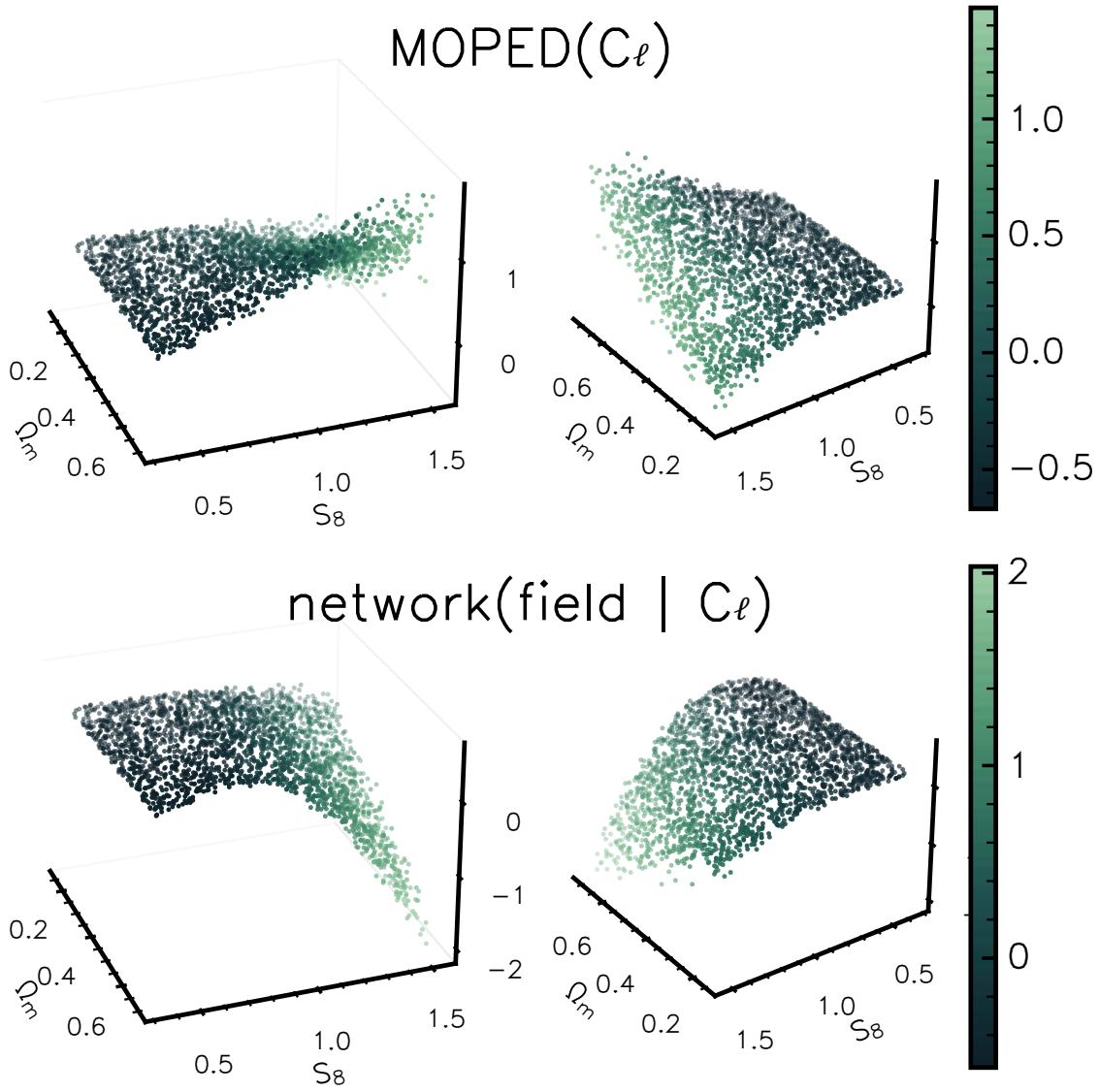


Figure 9.11: Computing additional information from the data endows the network summaries (bottom row) with more structure as a function of parameters (Ω_m, S_8) over a prior than MOPED two-point summaries (top row). Summaries for each method $\hat{\Omega}_m$ and $\hat{S}_8$ are indicated by z-direction and colourbar, respectively. It is highly visible via the increased structure in joint space that the network is able to capture information from the smaller scales it has access to. More scatter in z or colour at a particular parameter value in the MOPED summaries indicates a less informative compression of the simulated convergence data to from power spectrum summaries.

CHAPTER 10

HYBRID SUMMARY STATISTICS (PART II)

Hybrid Summary Statistics

T. Lucas Makinen[1*], Ce Sui[2*], Benjamin D. Wandelt[3,4], Natalia Porqueres[5], Alan Heavens[1]

[1]Imperial Centre for Inference and Cosmology (ICIC) & Astrophysics Group,
[2]Tsinghua University
[3]Sorbonne Université, CNRS, UMR 7095, Institut d'Astrophysique de Paris, 98 bis boulevard Arago, 75014 Paris, France
[4]Center for Computational Astrophysics, Flatiron Institute, 162 5th Avenue, New York, NY 10010, USA
[5]Department of Physics, University of Oxford, Denys Wilkinson Building, Keble Road, Oxford OX1 3RH, United Kingdom

Accepted to NeurIPS 2024 Machine Learning for the Physical Sciences; (Makinen et al., 2024)

Abstract

We present a way to capture high-information posteriors from training sets that are sparsely sampled over the parameter space for robust simulation-based inference. In physical inference problems, we can often apply domain knowledge to define traditional summary statistics to capture some of the information in a dataset. We show that augmenting these statistics with neural network outputs to maximise the mutual information improves information extraction compared to neural summaries alone or their concatenation to existing summaries and makes inference robust in settings with low training data. We introduce 1) two loss formalisms to achieve this and 2) apply the technique to two different cosmological datasets to extract non-Gaussian parameter information.

10.1 Introduction

Implicit (simulation-based) inference makes solving otherwise intractable inverse problems possible by employing neural compressors which can flexibly map big data vectors into informative summaries

[*]Equal contribution

(Cranmer et al., 2020a; Hoffmann & Onnela, 2023). These mappings can be made optimal (Charnock et al., 2018; Makinen et al., 2021; **?**; Lanzieri et al., 2024; Jeffrey et al., 2020), but often require large numbers of simulations to achieve convergence.

For many physics problems, such as large N-body and hydrodynamical solvers, forward simulations are expensive to generate, so networks tasked to compress these data for parameter inference must learn informative features using sparsely-sampled training sets, especially when large numbers of parameters are varied. Introducing informative priors in data feature (e.g. Battaglia et al., 2018; Ivanov et al., 2024) or likelihood (Modi & Philcox, 2023) space can simplify the task of a neural network to learn an objective.

Main Contributions. We present a way to obtain highly informative summaries over parameter space in low-training data settings that *boost* information extraction from data beyond existing statistics and the mere concatenation of neural summaries learned separately and traditional summaries. These "hybrid" statistics are neural summaries that are learned to maximise the mutual information (MI) beyond an existing summary (fixed function) of the data and the parameters of interest.

Summary of Results. We present two equivalent objectives that maximise MI information between a new, neural summary, and an existing summarisation of the data over parameter space. We apply this technique to two different cosmological problems where information can be lost to existing summary functions–21cm Epoch of Reionisation (21cm) and weak gravitational lensing (WL) parameter inference. We show that hybridised summaries are far more robust in settings with low available training simulations, indicating improved network optimisation over parameter space.

10.2 Formalism

10.2.1 The Mutual Information.

Consider the data compression problem illustrated in Fig. 10.1. We begin with the definition of the mutual information for continuous random variables X, Y:

$$I(x;y) = \int\int p(x,y) \log \frac{p(x,y)}{p(x)p(y)} dxdy \tag{10.1}$$

We can then ask about the mutual information of a third random variable, z, given some existing information y. This results in the conditional mutual information:

$$I(X;Z|Y) = \int\int\int p(x,y,z)\log\frac{p(x,y,z)p(y)}{p(x,y)p(z,y)}dxdydz \tag{10.2}$$

We are free to factor this expression into a more convenient form:

$$= \int\int\int p(x,y,z)\log\left(\frac{p(y)}{p(x,y)}p(x|y,z)\right)dxdydz \tag{10.3}$$

$$= \int\int\int p(x,y,z)\log\left(\frac{p(x|y,z)}{p(x|y)}\right)dxdydz \tag{10.4}$$

$$\tag{10.5}$$

which we can then write concisely in terms of the differential entropy $h(a) = \mathbb{E}_{p(a)}\left[-\log p(a)\right] = -\int p(a)\log p(a)da$ as

$$I(x;z|y) = h(x|y) - h(x|y,z). \tag{10.6}$$

We can then optimise criterion for the summary z. We have samples of all quantities from the

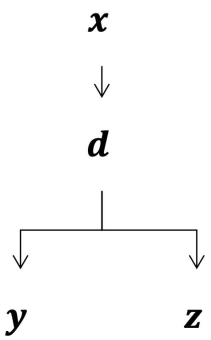


Figure 10.1: Diagram illustrating the joint distribution $p(x,y,z)$ via intermediate quantity d.

joint distribution $p(x,y,z)$ (Fig. 10.1), where x first passes through a function $m : x \mapsto d$ to an intermediate quantity d, before being passed through (potentially lossy) compressions to obtain y and z. The first term of Eq. 10.6 is a constant with respect to z. This means that if we wanted to find another function of $g_2(g(d))$ that preserves the entropy, e.g. $h(x|y) = h(x|g_2(y))$, we are free to do so. Proceeding to the second term, we can maximise I by minimising the expected posterior

entropy via a surrogate conditional density estimator q over the joint distribution:

$$\min_z \left[-h(x|[y,z])\right] \approx \min_z \left[-\frac{1}{n_b}\sum_i^{n_b} \log q(x_i|[y_i,z_i])\right]. \tag{10.7}$$

Expected Posterior Entropy Loss

This optimisation minimises the expected posterior entropy (EPE) (Hoffmann & Onnela, 2023; Barber & Agakov, 2004; Poole et al., 2019), and is identical to the variational mutual information maximisation (VMIM) (Jeffrey et al., 2020), although now with an existing function of the data y specified as a conditional input. Rewritten for our cosmological parameter objective (Fig 11.2), Eq 10.7 becomes (henceforth the EPE loss):

$$\min_{F,q} \mathcal{L} = -\mathbb{E}_{p(\theta,\mathbf{x})}\Big[\log q\left(\theta\big|[F(\mathbf{x}), t(\mathbf{x})]\right)\Big], \tag{10.8}$$

where $\mathbf{s} = F(x)$ is a neural network acting on the full data, $t = t(x)$ is an existing (fixed) function of the data, and q is a neural density estimator.

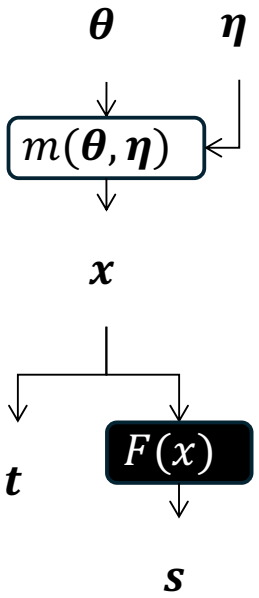


Figure 10.2: Plate diagram illustrating the joint distribution $p(\theta, \mathbf{x}, \mathbf{t}, \mathbf{s})$ via intermediate data quantity $\mathbf{x}$ and nuisance parameters η. We wish to optimise the function $\mathbf{s} = F(\mathbf{x})$ given that we know $\mathbf{t}(\mathbf{x})$.

Cross-Entropy Loss The mutual information objective can also be parameterised as a cross-entropy

classification problem by employing the Jensen-Shannon divergence (Chen et al., 2021a; Devon Hjelm et al., 2018) and variational representation from Nowozin et al. (2016):

$$\min_{F,c} \mathcal{L}(F,c) = \mathbb{E}_{p(\theta,\mathbf{x})}\left[\text{sp}(-c(\theta,[F(\mathbf{x}),y]))\right] + \mathbb{E}_{p(\theta)p(\mathbf{x})}\left[\text{sp}(c(\theta,[F(\mathbf{x}),y]))\right], \tag{10.9}$$

where s is the summarizer, c is a classifier tasked with distinguishing between data from $p(\theta, x)$ and $p(\theta)p(x)$ and $\text{sp}(z) = \log(1+e^z)$ is the softplus function. We refer to this as the Cross Entropy (CE) loss.

To test the information capture in learned summaries we employ a two-step process, illustrated in Fig. 10.3. We first optimise a neural compression (embedding network) to a few additional numbers conditional on the specified (not learned) existing summary $\mathbf{t}$, using either EPE or CE losses (denoted MI maximiser in Fig. 10.3). For the EPE loss we train a simple mixture density network (MDN) with two small hidden layers to approximate $q(\theta|[\mathbf{s},\mathbf{t}])$. For the CE loss we train a classifier fully-connected network with hidden sizes $[128, 64, 64]$ to one output. We then take the static learned and existing summaries and parameterise a separate posterior estimator using a masked autoregressive flow (MAF) to minimise $p(\theta|[\mathbf{s},\mathbf{t}])$ from the `LtU-ILI` package (Ho et al., 2024). We feed comparison summaries (power spectrum and competing network schemes) into the same MAF architecture to obtain a consistent comparison of information capture.

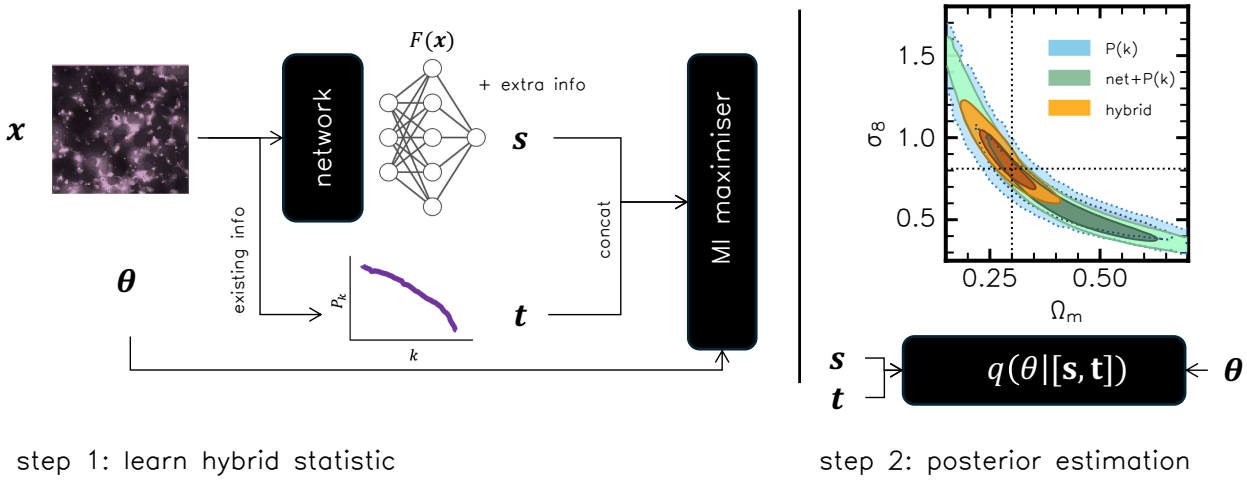


Figure 10.3: Schematic for learning a hybrid summary statistic $\mathbf{s}$ with a choice of mutual information maximiser and existing static $\mathbf{t}$. Black boxes denote neural functions to be learned.

10.3 Experiments

Here we validate that the two different loss formalisms improve information capture beyond the two-point function from two different types of cosmological simulations. In the weak lensing case we also demonstrate the improved efficiency of network learning in the hybrid scheme by restricting the simulation budget available to the network.

10.3.1 21cm Parameter Inference & Loss Comparison.

The 21 cm signal is non-Gaussian due to reionization patchiness. Therefore, the power spectrum alone cannot fully capture the information contained in images of the 21 cm signal. While many previous studies have focused on designing new summaries, we show that adding just a few supplementary features to the power spectrum can significantly enhance the extracted information content. The reionization parameters we vary are

(1) ζ, the ionizing efficiency. It primarily determines the timing of the EoR, with higher values leading to earlier ionization of the Universe. We vary ζ as $10 \leq \zeta \leq 250$, and

(2) $T_{\rm vir}$, the minimum virial temperature of halos that host ionizing sources. $T_{\rm vir}$ controls the timing of astrophysical epochs and influences the scales of heated and ionized regions. We vary this parameter as $4 \leq \log_{10}(T_{\rm vir}/{\rm K}) \leq 6$.

The 21 cm signals are simulated using the publicly available code `21cmFAST` (Mesinger & Furlanetto, 2007; Mesinger et al., 2011). The simulations were performed on a cubic box of 128 comoving Mpc on each side, with 64^3 grid cells. For this work, we use coeval boxes at redshift 12, and extract a single slice from each cube to form a 2D dataset.

In this initial experiment, we compare tree types of summaries, assuming a sufficiently large training set (10,000 samples):

1) Power spectrum only (11 k-bins).

2) Hybrid method: Power spectrum + two learned supplementary features (EPE Loss).

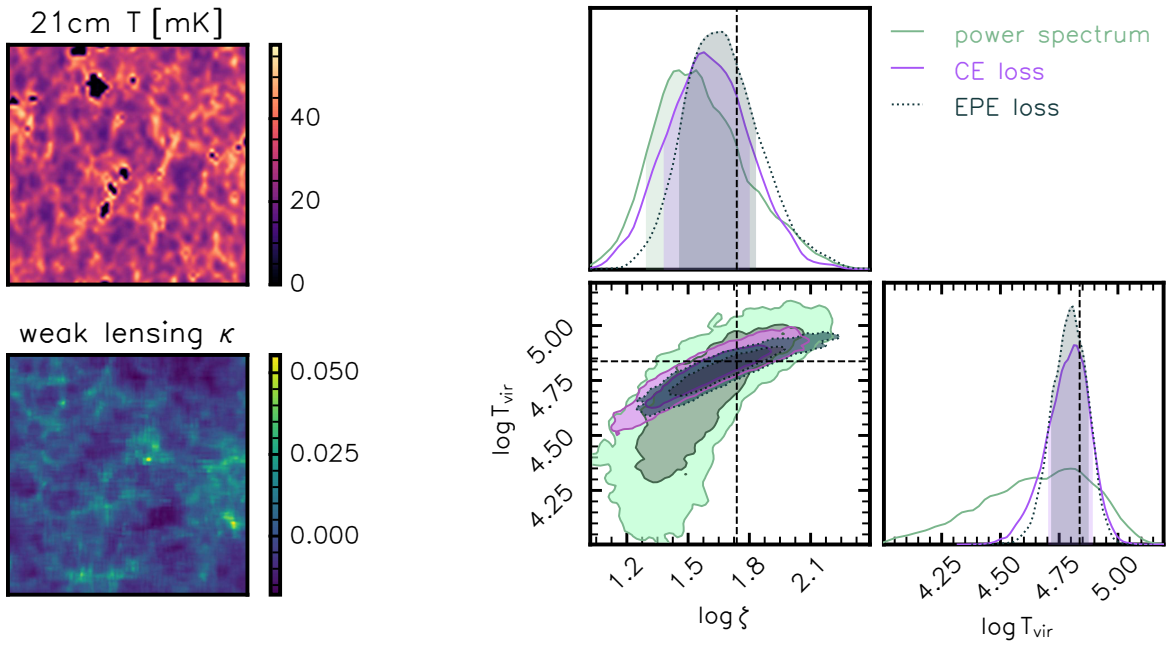


Figure 10.4: Both 21cm and weak lensing data exhibit non-Gaussian features (upper and lower left panels). Both EPE and CE loss formalisms result in consistent, tighter posteriors than power spectrum alone indicating information extraction from non-Gaussian features in the 21cm data (right). The black dashed line represents the true parameter values.

3) Hybrid method: Power spectrum + two learned supplementary features (CE Loss).

Network details. To obtain hybrid summaries, we use a CNN with three convolutional layers (32, 64, and 128 filters) followed by max pooling. The flattened output is processed through two fully connected layers, with ReLU activations throughout. For classification, we use an FCN with hidden layers of sizes 128, 64, and 64, each followed by ReLU activation. For the EPE loss, a mixture density network (MDN) with a 64-unit hidden layer and output layers for mixing coefficients, standard deviations, and means is employed. Mixing coefficients use softmax, standard deviations are exponentiated, and means are directly outputted, with five components used.

One of the resulting inferences is shown in Figure 10.4. The results show that both approaches capture non-Gaussian information and improve inference performance, yielding consistent posteriors. The slight difference in posterior recovery is likely due to differences between classifier and MDN network architectures for each loss, but each yielded similar convergence times in training. This suggests we can learn two additional parameters instead of new summaries, maintaining power spectrum interpretability while achieving near-optimal inference. The agreement also confirms both loss functions are effective, converging to the same results.

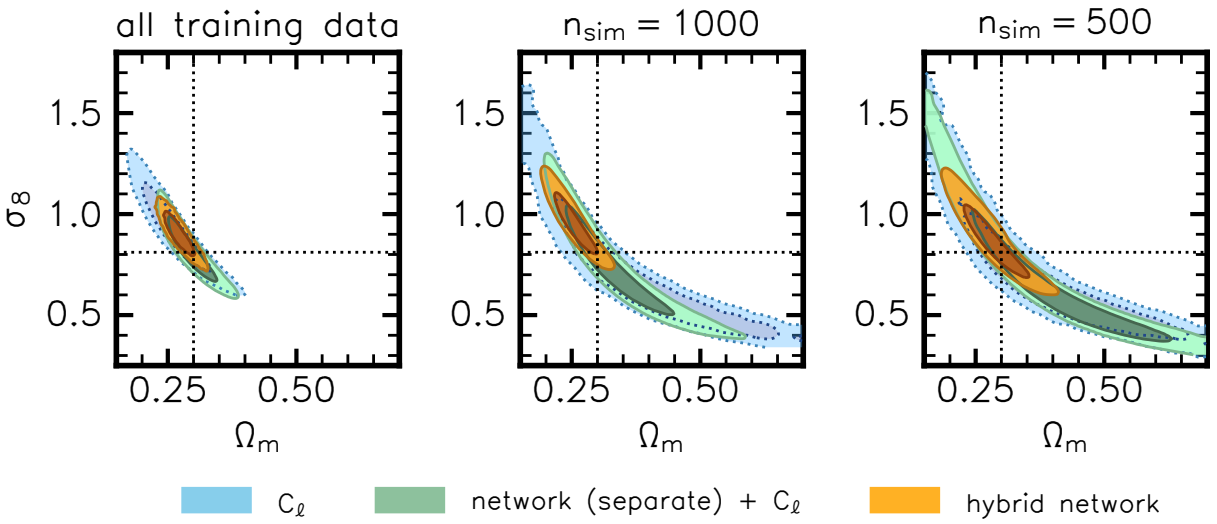


Figure 10.5: Hybrid summaries are able to capture more non-Gaussian parameter information in settings with both high (left) and low (right) numbers of training simulations than summaries from a larger network trained with the same loss separately from and then concatenated to the C_ℓ statistic for the final inference in step 2. When reducing the total available simulation volume all inferences suffer, but the hybrid statistics are most robust to this change, indicating that the MI objective improves capture of non-power spectrum features.

10.3.2 Tomographic Weak Lensing Inference & Ablation Study.

Weak gravitational lensing (WL) alters the trajectories of photons as they pass through massive structures of visible and dark matter. This observable is sensitive to (Ω_m, S_8), parameters that control the universe's matter content and dark matter clustering, respectively. Here we test hybrid summaries on noisy tomographic WL convergence image data of shape $(128, 128, 4)$ presented in **?** varied over a wide uniform prior. As an existing summary, we repeat **?**'s procedure and histogram all auto- and cross-power spectra for each redshift bin into a vector of 60 numbers for each simulation. The simulations are also subject to additive shape noise, which we add to the noise-free simulations on-the-fly during network training.

Network details. For obtaining hybrid summaries the convolutional neural network with symmetric Multipole Kernels (MPK) was adapted from **?**. The lightweight network is initialised without pretraining for this analysis and contains $1,615$ learnable parameters. For the large non-hybrid CNN we apply a 3×3 kernel to embed the field into 16 filters, and then down-sample with stride-2 convolutions with output filters $[32, 64, 128]$. The network is then mean-pooled in the spatial axes and the flattened filters are passed to a dense network with a specified output size to be fed into the mutual information maximiser (here a mixture density network for the EPE loss). This results in

97,971 learnable parameters. For both embedding networks we employ the `smooth_leaky` activation function from **?**. For the EPE loss configuration, a mixture density network (MDN) with hidden layers of size [70, 70] and output layers for mixing coefficients, standard deviations, and means is employed to parametrise a four-component mixture.

Ablation Study

We perform an ablation study to demonstrate the effectiveness of the hybrid statistics over neural-only methods and display results in Fig. 10.5. The embedding network in the hybrid setting is the lightweight CNN with symmetric kernels adapted from **?** to output 3 additional numbers alongside the C_ℓs. For comparison, we train a larger, more expressive CNN embedding network under the same EPE loss *without* access to the C_ℓ vector to output 3 numbers, and then for the density estimation (step 2) concatenate its outputs to the C_ℓs (green contours; network (separate) + C_ℓ). Step 2 probes the information content captured in the static network and C_ℓ summaries. We train the networks and density estimators from scratch first using all 5000 simulations available (split into 70% train and 30% validation sets). We then reduce the total number of available simulations to 1000 and further to 500 and re-learn the embeddings and posteriors from scratch. When the simulation budget is reduced, all inferences suffer, but the hybrid statistic formalism encourages the smaller network to find non-Gaussian features in the dataset that are more robust to this change. We also note that the more expressive, separately-trained CNN inference degrades almost to the level of the C_ℓ contour in the lowest-data setting, even when concatenated to the C_ℓ vector in the density estimation step.

10.4 Information Content of New Summaries

Hybrid summaries make learning more efficient, but are also useful for interpreting *how* a network has captured complementary information. We display posterior constraints from hybrid summaries alone (no C_ℓ) for the weak lensing problem in Fig. 10.6. Plotting the posterior in the $S_8 = \sigma_8\sqrt{\frac{\Omega_m}{0.3}}$ parameter space shows that the posterior from summaries $\mathbf{s}$ (pink contour) is, by itself, not very constraining, but intersects the existing C_ℓ constraints in a complementary fashion at the centre of both posteriors, resulting in the (separately-estimated) orange combined posterior. Complementary constraints like these echo those obtained by incorporating higher-order statistics like the bispectrum

or peak counts in a joint analysis (like those explored using graphs in Chapter 7), except here the network has isolated these complimentary features in an automatic way from the data.

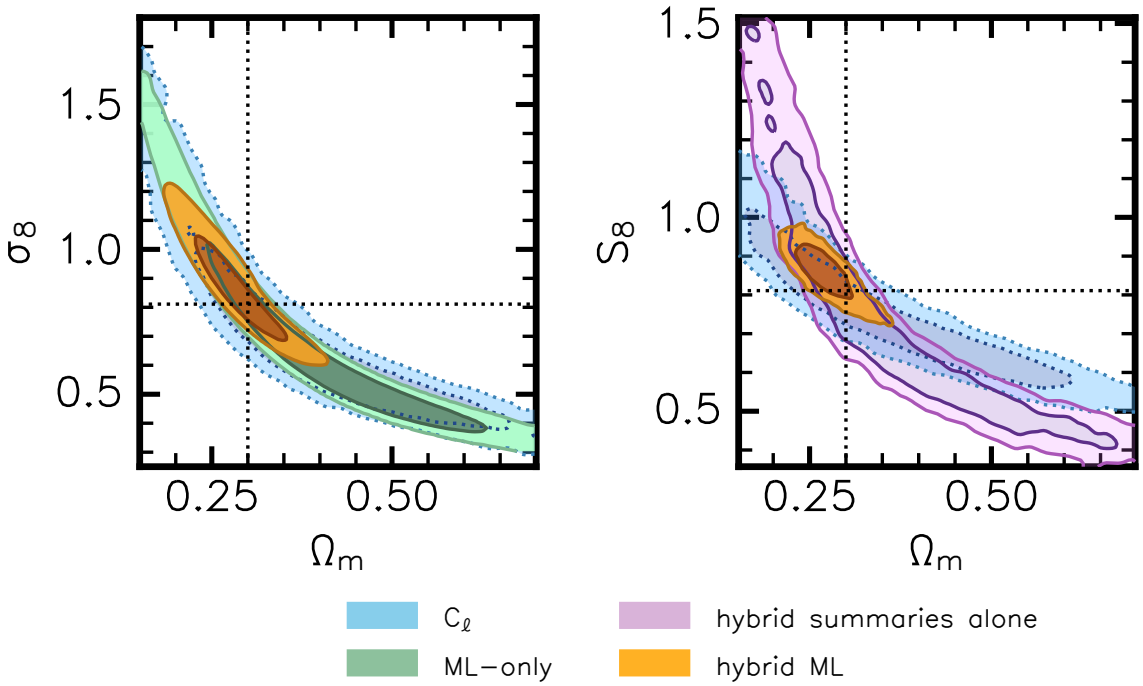


Figure 10.6: *Left:* Hybrid summaries (orange) are able to capture more non-Gaussian parameter information than larger networks (green) trained with existing methods with a limited number of simulations. *Right:* The new network summaries (pink) are learned to automatically complement two-point constraints (blue) to provide tight constraints together (orange).

10.5 Conclusions & Outlook

We detailed a method to learn compressed summary statistics that explicitly complement an existing summary of the data through mutual information maximisation. We show that these techniques can capture non-Gaussian information in two cosmological applications using two different loss criteria to significantly improve parameter information capture.

We additionally demonstrate that these hybridised summaries improve information capture when the training simulation budget is limited. This suggests that requiring a network to find patterns in the data that are explicitly complementary to a provided summary "tells it where to look" and improves the compression optimisation in smaller datasets over wide parameter space.

We note that MI can also be used as a static metric to quantify the information content in arbitrary summaries as in Sui et al. (2023). This technique could be extended towards exhaustive information

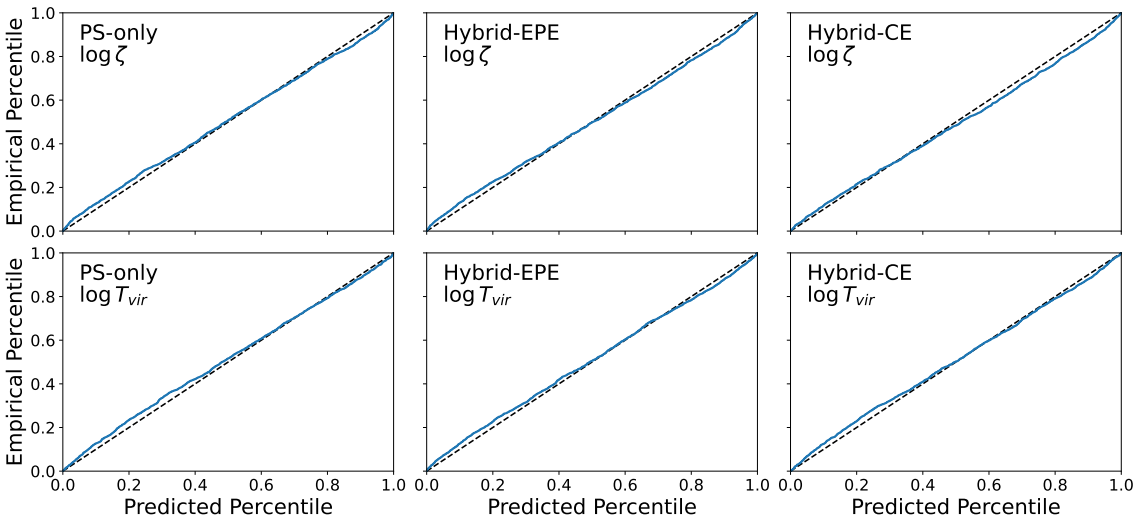


Figure 10.7: Posterior coverage for the 21 cm example across three summaries and two reionization parameters. The blue lines represent the actual calibration using 2,048 test samples, while the black dashed line indicates perfect calibration.

studies to measure how much more information might be unlocked with multiple traditional or neural statistics.

10.6 Acknowledgements

We thank Tao Jing, Justin Alsing, Tom Charnock, Niall Jeffrey, and David Spergel for conversations that inspired this work. This work was supported by the Simons Collaboration on "Learning the Universe". TLM acknowledges the support of the Imperial College London President's Scholarship and G-Research for conference travel costs. CS is supported by the National SKA Program of China (grant No. 2020SKA0110401) and NSFC (grant No. 11821303).

10.7 Appendix

Posterior Coverage. In Figure 10.4, we present the posterior for a single test sample. To demonstrate that these results are not due to overfitting, we also show calibration results for a test set of 2,048 samples. Posterior coverage is used as a validation metric, as shown in Figure 10.7. The predicted percentiles closely match the empirical percentiles, indicating that our summaries are robust and the SBI inference is neither overly confident nor conservative in any case.

CHAPTER 11

HYBRID STATISTICS PART III: DARK ENERGY

DES Y3 with Hierarchical Hybrid Statistics

T. Lucas Makinen[1], Josh Williamson[2], Natalia Porqueres[3], Alan Heavens[1], Niall Jeffrey[2], Benjamin D. Wandelt[4,5,6]

[1]Imperial Centre for Inference and Cosmology (ICIC) & Astrophysics Group,
Imperial College London, Blackett Laboratory, Prince Consort Road, London SW7 2AZ, UK
[2]Department of Physics & Astronomy, University College London, Gower Street, London, WC1E 6BT, UK
[3]Université Paris-Saclay, Université Paris Cité, CEA, CNRS, AIM, 91191, Gif-sur-Yvette, France
[4]Department of Physics and Astronomy, Johns Hopkins University, 3400 North Charles Street, Baltimore, MD 21218, USA
[5]Department of Applied Mathematics and Statistics, Johns Hopkins University, 3400 North Charles Street, Baltimore, MD 21218, USA
[6]CNRS & Sorbonne Université, Institut d'Astrophysique de Paris (IAP), UMR 7095, 98 bis bd Arago, F-75014 Paris, France

11.1 Introduction

In the last chapter, we extended the hybrid statistics formalism from a local Fisher Matrix optimisation to a compression problem over parameter space using the Mutual Information and two tractable lower bounds to optimise the objective given data and parameter samples from the joint distribution. We showed that incorporating domain knowledge in the form of the two-point function helped networks acting on the full data learn more efficiently from a limited number of simulations, indicating that the two-point function gives networks a "leg up" during hybrid optimisation.

This chapter will extend the formalism introduced in Chapter 10 to a *hierarchy* of data compressions under a common objective. The motivation is that massive datasets like cosmological surveys cannot be easily fed into a single network to be summarised. In the case of catalogues of discrete objects, we handled this obstacle by constructing ragged graphs of the data and learned non-linear functions of an aggregation of features.

Weak lensing surveys like the Dark Energy Survey (DES) (DES Collaboration et al., 2018) or Hyper Suprime-Cam (HSC) (Dalal et al., 2023) often consist of more than one "patch" of cosmic shear measurements on the sky, but there is currently no prescription for how to combine non-linear compressions of these data, especially when subjected to different survey masks and cuts. One way (e.g. Lu et al., 2023) is to learn a single compression function over all patches, implicitly marginalising over differences in patch construction and masks. To compress three sky patches from the DES Y3 survey, Jeffrey et al. (2024) train separate networks on each piece of sky and average the resulting statistics together before performing inference. While demonstrating improvement over standard two-point lensing analyses, these methods lack firm information-theoretic guarantees.

Armed with the new hybrid statistics formalism, our explicit objective will be measurement of the weak lensing parameters (Ω_m, S_8) and the Dark Energy equation of state parameter, w using the Dark Energy Survey data products. We adopt the same network and simulation setup presented in Jeffrey et al. (2024) with a new loss function.

11.2 Hierarchical Hybrid Statistics

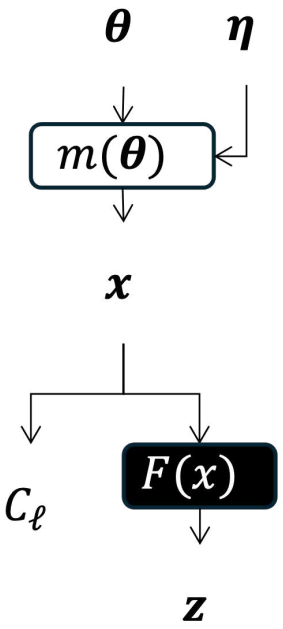


Figure 11.1: Plate diagram illustrating the joint distribution $p(\theta, x, C_\ell, z)$ via intermediate data quantity x and nuisance parameters η. We wish to optimise the function $z = F(x)$ given that we know C_ℓ.

To extend the hybrid statistics formalism in a hierarchical fashion we will consider a weak lensing data setup, where the full data x are image data, e.g. a convergence κ map, and a measured existing statistic measured, the power spectrum $y = C_\ell$, which is measured from x. We assume that we have access to samples from the joint distribution $p(\theta, x, C_\ell)$, illustrated in Fig. 11.1.

11.2.1 Single Data Compression

We first consider the compression of one data product, in this case the existing summary vector C_ℓ. We can write down the optimal compression function $J^*(C_\ell)$ as the J that maximizes the following mutual information:

$$I(J(C_\ell); \theta) = h(\theta) - h(\theta|J(C_\ell)) \tag{11.1}$$

for target parameters θ and differential entropy h. The second term is related to the expectation of the log posterior $p(\theta|J(C_\ell))$. This is the Variational Mutual Information Maximisation criterion (VMIM; Jeffrey et al. (2020)), and is lower-bounded by the Expected Posterior Entropy (EPE) minimisation (Hoffmann & Onnela, 2023).

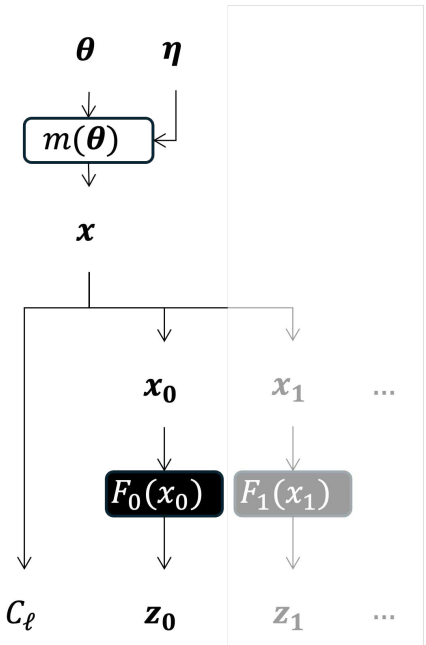


Figure 11.2: Plate diagram illustrating the joint distribution $p(\theta, x, C_\ell, z)$ via intermediate data quantity $x \to x_0, x_1$ and nuisance parameters η. We wish to optimise the function $z_0 = F_0(x_0)$ given that we know C_ℓ, and then $z_1 = F_1(x_1)$ given that we know C_ℓ, z_0.

11.2.2 Hybrid Compression

Now, let us assume we want to compress only x in a way that is optimal given we already have $t = C_\ell$, i.e. corresponding to $p(\theta|t, F(x))$. In this case, the mutual information is given by:

$$I(\theta; F(x)|t) = h(\theta|t) - h(\theta|t, F(x)). \tag{11.2}$$

The term $h(\theta|t)$ does not depend on F, so it does not impact the optimization. The trained compression function F^* is found by minimizing $h(\theta|t, F(x))$ via a surrogate density estimator. In the end, $I(\theta; F^*(x)|t) = I(\theta; x|t)$ for the optimal F.

11.2.3 Hybrid Statistics with full compression

How then should we use our learned map compression F^* and compress the resulting summary vector $[t, F^*(x)]$ in a way that the mutual information is still conserved? Consider the mutual information for the optimal F compression:

$$\begin{aligned} I^* = I(\theta; x|t) &= I(\theta; F^*(x)|t) \\ &= h(\theta|t) - h(\theta|[t, F^*(x)]) \end{aligned} \tag{11.3}$$

Define a new compression $G(t, F^*(x))$ that acts on the compressed map summaries $F^*(x)$ and t. As before, we can use the same loss to minimize $h(\theta|G(t, F^*(x)))$ to find the optimal second compression G^*. This affects only the last term in the above expression. If the optimum is found, then

$$h(\theta|t, x) = h(\theta|t, F^*(x)) = h(\theta|G^*(t, F^*(x))). \tag{11.4}$$

In some cases the existing summary vector $t = C_\ell$ may be prohibitively large to feed into a surrogate density estimator. In this case the existing summaries can be compressed using Eq. 11.1 to obtain $t = J^*(C_\ell)$. We can summarise the method as

1. Train the first compression F of the map with the VMIM loss, conditioned on the power spectrum $t = C_\ell$ (or compressed power spectrum $t = J^*(C_\ell)$) to minimise $h(\theta|t, F(x))$.
2. Train the second compression G of the power spectrum $t = C_\ell$ and compressed map summaries $F^*(x)$ with the VMIM loss: $h(\theta|G(t, F^*(x)))$.

This method can be generalized beyond two summaries (e.g. multiple patches of the map). However, if the map is split into patches (e.g. $x \to x_0, x_1$), and there are features of the data that cover multiple patches, then $I([x_0, x_1]; \theta) \neq I(x; \theta)$, even before the compression has started. Subsequent compressions can be learned on these data products conditioned on existing ones, as illustrated in Fig. 11.2. We formalise this procedure in the form of an aggregated statistic $\mathbf{t}$ in Algorithm 1.

Algorithm 1 Calculation of hierarchical hybrid summary statistics

Generate parameter and data-tuple pairs: $\{\theta_i, \mathbf{x}_i\} \sim p(\theta, \mathbf{x}) = p(\theta, (x_1, ..., x_n))$.
for $j \leftarrow 1$ to n **do**
  initialise network $F_j(x_j; \mathbf{w}_j)$, summary $\mathbf{t}$, NDE q_j
  optimise $\min_{q_j, \mathbf{w}_j} \mathbb{E}_{p(\theta,\mathbf{x})} \left[-\log q_j(\theta | \left[\mathbf{t}, F_j(x_j; \mathbf{w})\right])\right]$
  $\mathbf{t} \leftarrow [\mathbf{t}, F_j^*(x_j)]$ ▷ concatenate new summary
end for
return t
Perform density estimation using compressed summaries $p(\theta|\mathbf{t}) \propto p(\mathbf{t}|\theta)p(\theta)$

Loss Function. Our loss function is explicitly a function of both compression network weights $\mathbf{w}$ and NDE surrogate q. Our loss function for a given data product j is

$$\mathcal{L}(q, \mathbf{w}_j) = \mathbb{E}_{p(\theta,\mathbf{x})} \left[-\log q(\theta | \left[t, F_j(x_j; \mathbf{w}_j)\right])\right] . \tag{11.5}$$

The dimensionality of each summary $z_j = F_j(x_j)$ is a hyperparameter that controls the bottleneck of the information flow to the density estimator $q(\theta|\cdot)$. The final, aggregated statistic vector can be compressed to a chosen dimensionality with the same loss.

11.3 Application to DES Y3 Data

The hierarchical scheme is ideally suited to weak lensing or large-scale structure inference because large-scale information is mostly Gaussian and well-captured by the power spectrum, while small-scale information obtained from cosmological field imaging can be learned iteratively by splitting the map-level data into sections and fed into neural networks. So long as the data products x_j contain sufficiently large scales (e.g. an overlap with Gaussian field scales), the splitting into sub-patches $x \to \{x_j\}$ can theoretically be made lossless.

11.3.1 DES Year 3 Shear Field

The Dark Energy Survey is a photometric galaxy survey that maps ~ 5000 square degrees of the Southern Galactic Cap. The Dark Energy Camera (DECam, Flaugher et al. (2015)) is a 570-megapixel device mounted on the Cerro Tololo Inter-American Observatory Blanco Telescope in Chile that images the extragalactic field in *grizY* colour filters. The Year 3 data release measured $100,204,026$ galaxies, resulting in a weighted effective galaxy number density $n_{\rm eff} = 5.59$ galaxies arcmin^{-2} over $4,139$ square degrees. The cosmic shear field is estimated from this catalogue using the `METACALIBRATION` algorithm (Sheldon & Huff, 2017; Huff & Mandelbaum, 2017), which utilises a self-calibration method to correct for selection effects in the form of a multiplicative bias during shear measurement from multiband, noisy images of observed galaxies to the 2 or 3-percent level (MacCrann et al., 2022).

Once estimated, the mean-subtracted shear map is divided into four tomographic bins with roughly equal galaxy number densities, and then projected onto HEALPix (Górski et al., 2005) pixels with an `NSIDE` of 512. This lowers the resolution of the shear field, which removes small nonlinear scales which are difficult to model (Jeffrey et al., 2024). The estimated shear is given by

$$\gamma^{\nu}_{\rm obs} = \frac{\sum_j \epsilon^{\nu}_j w_j}{\bar{R} \sum_j w_j}; \tag{11.6}$$

where $\nu = 1, 2$ indexes the two shear components, w_j is the per-galaxy inverse variance weight, and $\bar{R}$ is `METACALIBRATION` response of the sample, and j indexes the galaxies in the given pixel.

11.3.2 Gower Street Simulation Suite

The Gower Street Simulations is a suite of 791 gravity-only full-sky N-body simulations, presented in Jeffrey et al. (2024), and created with the PKDGRAV3 code (Potter et al., 2016) over a hyperprior of seven wCDM cosmological parameters: $(\Omega_m, \sigma_8, n_s, h, \Omega_b h^2, w, m_\nu)$. The physical box size is $L = 1250h^{-1}$ Mpc with resolution $N = 1080$. N-body evolution is performed by calculating forces on all particles (akin to Eq 2.108) in phase space from initial conditions at $z_0 = 49$, perturbed using second-order Lagrangian Perturbation Theory. The initial conditions as well as complex structure formation are both influenced by the specified cosmological parameters (see e.g. Efstathiou et al., 1985). Snapshots of the simulations as they evolve over cosmic time are then projected onto an

observer's lightcone, restricting the worldlines of visible structures to those that intersect with the obeserver's point of view, and then binned in redshift between snapshots onto HEALPix pixels with `NSIDE` = 2048 on the sky.

Cosmological Parameters. As described in Jeffrey et al. (2024), each simulation is generated with a different draw of cosmological parameters from a prior *and* a unique random seed (see Jeffrey et al. (2024), Fig. 4). This is to ensure marginalisation of data variations that arise from cosmic variance. The priors for n_s, h, and $\Omega_b h^2$ were chosen were chosen to be normal distributions wide enough to be consistent with Planck (Planck Collaboration et al., 2021) and SH0ES (Riess et al., 2022). A log-uniform prior is adopted for m_ν.

The simulation sampling distribution for Ω_m and σ_8 were chosen using an *active learning* strategy (Alsing et al., 2018) using both existing DES constraints and iteratively using simulations already produced. The resulting distribution of simulations is more concentrated in areas of high posterior probability and is *not* representative of the priors to be used for inference. We detail how to reconcile this effect in Section 11.4.1.

The Dark Energy equation of state parameter is also sampled from a mixed distribution. Most of the simulations come from $w \sim \mathcal{N}(-1, 1/3)$, with values $w < -1$ and $w > -1/3$ discarded. Approximately 64 simulations are retained for $w < -1$, corresponding to phantom dark energy. The stability of the N-body simulations (e.g. when paired with low Ω_m values) under these more extreme w values were validated, and using these simulations for compression and density estimation helps mitigate hard prior boundary of $w = -1$ enforced at the inference stage. Since the distribution of simulations does *not* match the priors we wish to enforce for inference, a neural likelihood estimation scheme must be adopted (see Section 11.4.1).

11.3.3 N-body to Weak Lensing Fields

From the suite of N-body simulations, weak lensing mocks can be calculated using the relationships of matter overdensity, gravitational and lensing potentials described in Section 2.4.2 and displayed in Fig 2.4. For details behind DES-specific implementation see Jeffrey et al. (2021, 2024). The overdensity field is represented as a set of lens planes over 100 shells. For each shell s, the overdensity is calculated via

$$\delta_{\rm shell}(\phi, s) = \frac{n_{\rm part}(\phi, s)}{\langle n_{\rm part}(\phi, s)\rangle_\phi} - 1, \tag{11.7}$$

where $n_{\rm part}(\phi, s)$ denotes the number of (dark matter) particles in pixel ϕ for shell s, and the average is taken over pixels. The lens planes are converted to convergence $\kappa_{\rm shell}(\phi, \chi)$ using the Born approximation and ray tracing in the line-of-sight direction. The HEALPix resoltion is downsampled from `NSIDE`=2048 to 512, corresponding to a physical pixel size of 7.2 arcmin. The output harmonically-related (Eq 2.79) κ and shear γ maps are the noise-free, true fields in redshift shells. To recover observed fields, these shells must be i) integrated over a redshift distribution $n(z)$ of source galaxies, ii) incorporate the effect galaxies' *intrinsic alignments*, and iii), be corrupted by galaxy shape noise and survey masks.

11.3.4 Intrinsic Alignments.

Intrinsic alignments (IA) of galaxies are physical correlations that arise from galaxy shapes, spins, and tidal forces from their dark matter halo habitats (see Lamman et al., 2024, for a comprehensive introduction). These correlations interfere with the distortions one hopes to measure with respect to weak gravitational lensing along the observer's line of sight, so can bias cosmological inference if not accounted for in a forward model or likelihood. Jeffrey et al. (2024) model IA using the weighted Non-Linear Alignment (NLA) model (Hirata & Seljak, 2003; Bridle & King, 2007). The approach models the shear field as it *would* appear without lensing. This term is then added to the lensing shear and integrated over the redshift distribution (Eq. 11.10). The IA-only convergence is

$$\kappa_{\rm IA}(\phi, z) = -A_{\rm IA} C_1 \epsilon_{\rm crit} \frac{\Omega_m}{D(z)} \left(\frac{1+z}{1+z_0}\right)^{\eta_{\rm IA}} \delta(\phi, z), \tag{11.8}$$

for a pixel ϕ z_0=0.62, and $C_1 = 5 \times 10^{-14} M_\odot h^{-2} {\rm Mpc}^2$. The free IA parameters are varied over the uniform priors $A_{\rm IA} \sim \mathcal{U}(-3, 3)$ and $\eta_{\rm IA} \sim \mathcal{U}(-5, 5)$ in the forward simulations. The shear $\gamma_{\rm IA}$ is obtained from the convergence using Eq 2.79.

11.3.5 Source Clustering

The source galaxies which trace the underlying structure have a tendency to cluster in higher-density regions. This means that the redshift distribution of sources $n(z)$ is in general not constant across the sky (Gatti et al., 2023). To account for this, the simulations introduce modulated per-pixel redshift

using the sky-averaged redshift distribution $\bar{n}(z)$ via:

$$n(z) \propto \bar{n}(z)(1 + b_g\delta(\phi), \tag{11.9}$$

where a linear galaxy biasing model is parameterised by $b_g = 1$. Misspecification of bias can be detected in higher-order statistics, as reported in Gatti et al. (2023).

Altogether, the per-pixel shear is integrated in redshift bins via

$$\gamma(\phi) = \frac{\sum_z \bar{n}(z)\,[1 + b_g\delta(\phi, z)]\,(1 + m_b)[\gamma(\phi, z) + \gamma_{\rm IA}(\phi, z)]}{\sum_z \bar{n}(z)[1 + b_g\delta(\phi, z)]} + \left(\frac{\sum_z \bar{n}(z)}{\sum_z \bar{n}(z)[1 + b_g\delta(\phi, z)]}\right)^{1/2} F_n(\phi)\frac{\sum_g w_g e_g}{\sum_g w_g}, \tag{11.10}$$

where $\bar{n}(z)$ is obtained from a `HYPERRANK` (Cordero et al. (2022); see Jeffrey et al. (2024), Fig. 5) sample varied between each simulation, m_b is a multiplicative shear bias varied for each tomographic bin b, $F_n(\phi) = A(1-B\sigma_e^2(\phi))^{1/2}$ is a factor that rescales the shape noise, with fixed parameters (A, B) for each of the four tomographic bins, and $\sigma_e^2(\phi)$ is the shape noise variance per pixel estimated from random rotations of real DES galaxies.

11.3.6 Map and Patch Construction

To form the DES Y3 footprint, the HEALPix shear maps are then masked and degraded to a resolution of `NSIDE`=512, corresponding to a scale cut of 6.9 arcmin. This cut in pixel space translates to a smooth, 30 percent suppression of power in harmonic space at $\ell = 1024$. A further hard cut is then performed in harmonic space at $\ell = 1024$. Reconstructed convergence κ maps are then constructed using the Kaiser-Squires (Kaiser et al., 1995) and Eq 2.79. To form patches of the data, the DES footprint in κ is split into three equally-sized patches corresponding to all `NSIDE`=512 pixels that fall within the minimum HEALPix resolution of `NSIDE`=1 pixels. These patches are labelled "A", "B", and "C" (see Fig. 11.3. Each of these 512^2 images consists of four channels in redshift and three unique channels for EE, EB, and BB modes.

11.3.7 Angular Power Spectra

As an existing statistic, the power spectrum $C(\ell)$ is calculated on the sphere via

$$\langle a_{\ell m} a^*_{\ell m'} \rangle_{\text{realisation}} = C(\ell)\delta_{mm'}\delta_{\ell\ell'}, \tag{11.11}$$

where $a_{\ell m}$ are spherical harmonic coefficients of the field, δ_{mm} is the Kronecker delta. The unbiased estimator or (also called "pseudo-C_ℓ" or empirical power spectrum) taken from spherical maps is

$$\hat{C}_\ell = \frac{1}{2\ell+1} \sum_{m=-\ell}^{\ell} |a_{\ell m}|^2, \tag{11.12}$$

which is measured from the shear field (see Jeffrey et al. (2024), Fig. 6). The measured shear field is decomposed into E- and B- modes, yielding power spectra C_ℓ^{EE}, $C_\ell^{EB} = C_\ell^{BE}$ and C_ℓ^{BB}. For data and network input, we utilise C_ℓ^{EE} and C_ℓ^{BB}, following Jeffrey et al. (2024).

11.3.8 Mean Square Error Compression Scheme

Jeffrey et al. (2024) implement a suite of "weak learner" networks to form summary statistics from each of the data products independently. For each of the three cosmological parameters, 12 networks are initialised independently for each data product (C_ℓ and patches) and optimised via, for e.g. Ω_m

$$\mathcal{L} = -\frac{1}{n_{\text{batch}}} \sum_{k}^{n_{\text{batch}}} (\hat{\Omega}_m^k - \Omega_m^k)^2 \tag{11.13}$$

To aggregate the output point estimates, a weighted average of all patch summaries based on validation loss is performed, and concatenated to the weighted average of 12 MLP networks acting on the C_ℓs to yield 6 summaries per simulation. This uninformed concatenation is similar to the comparison method ("network (separate) + C_ℓ") in Fig 10.5.

11.3.9 Hierarchical Hybrid Statistics Implementation

Our objective is to obtain joint constraints on the three-dimensional weak lensing-Dark Energy parameter set (Ω_m, S_8, w). For our explicit implementation of the procedure, we choose to parameterise

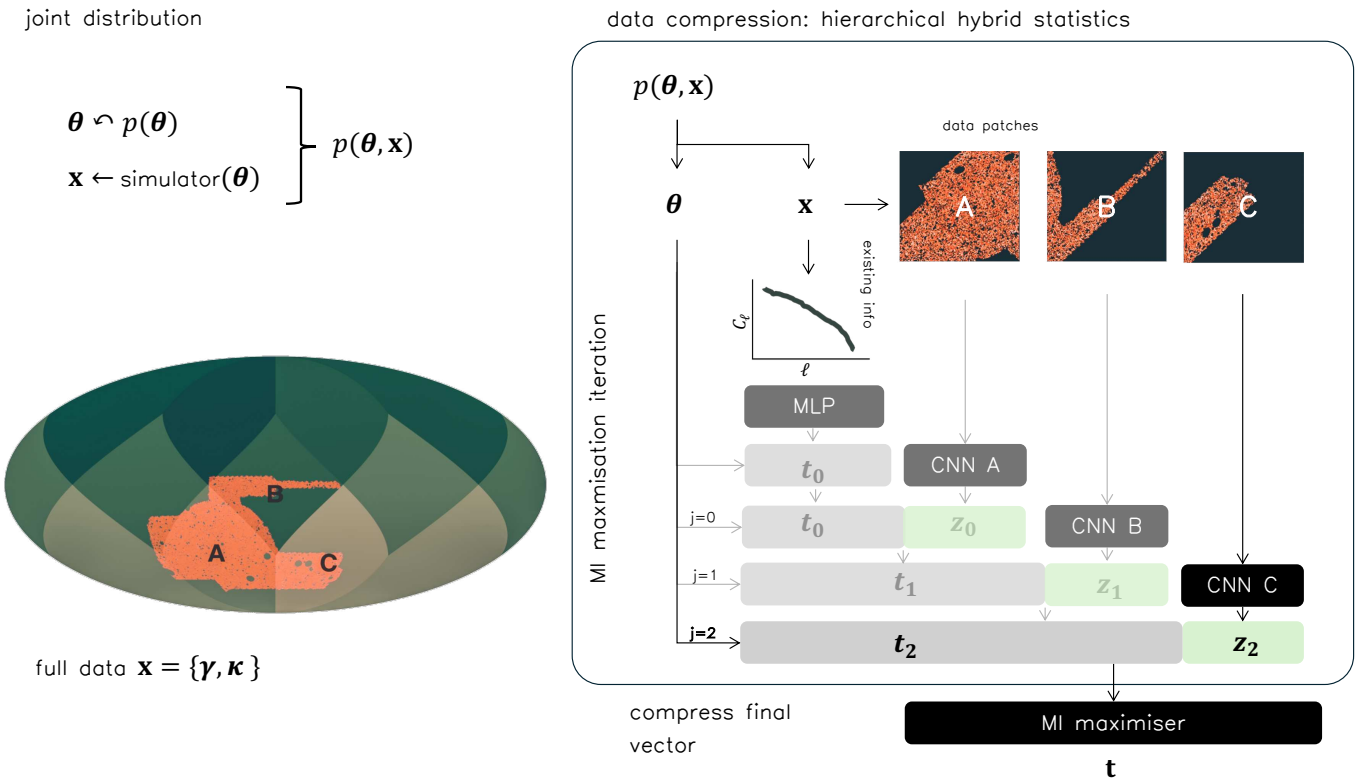


Figure 11.3: Hierarchical hybrid statistics scheme adapted for DES Y3 data products. The DES footprint is simulated over a prior in cosmology with nuisance parameters (*left*). Empirical C_ℓ vectors are computed from the full shear field (γ) footprint and compressed via MI maximisation to 10 numbers. The reconstructed convergence field κ is split into patches A,B, and C and passed hierarchically through separate CNNs to $z_j = 4$ numbers each to complement existing summaries in sequence (see Alg. 1). The resulting 22 numbers are finally compressed to a summary vector $\mathbf{t}$ of six numbers.

the mutual information maximiser surrogate $q(\theta|\cdot)$ with a simple mixture density network (MDN) (Bishop, 1994) comprised of a mixture of four Gaussians and single ReLU-activated hidden layer of width 100. We found empirically that simpler MDN configurations performed better, that is, the gradient descent optimisation is dominated by changes in the compression networks F_j at each stage of the procedure, as opposed to the density estimation. We detail each stage of the compression, illustrated in Fig 11.3 below.

Initial C_ℓ compression. The E- and B-mode C_ℓ vector with appropriate cuts is 440 numbers long. We use the same fully-connected network employed by Jeffrey et al. (2024) with an output summary size of $\dim(\mathbf{z}_{C_\ell}) = 10$. We chose this summary dimensionality to allow for a large enough bottleneck for information to propagate from summaries to NDE $q(\theta|\cdot)$.

Patch Compression. Here we use the same CNN architecture used in Jeffrey et al. (2024) to compress each patch down to $\dim(\mathbf{z}_{\text{patch}}) = 4$ numbers to accompany the C_ℓ summaries, yielding a final concatenated summary vector of 22 numbers. In principle, one could reuse the weights and biases of preceding networks (including q) to speed up training. In practice, however, we initialise each network anew for each data product's compression. We operate on patches A, B, and C in sequence, but note that other permutations might yield different and / or better results, as the hybrid formalism is not permutation-invariant. Patch compression in hierarchical fashion expresses *what the network can learn from patch B, given that it has already looked at C_ℓ and patch A.*

Final compression. We compress the final accumulated hybrid statistic vector of length 22 down to $\dim(\mathbf{t}) = 6$ numbers using an EPE loss, matching the dimensionality used in Jeffrey et al. (2024). For comparison to the C_ℓ summaries, we separately compress those 10 numbers to 4.

11.3.10 Neural Density Estimation

Simulation-based inference circumvents the need for a tractable likelihood $p(\mathbf{d}|\boldsymbol{\theta})$, and instead seeks to parameterise the underlying, implicit likelihood or posterior present in forward simulations of the data. Neural density estimators (NDEs; e.g. Bishop, 1994) use neural networks that give some estimate $q(\boldsymbol{\theta}, \mathbf{x}; \varphi)$ of the desired conditional probability distribution by varying weights and biases

(parameterised as φ) to minimize the loss

$$U(\varphi) = -\sum_{i=1}^{N} \ln q(x_i|\theta_i;\varphi), \tag{11.14}$$

over batches of parameter-data samples drawn from the joint distribution $(\boldsymbol{\theta}_i, \mathbf{x}_i) \sim p(\boldsymbol{\theta}, \mathbf{x}_i)$. This loss is equivalent to minimising the Kullback-Leibler divergence between the target distribution and q (Kullback & Leibler, 1951). In this work we employ Masked Autoregressive Flows (MAF; Papamakarios et al., 2017) to model a surrogate likelihood function (NLE; Neural Likelihood Estimation), following the same architecture prescription as Jeffrey et al. (2024). This allows us to learn the hierarchical compression over the strange simulation sampling distribution in the target parameters. We employ four MAFs with hidden sizes 40 or 50 with 3,5, or 6 transformations within the `ltu-ili` package (Ho et al., 2024). To deal with the hard prior boundary at $w = -1.0$ in our analysis, we minimise Eq. 11.14 using a training suite with simulations $w < -1.0$ to allow the learned likelihood to estimate densities consistent with $w = -1.0$. We ensure that posteriors and coverage tests are insensitive to the chosen $w < -1.0$ cutoff, e.g. that we have enough simulations to reconstruct densities at low w values.

11.3.11 Neural Posterior Coverage

To validate the density estimation scheme, we perform coverage tests to check whether credible intervals contain the expected probabilities, or fraction of simulations. We can view the posterior inference, which yields $p(\theta|x_O)$ at a given observed data x_O as a way to obtain a credible interval for a parameter θ. We draw test parameters $\theta_{\rm test}$ from the prior specified $p(\theta)$, which yields output data or summary $x_{\rm test}$, for which a posterior can be obtained $p(\theta|x_{\rm test})$. For a pre-defined fixed interval (e.g. 68%), the true test parameter value should fall in this interval 68% of the time. This procedure can be repeated with many $\theta_{\rm test}$ drawn from the prior, from which the *expected* coverage can be computed, enhancing confidence in the estimation of the true posterior. We utilise Lemos et al.'s generalised coverage "Test of Accuracy with Random Points" (TARP) algorithm to check coverage in high dimensions.

Parameter	Prior Probability Distribution
Ω_m	$\mathcal{U}(0.15, 0.52)$
S_8	$\mathcal{U}(0.5, 1.0)$
w	$\mathcal{N}(-1, \frac{1}{3})$

Table 11.1: Prior probability distribution used for inference.

11.4 Results

11.4.1 Neural Likelihood & Coverage tests

In our analysis, we obtain the posterior from a surrogate for the likelihood via an affine MCMC scheme; $p(\theta|x_{\text{test}}) \propto p(x_{\text{test}}|\theta)p(\theta)$. We repeat the inference procedure using 100 test simulations from a held-out noise realisation from the Gower Street suite, resampled without replacement via rejection sampling to adhere to the specified cosmology priors. We pass these simulations through the hybrid scheme to obtain summaries, for which we repeat the MCMC inference procedure using the NDE likelihood surrogate ensemble. We then feed these posterior chains to the `TARP` package to estimate the coverage probabilities in the $\{\Omega_m, S_8, w\}$ parameter space. We display an example coverage test for the full hybrid statistic vector (all data products) in Fig. 11.4. The bootstrapped expected coverage over different `TARP` calculations is consistent with the credibility level, indicating that the estimated posterior distribution does represent the true, underlying parameter distributions given the observed data.

11.4.2 Systematics & Robustness Testing

Before applying the compression and inference scheme to real data, it is imperative to test that the statistics used in the pipeline are robust to sources of systematic error, or deviations from the training set distribution. We use 320 simulations from the `CosmoGridV1` simulation suite (Kacprzak et al., 2023) as an independently-created test set before real-data deployment. The simulations all share the same fiducial cosmology, $\sigma_8 = 0.84,\ \Omega_m = 0.26,\ w = -1,\ H_0 = 67.36,\ \Omega_b = 0.0493,\ n_s = 0.9649$, and were created using the `PKDGRAV3` code (Potter et al., 2016). For each source of systematic error we generate two sets of mock data with different levels of error to test the effect on the downstream posterior distributions for cosmological parameters.

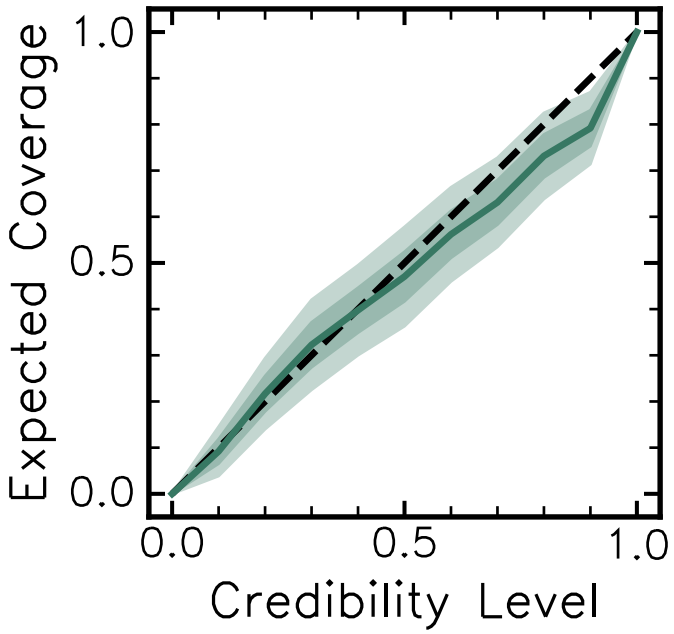


Figure 11.4: Coverage test result (using `TARP`; Lemos et al. (2023a)) to validate the density estimation scheme for hybrid statistics. Using repeated mock data parameter inference, the fraction of true values in the appropriate credible intervals matches the expected fraction. The figure shows the result for all three patches and C_ℓ compression.

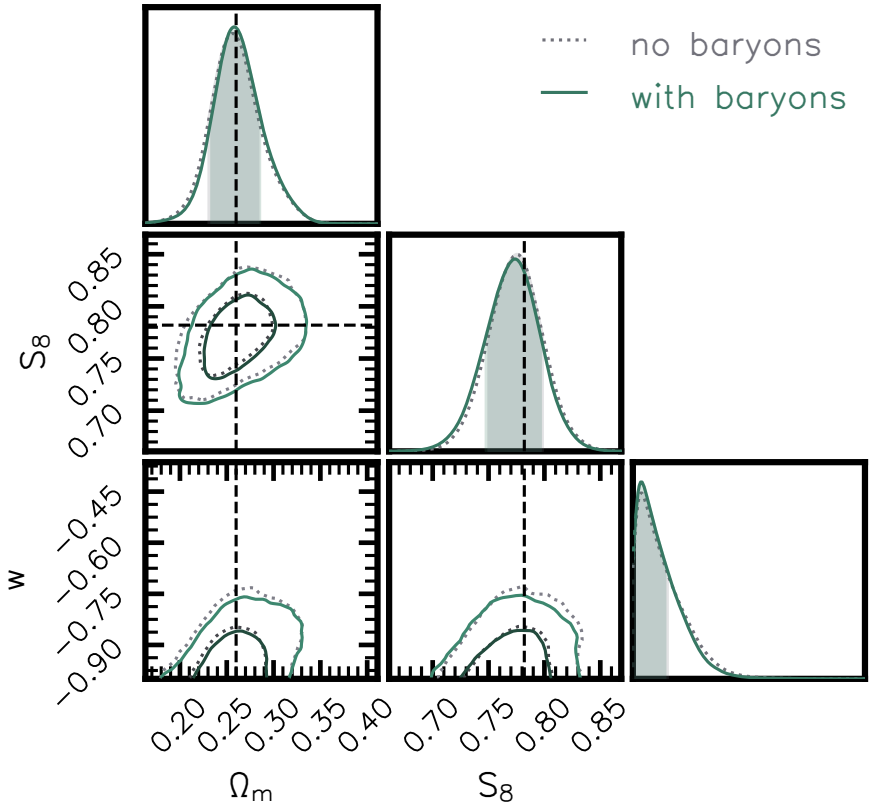


Figure 11.5: Hybrid statistics are robust to two sources of systematic error in mock test data: baryon feedback and changing source galaxy bias. Both systematics induce changes in the posterior below 0.3σ in the marginal $\Omega_m - S_8$ plane, as well as in the w marginal. The results also demonstrate successful recovery of the true parameter values from independent mock data.

Parameter	C_ℓ	hybrid statistics	DES Y3 $C_\ell\times$ CNN
Ω_m	$0.283^{+0.026}_{-0.025}$	$0.258^{+0.027}_{-0.024}$	$0.283^{+0.020}_{-0.027}$
S_8	0.789 ± 0.024	$0.774^{+0.022}_{-0.027}$	$0.804^{+0.025}_{-0.017}$
w	$-0.68^{+0.18}_{-0.26}$	< -0.914	< -0.803

Table 11.2: Comparison of marginal posterior probability constraints explored in this work on mock data compared with real-data DES Y3 constraints. Although different inferences, the measurements are consistent with hybrid statistics offering significant improvement in w constraints.

We perform the same tests on averaged compressed statistics $\langle \mathbf{t} \rangle_{i=1}^{N}$ that Jeffrey et al. (2024) employed, but additionally explore how *each* individual target data in our test set's posteriors shift under changing systematics. The goal of this test is that a given posterior doesn't shift in the $\Omega_m - S_8$ plane and w marginal more than 0.3σ under systematic injection, in line with the standard DES criterion. Here we investigate both the *average statistic*'s posterior shift, as well as the average shift of multiple independently-analysed posteriors from the test set. One of the biggest sources of systematic error is *baryonic feedback*. At small scales, feedback effects from galaxy formation can lead to the apparent suppression of cosmological structure. The training simulations are dark-matter only simulations that *do not include* baryonic physics. The standard DES approach is to cut physical scales likely affected by baryons and ensure that introducing a change in systematic prescription does not contaminate the cosmological results. Full forward baryonic feedback models require expensive and ill-specified hydrodynamical simulations over the full cosmological volume, which are not available for DES-like volumes. The `CosmoGridV1` simulations employ a baryon correction model, which changes the N-body density fields in post-processing to emulate baryon feedback (Kacprzak et al., 2023).

We show the results of this test on an example test data vector in Fig. 11.5. We observe a very small effect of each systematic on the posterior distribution. This test was also passed for the C_ℓ-alone compression. In Figure 11.6 we show the posterior and marginals of another test data inference with baryonic effects compared to the summaries obtained from the C_ℓ-only compression, with summarised constraints in Table 11.4.2. These results are consistent with the Ω_m, S_8 measurement recorded by Jeffrey et al. (2024), with a forecasted factor of 2 improvement in w measurement. The physical interpretation for this improvement is that unlike a separately-trained CNN, the hybrid statistics approach leverages existing, large-scale cosmological information present in the power spectrum whilst training the patch-level CNNs to look for complementary small-scale information.

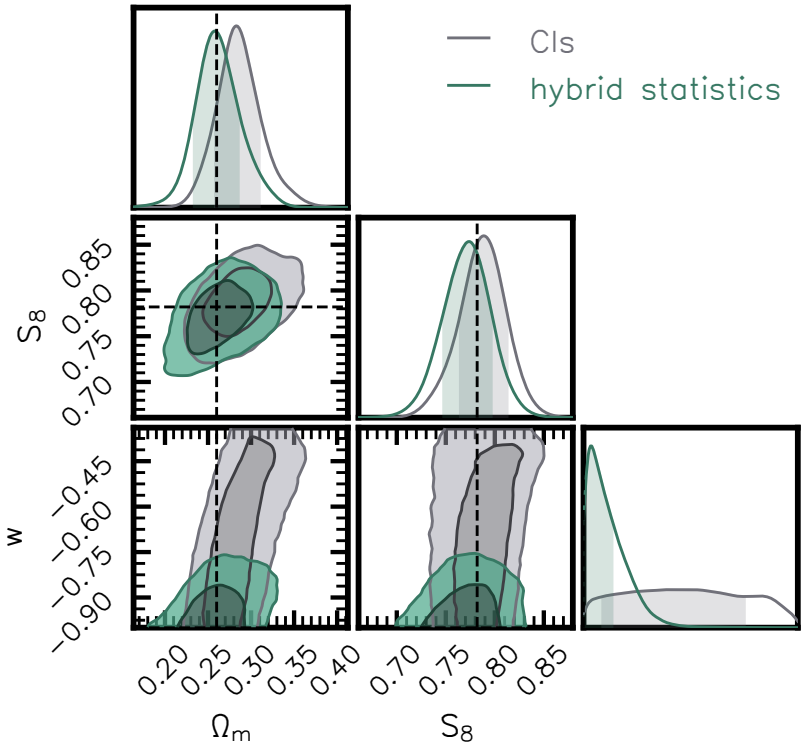


Figure 11.6: Hybrid statistics demonstrate remarkable improvement in w constraints over C_ℓ summaries.

11.4.3 Investigating Information Capture

The hierarchical hybrid statistics formalism casts information capture from subsequent data products as an information-update scheme. For another example baryon simulation, Fig 11.7 shows that information capture increases as more data products are aggregated in hierarchical fashion. Patch A (with the largest unmasked sky area) is the biggest contributor to the improvement over the C_ℓ posteriors, but incorporating optimised summaries for patches B and C appear to contribute incremental improvements in w and $\Omega - S_8$ degeneracy resolution. It is worth noting the difference in objective from the one explored in Jeffrey et al. (2024). Here we optimise summaries for the *joint* constraints on cosmological parameters, whereas Jeffrey et al. (2024) find summaries one parameter at a time. Our improvement in w could also stem from the fact that the loss function is variational and allows for the degeneracies in (Ω_m, S_8, w) to be explored thoroughly by the network as it learns useful summaries.

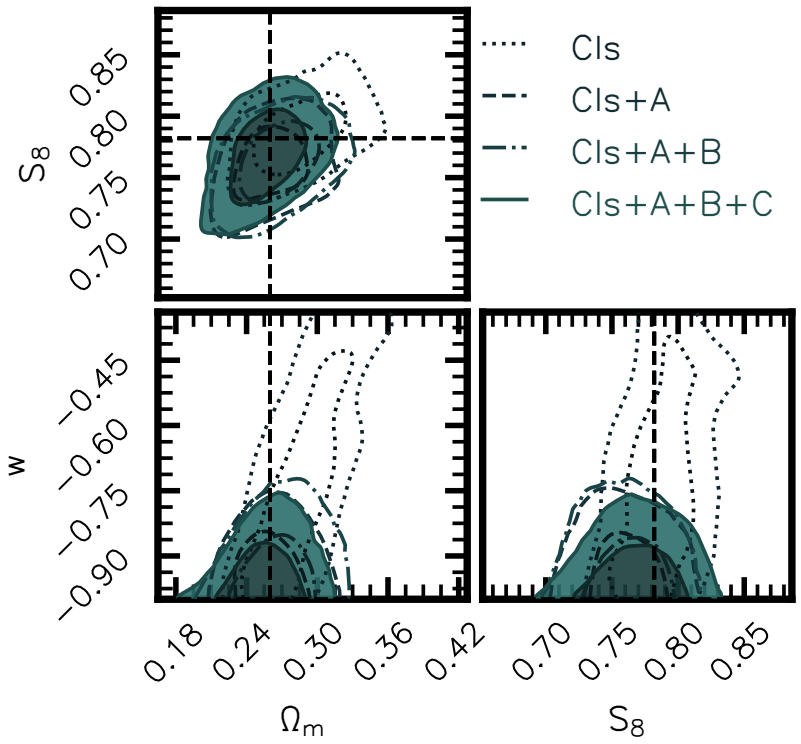


Figure 11.7: Information capture in cosmological parameters improves as more data products are compressed hierarchically for the same test simulation. Patch A demonstrably reduces uncertainties in all three parameters over the C_ℓ summaries, with patches B and C adding incremental improvements in w and $\Omega_m - S_8$ alignment.

11.4.4 Ablation Study: Where is the information coming from ?

We have so far successfully validated consistent information capture using hybrid statistics from the Gower Street DES simulation suite using the same neural network architectures to ensure consistency across studies. It is of interest to explore other architectures, particularly in the context of sourcing what data features contribute most to w measurement. We explore a modification to Jeffrey et al. (2024)'s CNN, in which the first "embedding" layers make use of the Multipole Kernels presented in Makinen et al. (2025) (see Chapter 9 for details). These kernels are weight-shared in particular symmetries that deviate from rotational (spherical) symmetry in a multipole expansion, and can be thought of as a "smooth, learnable filter" when passed over data to isolate physical patterns under noise artefacts. We train a MPK network with a 3×3 kernel and a `relu` activation using the hybrid scheme, and display the inference in Fig 11.8. Restricting the kernel to the multipole basis still yields consistent constraints with respect to the original architecture, indicating that the requisite data features used by the network do not deviate from this symmetry. We can exploit this kernel for an ablation study since the output CNN filters now have more physically-meaningful labels attached

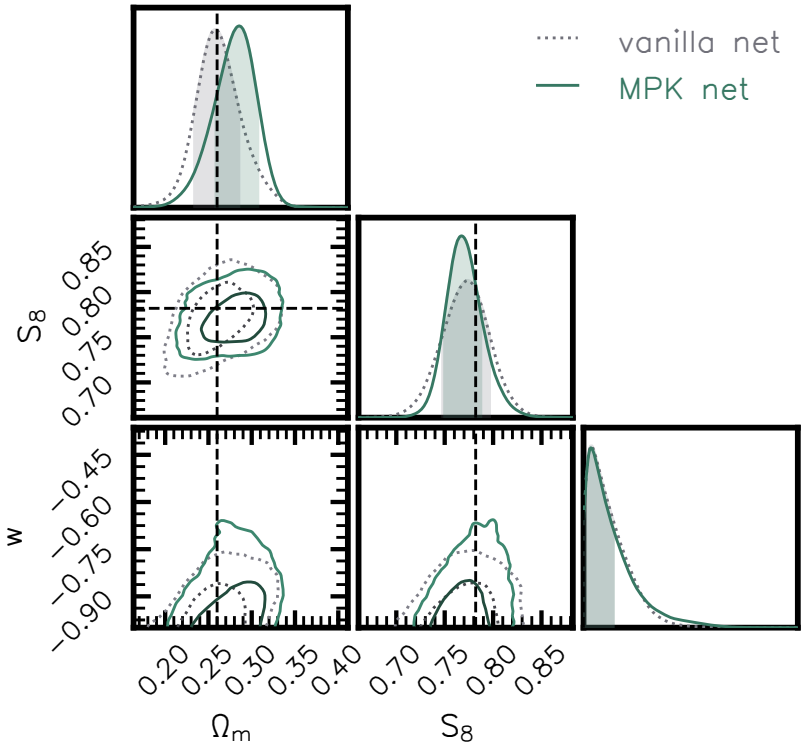


Figure 11.8: Imposing symmetry in CNN kernel weights captures consistent information with respect to a free 3×3 CNN kernel on the same target simulation. Constraints shown for hybrid compression of all data products.

to them; each filter is associated with a shared weight at a particular orientation from the centre of the kernel (as opposed to an unstructured set of features across 32 filters in the vanilla network). We display saliency maps in Fig 11.9 from three of the six learned MPK kernels from patch A, zoomed in on 256^2 sub-tile. The left-hand column shows a reference input data slice (first tomographic bin), the centre column displays the output of a randomly-initialised MPK kernel, and the last column shows the output using the learned kernel used in the network to produce the constraints in Fig 11.8. Qualitatively, these activation maps show that different types of features are isolated for each of the indexed output filters. The rotationally-symmetric kernel $\ell^k = 0$ appears to isolate small-scale features akin to cluster centres, the dipole $\ell^k = 1$ highlights meso-scaled features, and the quadrupole $\ell^k - 2$ finds larger-scale patterns. Future work could expand upon these observations by quantifying these nonlinear saliency maps in terms of features and cross-correlations, and perform posterior predictive checks using evidence networks to see how much predictive information is stored in each filter (Jeffrey & Wandelt, 2024). To link these features to parameters, one might further restrict CNN kernel freedom in training and analysing how saliency maps and parameter constraints degrade in response, or train downstream hybrid networks on a masked subset of the pre-trained MPK outputs.

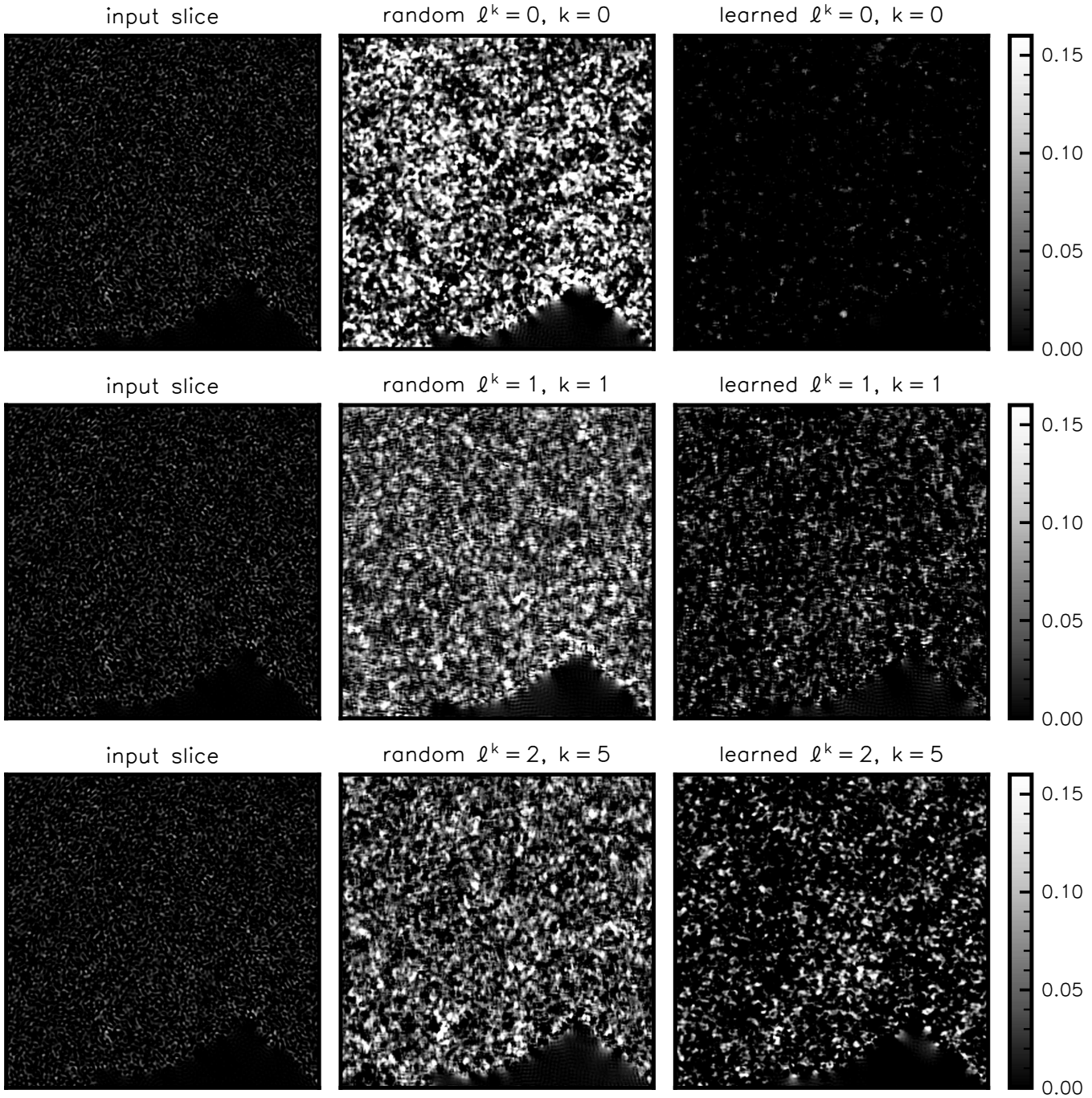


Figure 11.9: Example slices from random (*middle*) and learned (*right*) MPK embeddings on a patch A sub-tile, indexed by multipole ℓ^k. A reference input data (first tomographic bin) is shown. Learned kernels appear to identify small ($\ell^k = 0$, *top*), meso ($\ell^k = 1$, *middle*), and large-scale ($\ell^k = 2$, *bottom*) structures.

11.5 Discussion & Conclusion

We have presented a novel, information-theoretic hierarchical compression scheme for obtaining improved constraints on wCDM parameters from DES Y3 mock data. We leverage the same simulation-based inference setup presented in Jeffrey et al. (2024), with modification to the objective used when learning compression from simulated data products. Although performed on test simulations, this algorithmic improvement forecasts a factor of 2 improvement in marginal Dark Energy equation of state parameter measurement on real DES Y3 data.

CHAPTER 12

CONCLUSION

12.1 Summary of Thesis Achievements

This thesis cast modern cosmology as an optimisation problem to be accelerated by simulation-based inference and neural statistics. We presented an introduction to the cosmological theory behind large-scale structure formation and weak gravitational lensing measurement, and argued that higher-order statistics were needed to resolve parameter degeneracies and distinguish non-Gaussian morphologies from one another. We then unified several information-theoretic objectives for finding statistics from generic data, and coupled this to cosmology under neural implicit inference techniques. Chapter 6 illustrated a miniature "optimisation problem" using Information Maximising Neural Networks on toy cosmological fields with a non-Gaussianity parameter. Chapter 7 presented information-theoretic compression of arbitrary data manifolds (graphs) in the context of large-scale structure. This work interpreted learned statistics from the graph representation of structure as a non-linear combination of clustering and mass statistics. Chapter 8 tackled the problem of aggregation in set- and graph-based learning. Fishnets use the Fisher information matrix as a way to design a weight matrix to be learned alongside node or edge embeddings to make information capture lossless (with respect to known problems) and more efficient in common graph dataset benchmark tasks. The last portion of the thesis introduced "hybrid statistics" in three different parts. Beginning with local compression at a fiducial point, we built up the formalism towards generic compression over batches of parameter-data pairs over the joint distribution for a hierarchy of data products. We demonstrated this technique on Dark Energy Survey Y3 mock datasets, and showed that improvements in the optimisation objective is forecast to yield state-of-the-art Dark Energy measurement from weak gravitational lensing.

12.2 Future Work

In the near-term, future efforts will be dedicated towards validation and robustness checks for the results presented in Chapter 11 ahead of real-data inference. The hybrid statistics framework provides a means to interpret downstream parameter constraints' responses to systematic changes at the level of the data or intermediate compression (C_ℓ). An avenue to be explored is the effect of source clustering bias effects (e.g. Gatti et al., 2023) on hybrid statistics. These effects were discussed in Jeffrey et al. (2024), but the simulations required to test robustness in the present scheme have not been made available at the time of writing. An interesting approach might be to quantify deviations in constraints or network statistics as a function of a systematic parameter fixed during network training. Furthermore, leveraging an interpretable kernel basis for the convolutional network, as discussed in Section 11.4.4 might identify physical aspects or scales of the data that are encouraged to "light up" within the hybrid optimisation framework.

More broadly, future work in this intersection of cosmology, statistics, and machine learning should focus on new generative modelling methods (e.g. Pandey et al., 2024; Alsing et al., 2024) to improve the accuracy of forward and systematic models to mitigate poorly-understood effects like baryonic feedback in the context of cosmological inference. Bayesian model comparison, especially in simulation-based settings, will become essential as constraints in cosmology shrink and tensions between probes increase. Studies like Adame et al. (2025) who presented preference for evolving Dark Energy from Baryonic Acoustic Oscillation measurement can and should be replicated and verified using simulation-based methods, especially model comparison.

BIBLIOGRAPHY

Abadi, M., Agarwal, A., Barham, P., et al. 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org/

Adame, A., Aguilar, J., Ahlen, S., et al. 2025, Journal of Cosmology and Astroparticle Physics, 2025, 021, doi: 10.1088/1475-7516/2025/02/021

Adami, C., & Mazure, A. 1999, Astron. Astrophys. Suppl. Ser., 134, 393, doi: 10.1051/aas:1999145

Ade, P. A. R., Aghanim, N., Arnaud, M., et al. 2016, Astronomy & Astrophysics, 594, A13, doi: 10.1051/0004-6361/201525830

Alon, U., & Yahav, E. 2021, On the Bottleneck of Graph Neural Networks and its Practical Implications, doi: https://doi.org/10.48550/arXiv.2006.05205

Alpaslan, M., Robotham, A. S. G., Driver, S., et al. 2014, Monthly Notices of the Royal Astronomical Society, 438, 177, doi: 10.1093/mnras/stt2136

Alsing, J., Charnock, T., Feeney, S., & Wandelt, B. 2019, Monthly Notices of the Royal Astronomical Society, doi: 10.1093/mnras/stz1960

Alsing, J., Heavens, A., & Jaffe, A. H. 2016, Monthly Notices of the Royal Astronomical Society, 466, 3272, doi: 10.1093/mnras/stw3161

Alsing, J., Thorp, S., Deger, S., et al. 2024, The Astrophysical Journal Supplement Series, 274, 12, doi: 10.3847/1538-4365/ad5c69

Alsing, J., & Wandelt, B. 2018, Monthly Notices of the Royal Astronomical Society, 476, L60, doi: 10.1093/mnrasl/sly029

Alsing, J., Wandelt, B. D., & Feeney, S. M. 2018, arXiv e-prints, arXiv:1808.06040, doi: 10.48550/arXiv.1808.06040

Amari, S.-i. 2021, Japanese Journal of Mathematics, 16, 1, doi: 10.1007/s11537-020-1920-5

Amon, A., Gruen, D., Troxel, M. A., et al. 2022, , 105, 023514, doi: 10.1103/PhysRevD.105.023514

Artis, E., Melin, J.-B., Bartlett, J. G., & Murray, C. 2021, Astronomy & Astrophysics, 649, A47, doi: 10.1051/0004-6361/202140293

Asgari, M., Lin, C.-A., Joachimi, B., et al. 2021, Astronomy and Astrophysics, 645, A104, doi: 10.1051/0004-6361/202039070

Barber, D., & Agakov, F. 2004, Advances in neural information processing systems, 16, 201

Barrow, J. D., Bhavsar, S. P., & Sonoda, D. H. 1985, Monthly Notices of the Royal Astronomical Society, 216, 17, doi: 10.1093/mnras/216.1.17

Bartelmann, M., & Maturi, M. 2016, Weak gravitational lensing, arXiv, doi: 10.48550/arXiv.1612.06535

Bartelmann, M., & Schneider, P. 2001, Phys. Rept., 340, 291, doi: 10.1016/S0370-1573(00)00082-X

Bartlett, D. J., Chiarenza, M., Doeser, L., & Leclercq, F. 2025, Astronomy & Astrophysics, 694, A287, doi: 10.1051/0004-6361/202452217

Battaglia, P. W., Hamrick, J. B., Bapst, V., et al. 2018, Relational inductive biases, deep learning, and graph networks. https://arxiv.org/abs/1806.01261

Beuret, M., Billot, N., Cambrésy, L., et al. 2017, Astronomy and Astrophysics, 597, A114, doi: 10.1051/0004-6361/201629199

Bhavsar, S. P., & Ling, E. N. 1988, Publications of the Astronomical Society of the Pacific, 100, 1314, doi: 10.1086/132325

Bishop, C. 1994, Aston University. https://publications.aston.ac.uk/id/eprint/373/1/NCRG_94_004.pdf

Biswas, R., Alizadeh, E., & Wandelt, B. D. 2010, , 82, 023002, doi: 10.1103/PhysRevD.82.023002

Bommasani, R., Hudson, D. A., Adeli, E., et al. 2022, On the Opportunities and Risks of Foundation Models. https://arxiv.org/abs/2108.07258

Bonnaire, T., Aghanim, N., Decelle, A., & Douspis, M. 2020, Astronomy &amp Astrophysics, 637, A18, doi: 10.1051/0004-6361/201936859

Bonnaire, T., Aghanim, N., Kuruvilla, J., & Decelle, A. 2022, Astronomy &amp Astrophysics, 661, A146, doi: 10.1051/0004-6361/202142852

Boruah, S. S., & Rozo, E. 2023, Map-based cosmology inference with weak lensing – information content and its dependence on the parameter space. https://arxiv.org/abs/2307.00070

Bradbury, J., Frostig, R., Hawkins, P., et al. 2018, JAX: composable transformations of Python+NumPy programs, 0.3.13. http://github.com/google/jax

Bridle, S., & King, L. 2007, New Journal of Physics, 9, 444, doi: 10.1088/1367-2630/9/12/444

Bronstein, M. M., Bruna, J., Cohen, T., & Veličković, P. 2021, Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges, arXiv, doi: 10.48550/ARXIV.2104.13478

Carroll, S. M. 2019, Spacetime and Geometry (Cambridge University Press)

Carron, J., & Szapudi, I. 2013, Monthly Notices of the Royal Astronomical Society, 434, 2961, doi: 10.1093/mnras/stt1215

Castro, P. G., Heavens, A. F., & Kitching, T. D. 2005, Physical Review D, 72, 023516, doi: 10.1103/PhysRevD.72.023516

Cauchy, A. 1847, Comp. Rend. Sci. Paris, 25, 536

Charnock, T. 2019, Why neural networks don't work and how to use them. https://www.aquila-consortium.org/method/machinelearning/nn.html

Charnock, T., Lavaux, G., & Wandelt, B. D. 2018, Physical Review D, 97, doi: 10.1103/physrevd.97.083004

Charnock, T., Lavaux, G., Wandelt, B. D., et al. 2020, Monthly Notices of the Royal Astronomical Society, 494, 50–61, doi: 10.1093/mnras/staa682

Chartier, N., & Wandelt, B. D. 2021, Monthly Notices of the Royal Astronomical Society, stab3097, doi: 10.1093/mnras/stab3097

Chen, Y., Zhang, D., Gutmann, M., Courville, A., & Zhu, Z. 2021a, Neural Approximate Sufficient Statistics for Implicit Models. https://arxiv.org/abs/2010.10079

Chen, Y., Zhang, D., Gutmann, M. U., Courville, A., & Zhu, Z. 2021b

Cheng, S., Marques, G. A., Grandón, D., et al. 2024, Cosmological constraints from weak lensing scattering transform using HSC Y1 data. https://arxiv.org/abs/2404.16085

Cheng, S., Ting, Y.-S., Ménard, B., & Bruna, J. 2020, Monthly Notices of the Royal Astronomical Society, 499, 5902, doi: 10.1093/mnras/staa3165

Chevallier, M., & Polarski, D. 2001, International Journal of Modern Physics D, 10, 213, doi: 10.1142/S0218271801000822

Clevert, D.-A., Unterthiner, T., & Hochreiter, S. 2015, Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs), arXiv, doi: 10.48550/ARXIV.1511.07289

Colberg, J. M. 2007, Monthly Notices of the Royal Astronomical Society, 375, 337, doi: 10.1111/j.1365-2966.2006.11312.x

Cole, A., Miller, B. K., Witte, S. J., et al. 2022, Journal of Cosmology and Astroparticle Physics, 2022, 004, doi: 10.1088/1475-7516/2022/09/004

Coles, P. 2001, Large-Scale Structure, Theory and Statistics, arXiv, doi: 10.48550/arXiv.astro-ph/0103017

Coles, P., & Jones, B. 1991, MNRAS, 248, 1, doi: 10.1093/mnras/248.1.1

Coles, P., & Lucchin, F. 2002, Cosmology: The Origin and Evolution of Cosmic Structure, Second Edition. https://ui.adsabs.harvard.edu/abs/2002coec.book.....C

Coles, P., Pearson, R. C., Borgani, S., Plionis, M., & Moscardini, L. 1998, Monthly Notices of the Royal Astronomical Society, 294, 245, doi: 10.1046/j.1365-8711.1998.01147.x

Collaboration, P., Akrami, Y., Arroja, F., et al. 2019, Planck 2018 results. IX. Constraints on primordial non-Gaussianity, arXiv, doi: 10.48550/arXiv.1905.05697

Cordero, J. P., Harrison, I., Rollins, R. P., et al. 2022, Monthly Notices of the Royal Astronomical Society, 511, 2170, doi: 10.1093/mnras/stac147

Corso, G., Cavalleri, L., Beaini, D., Liò, P., & Velickovic, P. 2020a, CoRR, abs/2004.05718

Corso, G., Cavalleri, L., Beaini, D., Liò, P., & Veličković, P. 2020b, Principal Neighbourhood Aggregation for Graph Nets. https://arxiv.org/abs/2004.05718

Coulton, W. R., & Wandelt, B. D. 2023, How to estimate Fisher information matrices from simulations. https://arxiv.org/abs/2305.08994

Coulton, W. R., Villaescusa-Navarro, F., Jamieson, D., et al. 2022, Quijote-PNG: Simulations of primordial non-Gaussianity and the information content of the matter field power spectrum and bispectrum, arXiv, doi: 10.48550/ARXIV.2206.01619

Cramér, H. 1946, Mathematical methods of statistics, by Harald Cramer, .. (The University Press)

Cranmer, K., Brehmer, J., & Louppe, G. 2020a, Proceedings of the National Academy of Sciences, 117, 30055, doi: 10.1073/pnas.1912789117

Cranmer, M., Sanchez-Gonzalez, A., Battaglia, P., et al. 2020b, Discovering Symbolic Models from Deep Learning with Inductive Biases. https://arxiv.org/abs/2006.11287

Dai, B., & Seljak, U. 2022, arXiv e-prints, arXiv:2202.05282. https://arxiv.org/abs/2202.05282

Dai, B., & Seljak, U. 2024, Multiscale Flow for Robust and Optimal Cosmological Analysis. https://arxiv.org/abs/2306.04689

Dalal, R., Li, X., Nicola, A., et al. 2023, , 108, 123519, doi: 10.1103/PhysRevD.108.123519

Davis, M., Efstathiou, G., Frenk, C. S., & White, S. D. M. 1985, , 292, 371, doi: 10.1086/163168

DES Collaboration, Abbott, T. M. C., Allam, S., et al. 2018, arXiv e-prints. https://arxiv.org/abs/1811.02374

Devon Hjelm, R., Fedorov, A., Lavoie-Marchildon, S., et al. 2018, arXiv e-prints, arXiv:1808.06670, doi: 10.48550/arXiv.1808.06670

Dickey, J. M., Jiang, J.-M., & Kadane, J. B. 1987, Journal of the American Statistical Association, 82, 773. http://www.jstor.org/stable/2288786

Ding, S., Lavaux, G., & Jasche, J. 2024, PineTree: A generative, fast, and differentiable halo model for wide-field galaxy surveys. https://arxiv.org/abs/2407.01391

Dinh, L., Sohl-Dickstein, J., & Bengio, S. 2016, Density estimation using Real NVP, arXiv, doi: 10.48550/ARXIV.1605.08803

Doeser, L., Jamieson, D., Stopyra, S., et al. 2024, Monthly Notices of the Royal Astronomical Society, 535, 1258, doi: 10.1093/mnras/stae2429

Dyson, F. W., Eddington, A. S., & Davidson, C. 1920, Philosophical Transactions of the Royal Society of London Series A, 220, 291, doi: 10.1098/rsta.1920.0009

Efstathiou, G., Davis, M., White, S. D. M., & Frenk, C. S. 1985, , 57, 241, doi: 10.1086/191003

Efstathiou, G., & Silk, J. 1983, The Formation of Galaxies, Cosmic Physics. https://ned.ipac.caltech.edu/level5/March02/Efstathiou/Efst_contents.html

Einstein, A. 1916, Annalen der Physik, 354, 769, doi: 10.1002/andp.19163540702

Eisenstein, D. J., & Hu, W. 1999, The Astrophysical Journal, 511, 5–15, doi: 10.1086/306640

Euclid Collaboration. 2022, Astronomy and Astrophysics, 657, A91, doi: 10.1051/0004-6361/202141556

Euclid Collaboration, Ajani, V., Baldi, M., et al. 2023, A & A, 675, A120, doi: 10.1051/0004-6361/202346017

Feroz, F., Hobson, M. P., & Bridges, M. 2009, Monthly Notices of the Royal Astronomical Society, 398, 1601–1614, doi: 10.1111/j.1365-2966.2009.14548.x

Fey, M., & Lenssen, J. E. 2019, in ICLR Workshop on Representation Learning on Graphs and Manifolds

Flaugher, B., Diehl, H. T., Honscheid, K., et al. 2015, , 150, 150, doi: 10.1088/0004-6256/150/5/150

Fluri, J., Kacprzak, T., Lucchi, A., et al. 2019, Physical Review D, 100, doi: 10.1103/physrevd.100.063514

Fluri, J., Kacprzak, T., Lucchi, A., et al. 2022, arXiv e-prints, arXiv:2201.07771. https://arxiv.org/abs/2201.07771

Fluri, J., Kacprzak, T., Refregier, A., et al. 2018, Physical Review D, 98, doi: 10.1103/physrevd.98.123518

Fluri, J., Kacprzak, T., Refregier, A., Lucchi, A., & Hofmann, T. 2021, Physical Review D, 104, doi: 10.1103/physrevd.104.123526

Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. 2013, Publications of the Astronomical Society of the Pacific, 125, 306–312, doi: 10.1086/670067

Friedmann, A. 1922, Zeitschrift fur Physik, 10, 377, doi: 10.1007/BF01332580

Gatti, M., Jeffrey, N., Whiteway, L., et al. 2023, Detection of the significant impact of source clustering on higher-order statistics with DES Year 3 weak gravitational lensing data. https://arxiv.org/abs/2307.13860

Gelman, A., Carlin, J., Stern, H., et al. 2013, Bayesian Data Analysis, 3rd edn. (United States: CreateSpace)

Gillet, N., Mesinger, A., Greig, B., Liu, A., & Ucci, G. 2019, Monthly Notices of the Royal Astronomical Society, doi: 10.1093/mnras/stz010

Giovanni, F. D., Rusch, T. K., Bronstein, M., et al. 2024, Transactions on Machine Learning Research. https://openreview.net/forum?id=KJRoQvRWNs

Godwin, J., Keck, T., Battaglia, P., et al. 2020, Jraph: A library for graph neural networks in jax., 0.0.1.dev. http://github.com/deepmind/jraph

Goodfellow, I. J., Bengio, Y., & Courville, A. 2016, Deep Learning (Cambridge, MA, USA: MIT Press)

Górski, K. M., Hivon, E., Banday, A. J., et al. 2005, , 622, 759, doi: 10.1086/427976

Hahn, C., Villaescusa-Navarro, F., Castorina, E., & Scoccimarro, R. 2020, Journal of Cosmology and Astroparticle Physics, 2020, 040, doi: 10.1088/1475-7516/2020/03/040

Hamaus, N., Sutter, P., Lavaux, G., & Wandelt, B. D. 2015, Journal of Cosmology and Astroparticle Physics, 2015, 036–036, doi: 10.1088/1475-7516/2015/11/036

Hamilton, W., Ying, Z., & Leskovec, J. 2017, in Advances in Neural Information Processing Systems, ed. I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett, Vol. 30 (Curran Associates, Inc.). https://proceedings.neurips.cc/paper_files/paper/2017/file/5dd9db5e033da9c6fb5ba83c7a7ebea9-Paper.pdf

Hazan, E. 2017, Introduction to Online Convex Optimization (Independently Published). https://books.google.com/books?id=FBEqtAEACAAJ

Heavens, A. 2023, GR and Lensing, Imperial College London

Heavens, A. F., Jimenez, R., & Lahav, O. 2000, Monthly Notices of the Royal Astronomical Society, 317, 965–972, doi: 10.1046/j.1365-8711.2000.03692.x

Heek, J., Levskaya, A., Oliver, A., et al. 2020, Flax: A neural network library and ecosystem for JAX, 0.5.2. http://github.com/google/flax

Hendrycks, D., & Gimpel, K. 2016, Gaussian Error Linear Units (GELUs), arXiv, doi: 10.48550/ARXIV.1606.08415

Hirata, C., & Seljak, U. 2003, Monthly Notices of the Royal Astronomical Society, 343, 459, doi: 10.1046/j.1365-8711.2003.06683.x

Ho, M., Bartlett, D. J., Chartier, N., et al. 2024, LtU-ILI: An All-in-One Framework for Implicit Inference in Astrophysics and Cosmology. https://arxiv.org/abs/2402.05137

Hoffmann, T., & Onnela, J.-P. 2023, Minimising the Expected Posterior Entropy Yields Optimal Summary Statistics. https://arxiv.org/abs/2206.02340

Hu, W., Fey, M., Ren, H., et al. 2021, arXiv preprint arXiv:2103.09430

Hu, W., Fey, M., Zitnik, M., et al. 2020, arXiv preprint arXiv:2005.00687

Huff, E., & Mandelbaum, R. 2017, Metacalibration: Direct Self-Calibration of Biases in Shear Measurement, arXiv, doi: 10.48550/arXiv.1702.02600

Hyvärinen, A. 2005, Journal of Machine Learning Research, 6, 695. http://jmlr.org/papers/v6/hyvarinen05a.html

Ivanov, M. M., Cuesta-Lazaro, C., Mishra-Sharma, S., Obuljen, A., & Toomey, M. W. 2024, Full-shape analysis with simulation-based priors: constraints on single field inflation from BOSS. https://arxiv.org/abs/2402.13310

Jamieson, D., Li, Y., de Oliveira, R. A., et al. 2023, The Astrophysical Journal, 952, 145, doi: 10.3847/1538-4357/acdb6c

Jasche, J., Leclercq, F., & Wandelt, B. 2015, Journal of Cosmology and Astroparticle Physics, 2015, 036, doi: 10.1088/1475-7516/2015/01/036

Jasche, J., & Wandelt, B. D. 2013, Monthly Notices of the Royal Astronomical Society, 432, 894, doi: 10.1093/mnras/stt449

Jeffrey, N., Alsing, J., & Lanusse, F. 2020, Monthly Notices of the Royal Astronomical Society, 501, 954, doi: 10.1093/mnras/staa3594

Jeffrey, N., Boulanger, F., Wandelt, B. D., et al. 2022, Monthly Notices of the Royal Astronomical Society, 510, L1, doi: 10.1093/mnrasl/slab120

Jeffrey, N., & Wandelt, B. D. 2020, Solving high-dimensional parameter inference: marginal posterior densities & Moment Networks, arXiv, doi: 10.48550/ARXIV.2011.05991

—. 2024, Machine Learning: Science and Technology, 5, 015008, doi: 10.1088/2632-2153/ad1a4d

Jeffrey, N., Gatti, M., Chang, C., et al. 2021, Monthly Notices of the Royal Astronomical Society, 505, 4626, doi: 10.1093/mnras/stab1495

Jeffrey, N., Whiteway, L., Gatti, M., et al. 2024, Dark Energy Survey Year 3 results: likelihood-free, simulation-based wCDM inference with neural compression of weak-lensing map statistics. https://arxiv.org/abs/2403.02314

Joachimi, B., Taylor, A. N., & Kiessling, A. 2011, Monthly Notices of the Royal Astronomical Society, 418, 145, doi: 10.1111/j.1365-2966.2011.19472.x

Jung, G., Karagiannis, D., Liguori, M., et al. 2022, Quijote-PNG: Quasi-maximum likelihood estimation of Primordial Non-Gaussianity in the non-linear dark matter density field, arXiv, doi: 10.48550/ARXIV.2206.01624

Kacprzak, T., Fluri, J., Schneider, A., Refregier, A., & Stadel, J. 2023, Journal of Cosmology and Astroparticle Physics, 2023, 050, doi: 10.1088/1475-7516/2023/02/050

Kaiser, N., Squires, G., & Broadhurst, T. 1995, The Astrophysical Journal, 449, 460, doi: 10.1086/176071

Kilbinger, M. 2015, Reports on Progress in Physics, 78, 086901, doi: 10.1088/0034-4885/78/8/086901

Kingma, D. P., & Ba, J. 2014, arXiv e-prints, arXiv:1412.6980. https://arxiv.org/abs/1412.6980

Kingma, D. P., & Dhariwal, P. 2018, Glow: Generative Flow with Invertible 1x1 Convolutions, arXiv, doi: 10.48550/arXiv.1807.03039

Kipf, T. N., & Welling, M. 2017, Semi-Supervised Classification with Graph Convolutional Networks. https://arxiv.org/abs/1609.02907

Kitagawa, G. 1996, Journal of Computational and Graphical Statistics, 5, 1. http://www.jstor.org/stable/1390750

Kodi Ramanah, D., Charnock, T., Villaescusa-Navarro, F., & Wandelt, B. D. 2020, Monthly Notices of the Royal Astronomical Society, 495, 4227–4236, doi: 10.1093/mnras/staa1428

Koopman, B. O. 1936, Transactions of the American Mathematical Society, 39, 399, doi: 10.1090/S0002-9947-1936-1501854-3

Kratochvil, J. M., Haiman, Z., & May, M. 2010, Physical Review D, 81, doi: 10.1103/physrevd.81.043519

Kratsios, A. 2019, The Universal Approximation Property: Characterizations, Existence, and a Canonical Topology for Deep-Learning. https://arxiv.org/abs/1910.03344

Krause, E., & Hirata, C. M. 2010, Astronomy and Astrophysics, 523, A28, doi: 10.1051/0004-6361/200913524

Kreisch, C. D., Pisani, A., Villaescusa-Navarro, F., et al. 2021, The GIGANTES dataset: precision cosmology from voids in the machine learning era, arXiv, doi: 10.48550/ARXIV.2107.02304

Krzewina, L. G., & Saslaw, W. C. 1996, Monthly Notices of the Royal Astronomical Society, 278, 869, doi: 10.1093/mnras/278.3.869

Kullback, S., & Leibler, R. A. 1951, The Annals of Mathematical Statistics, 22, 79 , doi: 10.1214/aoms/1177729694

Kwon, Y., Hong, S. E., & Park, I. 2020, Journal of the Korean Physical Society, 77, 49–59, doi: 10.3938/jkps.77.49

Lamman, C., Tsaprazi, E., Shi, J., et al. 2024, The Open Journal of Astrophysics, 7, 10.21105/astro.2309.08605, doi: 10.21105/astro.2309.08605

Lanzieri, D., Lanusse, F., Modi, C., et al. 2023, Astronomy & Astrophysics, 679, A61, doi: 10.1051/0004-6361/202346888

Lanzieri, D., Zeghal, J., Makinen, T. L., et al. 2024, Optimal Neural Summarisation for Full-Field Weak Lensing Cosmological Implicit Inference. https://arxiv.org/abs/2407.10877

Lavaux, G., & Wandelt, B. D. 2010, Monthly Notices of the Royal Astronomical Society, 403, 1392, doi: 10.1111/j.1365-2966.2010.16197.x

Leclercq, F. 2015, Bayesian large-scale structure inference and cosmic web analysis, arXiv, doi: 10.48550/ARXIV.1512.04985

Leclercq, F., & Heavens, A. 2021, On the accuracy and precision of correlation functions and field-level inference in cosmology. https://arxiv.org/abs/2103.04158

Legin, R., Ho, M., Lemos, P., et al. 2023, Posterior Sampling of the Initial Conditions of the Universe from Non-linear Large Scale Structures using Score-Based Generative Models. https://arxiv.org/abs/2304.03788

Lehmann, E. L., & Casella, G. 1998, Theory of Point Estimation, 2nd edn. (New York, NY, USA: Springer-Verlag)

Lemos, P., Coogan, A., Hezaveh, Y., & Perreault-Levasseur, L. 2023a, Sampling-Based Accuracy Testing of Posterior Estimators for General Inference. https://arxiv.org/abs/2302.03026

Lemos, P., Jeffrey, N., Cranmer, M., Ho, S., & Battaglia, P. 2022, Rediscovering orbital mechanics with machine learning. https://arxiv.org/abs/2202.02306

Lemos, P., Parker, L., Hahn, C., et al. 2023b, SimBIG: Field-level Simulation-Based Inference of Galaxy Clustering. https://arxiv.org/abs/2310.15256

Li, G., Xiong, C., Thabet, A., & Ghanem, B. 2020, DeeperGCN: All You Need to Train Deeper GCNs. https://arxiv.org/abs/2006.07739

Li, S.-S., Hoekstra, H., Kuijken, K., et al. 2023, Astronomy and Astrophysics, 679, A133, doi: 10.1051/0004-6361/202347236

Li, Y., Lu, L., Modi, C., et al. 2022, pmwd: A Differentiable Cosmological Particle-Mesh N-body Library. https://arxiv.org/abs/2211.09958

Libeskind, N. I., van de Weygaert, R., Cautun, M., et al. 2018, Monthly Notices of the Royal Astronomical Society, 473, 1195, doi: 10.1093/mnras/stx1976

Linder, E. V., & Jenkins, A. 2003, Monthly Notices of the Royal Astronomical Society, 346, 573, doi: 10.1046/j.1365-2966.2003.07112.x

Livet, F., Charnock, T., Borgne, D. L., & de Lapparent, V. 2021, Catalog-free modeling of galaxy types in deep images: Massive dimensional reduction with neural networks. https://arxiv.org/abs/2102.01086

Loureiro, A., Whiteaway, L., Sellentin, E., et al. 2023, The Open Journal of Astrophysics, 6, doi: 10.21105/astro.2210.13260

Lu, T., Haiman, Z., & Li, X. 2023, Monthly Notices of the Royal Astronomical Society, 521, 2050, doi: 10.1093/mnras/stad686

MacCrann, N., Becker, M. R., McCullough, J., et al. 2022, Monthly Notices of the Royal Astronomical Society, 509, 3371, doi: 10.1093/mnras/stab2870

MacKay, D. J. C. 2002, Information Theory, Inference & Learning Algorithms (USA: Cambridge University Press)

Makinen, T. L., Alsing, J., & Wandelt, B. D. 2023, Fishnets: Information-Optimal, Scalable Aggregation for Sets and Graphs, arXiv, doi: 10.48550/arXiv.2310.03812

Makinen, T. L., Bartlett, D., Jeffrey, N., & Wandelt, B. D. in prep.a

Makinen, T. L., Charnock, T., Alsing, J., & Wandelt, B. D. 2021, Journal of Cosmology and Astroparticle Physics, 2021, 049, doi: 10.1088/1475-7516/2021/11/049

Makinen, T. L., Charnock, T., Lemos, P., et al. 2022, The Open Journal of Astrophysics, 5, doi: 10.21105/astro.2207.05202

Makinen, T. L., Heavens, A., Porqueres, N., et al. 2025, Journal of Cosmology and Astroparticle Physics, 2025, 095, doi: 10.1088/1475-7516/2025/01/095

Makinen, T. L., Lancaster, L., Villaescusa-Navarro, F., et al. 2020, deep21: a Deep Learning Method for 21cm Foreground Removal. https://arxiv.org/abs/2010.15843

Makinen, T. L., Pandya, V., & Ho, M. in prep.b

Makinen, T. L., Sui, C., Wandelt, B. D., Porqueres, N., & Heavens, A. 2024, Hybrid Summary Statistics. https://arxiv.org/abs/2410.07548

Makinen, T. L., Williamson, J., Jeffrey, N., et al. in prep.c

Martínez-Galarza, R., & Makinen, T. Lucas. 2022, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 12186, Observatory Operations: Strategies, Processes, and Systems IX, ed. D. S. Adler, R. L. Seaman, & C. R. Benn, 121860J, doi: 10.1117/12.2629504

Massara, E., Villaescusa-Navarro, F., Hahn, C., et al. 2022, Cosmological Information in the Marked Power Spectrum of the Galaxy Field, arXiv, doi: 10.48550/ARXIV.2206.01709

Massey Jr., F. J. 1951, Journal of the American Statistical Association, 46, 68, doi: 10.1080/01621459.1951.10500769

Matarrese, S., Verde, L., & Heavens, A. F. 1997, Monthly Notices of the Royal Astronomical Society, 290, 651, doi: 10.1093/mnras/290.4.651

Matilla, J. M. Z., Sharma, M., Hsu, D., & Haiman, Z. 2020, Physical Review D, 102, doi: 10.1103/physrevd.102.123506

Mesinger, A., & Furlanetto, S. 2007, The Astrophysical Journal, 669, 663, doi: 10.1086/521806

Mesinger, A., Furlanetto, S., & Cen, R. 2011, Monthly Notices of the Royal Astronomical Society, 411, 955, doi: 10.1111/j.1365-2966.2010.17731.x

Miller, B. K., Cole, A., Forré, P., Louppe, G., & Weniger, C. 2021a, in Advances in Neural Information Processing Systems, ed. M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, & J. W. Vaughan, Vol. 34 (Curran Associates, Inc.), 129–143. https://proceedings.neurips.cc/paper_files/paper/2021/file/01632f7b7a127233fa1188bd6c2e42e1-Paper.pdf

Miller, B. K., Cole, A., Forré, P., Louppe, G., & Weniger, C. 2021b, Truncated Marginal Neural Ratio Estimation, doi: 10.5281/zenodo.5043706

Modi, C., Lanusse, F., & Seljak, U. 2020, FlowPM: Distributed TensorFlow Implementation of the FastPM Cosmological N-body Solver, arXiv, doi: 10.48550/arXiv.2010.11847

Modi, C., & Philcox, O. H. E. 2023, Hybrid SBI or How I Learned to Stop Worrying and Learn the Likelihood. https://arxiv.org/abs/2309.10270

Mudur, N., Cuesta-Lazaro, C., & Finkbeiner, D. P. 2025, The Astrophysical Journal, 978, 64, doi: 10.3847/1538-4357/ad8bc3

Mukhanov, V. 2005, Physical Foundations of Cosmology (Cambridge: Cambridge Univ. Press), doi: 10.1017/CBO9780511790553

Murray, S. 2014, HMF: Halo Mass Function calculator, Astrophysics Source Code Library, record ascl:1412.006. http://ascl.net/1412.006

Murray, S. G., Power, C., & Robotham, A. S. G. 2013, Astronomy and Computing, 3, 23, doi: 10.1016/j.ascom.2013.11.001

Naidoo, K., Massara, E., & Lahav, O. 2022, Monthly Notices of the Royal Astronomical Society, 513, 3596, doi: 10.1093/mnras/stac1138

Naidoo, K., Whiteway, L., Massara, E., et al. 2019, Monthly Notices of the Royal Astronomical Society, 491, 1709, doi: 10.1093/mnras/stz3075

Ni, Y., Genel, S., Anglés-Alcázar, D., et al. 2023, The CAMELS project: Expanding the galaxy formation model space with new ASTRID and 28-parameter TNG and SIMBA suites, arXiv, doi: 10.48550/arXiv.2304.02096

Nowozin, S., Cseke, B., & Tomioka, R. 2016, arXiv e-prints, arXiv:1606.00709, doi: 10.48550/arXiv.1606.00709

Oord, A. v. d., Li, Y., & Vinyals, O. 2019, Representation Learning with Contrastive Predictive Coding, arXiv, doi: 10.48550/arXiv.1807.03748

Pan, S., Liu, M., Forero-Romero, J., et al. 2020, Cosmological parameter estimation from large-scale structure deep learning. https://arxiv.org/abs/1908.10590

Pandey, S., Modi, C., Wandelt, B. D., et al. 2024, CHARM: Creating Halos with Auto-Regressive Multi-stage networks, arXiv, doi: 10.48550/arXiv.2409.09124

Papamakarios, G., Pavlakou, T., & Murray, I. 2017, in Advances in Neural Information Processing Systems, ed. I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett, Vol. 30 (Curran Associates, Inc.). https://proceedings.neurips.cc/paper/2017/file/6c1da886822c67822bcf3679d04369fa-Paper.pdf

Papamakarios, G., Sterratt, D., & Murray, I. 2019a, in Proceedings of Machine Learning Research, Vol. 89, Proceedings of the Twenty-Second International Conference on Artificial Intelligence and

Statistics, ed. K. Chaudhuri & M. Sugiyama (PMLR), 837–848. https://proceedings.mlr.press/v89/papamakarios19a.html

Papamakarios, G., Sterratt, D. C., & Murray, I. 2019b, Sequential Neural Likelihood: Fast Likelihood-free Inference with Autoregressive Flows. https://arxiv.org/abs/1805.07226

Peebles, P. J. E. 1980, The large-scale structure of the universe

Peel, A., Lin, C.-A., Lanusse, F., et al. 2017, Astronomy & Astrophysics, 599, A79, doi: 10.1051/0004-6361/201629928

Petri, A., Haiman, Z., Hui, L., May, M., & Kratochvil, J. M. 2013, Phys. Rev. D, 88, 123002, doi: 10.1103/PhysRevD.88.123002

Phan, D., Pradhan, N., & Jankowiak, M. 2019a, Composable Effects for Flexible and Accelerated Probabilistic Programming in NumPyro. https://arxiv.org/abs/1912.11554

—. 2019b, arXiv preprint arXiv:1912.11554

Philcox, O. H., & Ivanov, M. M. 2022, Physical Review D, 105, doi: 10.1103/physrevd.105.043517

Planck Collaboration, Aghanim, N., Akrami, Y., et al. 2020, Astronomy and Astrophysics, 641, A6, doi: 10.1051/0004-6361/201833910

—. 2021, Astronomy and Astrophysics, 652, C4, doi: 10.1051/0004-6361/201833910e

Poole, B., Ozair, S., van den Oord, A., Alemi, A. A., & Tucker, G. 2019, arXiv e-prints, arXiv:1905.06922, doi: 10.48550/arXiv.1905.06922

Porqueres, N., Heavens, A., Mortlock, D., & Lavaux, G. 2021a, Monthly Notices of the Royal Astronomical Society, 502, 3035, doi: 10.1093/mnras/stab204

—. 2021b, Monthly Notices of the Royal Astronomical Society, 509, 3194–3202, doi: 10.1093/mnras/stab3234

Porqueres, N., Heavens, A., Mortlock, D., Lavaux, G., & Makinen, T. L. 2023, Field-level inference of cosmic shear with intrinsic alignments and baryons. https://arxiv.org/abs/2304.04785

Potter, D., Stadel, J., & Teyssier, R. 2016, PKDGRAV3: Beyond Trillion Particle Cosmological Simulations for the Next Era of Galaxy Surveys, arXiv, doi: 10.48550/arXiv.1609.08621

Prangle, D., Fearnhead, P., Cox, M. P., Biggs, P. J., & French, N. P. 2014, Statistical Applications in Genetics and Molecular Biology, 13, 67, doi: 10.1515/sagmb-2013-0012

Prelogović, D., Mesinger, A., Murray, S., Fiameni, G., & Gillet, N. 2021, Machine learning galaxy properties from 21 cm lightcones: impact of network architectures and signal contamination. https://arxiv.org/abs/2107.00018

Press, W. H., & Schechter, P. 1974, , 187, 425, doi: 10.1086/152650

Prince, S. J. 2023, Understanding Deep Learning (The MIT Press). http://udlbook.com

Pritchard, J. R., & Loeb, A. 2012, Reports on Progress in Physics, 75, 086901, doi: 10.1088/0034-4885/75/8/086901

Qi, X., Zhou, S., & Plummer, M. 2022, BMC Bioinformatics, 23, 102, doi: 10.1186/s12859-021-04496-8

Ramachandran, P., Zoph, B., & Le, Q. V. 2017, Searching for Activation Functions. https://arxiv.org/abs/1710.05941

Ramanah, D. K., Lavaux, G., Jasche, J., & Wandelt, B. D. 2019, Astronomy and Astrophysics, 621, A69, doi: 10.1051/0004-6361/201834117

Rao, C. R. 1945, Bulletin of the Calcutta Mathematical Society, 37, 81–89

Ravanbakhsh, S., Oliva, J., Fromenteau, S., et al. 2017, Estimating Cosmological Parameters from the Dark Matter Distribution. https://arxiv.org/abs/1711.02033

Reed, D. S., Bower, R., Frenk, C. S., Jenkins, A., & Theuns, T. 2006, Monthly Notices of the Royal Astronomical Society, 374, 2, doi: 10.1111/j.1365-2966.2006.11204.x

Ribli, D., Pataki, B. A., Zorrilla Matilla, J. M., et al. 2019, Monthly Notices of the Royal Astronomical Society, 490, 1843–1860, doi: 10.1093/mnras/stz2610

Ribli, D., Ármin Pataki, B., & Csabai, I. 2018, An improved cosmological parameter inference scheme motivated by deep learning. https://arxiv.org/abs/1806.05995

Riess, A. G., Yuan, W., Macri, L. M., et al. 2022, , 934, L7, doi: 10.3847/2041-8213/ac5c5b

Riley, K. F., Hobson, M. P., & Bence, S. J. 2006, Mathematical Methods for Physics and Engineering: A Comprehensive Guide, 3rd edn. (Cambridge University Press)

Robbins, H., & Monro, S. 1951, The Annals of Mathematical Statistics, 22, 400, doi: 10.1214/aoms/1177729586

Robert, C., & Casella, G. 2011, Statistical Science, 26, doi: 10.1214/10-STS351

Rogers, J. G., Janó Muñoz, C., Owen, J. E., & Makinen, T. Lucas. 2023, Monthly Notices of the Royal Astronomical Society, 519, 6028, doi: 10.1093/mnras/stad089

Ryden, B. 2003, Introduction to cosmology

Satorras, V. G., Hoogeboom, E., Fuchs, F. B., Posner, I., & Welling, M. 2021, E(n) Equivariant Normalizing Flows, arXiv, doi: 10.48550/ARXIV.2105.09016

Schapire, R. 2018, Lecture notes in Statistical Learning Theory, Princeton University (internal). https://www.cs.princeton.edu/courses/archive/spring19/cos511/schedule.html

Scoccimarro, R., Couchman, H. M. P., & Frieman, J. A. 1999, The Astrophysical Journal, 517, 531, doi: 10.1086/307220

Secco, L. F., Samuroff, S., Krause, E., et al. 2022, , 105, 023515, doi: 10.1103/PhysRevD.105.023515

Sellentin, E., & Heavens, A. F. 2017, Monthly Notices of the Royal Astronomical Society, 473, 2355, doi: 10.1093/mnras/stx2491

Sellentin, E., Loureiro, A., Whiteway, L., et al. 2023, The Open Journal of Astrophysics, 6, 31, doi: 10.21105/astro.2305.16134

Seo, H.-J., Sato, M., Dodelson, S., Jain, B., & Takada, M. 2010, The Astrophysical Journal Letters, 729, L11, doi: 10.1088/2041-8205/729/1/L11

Shalev-Shwartz, S., & Ben-David, S. 2014, Understanding Machine Learning: From Theory to Algorithms (Cambridge University Press), doi: 10.1017/CBO9781107298019

Sharma, D., Dai, B., & Seljak, U. 2024, A comparative study of cosmological constraints from weak lensing using Convolutional Neural Networks. https://arxiv.org/abs/2403.03490

Sheldon, E. S., & Huff, E. M. 2017, The Astrophysical Journal, 841, 24, doi: 10.3847/1538-4357/aa704b

Simpson, F., Harnois-Déraps, J., Heymans, C., et al. 2015, Monthly Notices of the Royal Astronomical Society, 456, 278–285, doi: 10.1093/mnras/stv2474

Sisson, S. A., Fan, Y., & Beaumont, M., eds. 2018, Handbook of Approximate Bayesian Computation (New York: Chapman and Hall/CRC), doi: 10.1201/9781315117195

Soelch, M., Akhundov, A., van der Smagt, P., & Bayer, J. 2019, On Deep Set Learning and the Choice of Aggregations (Springer International Publishing), 444–457, doi: 10.1007/978-3-030-30487-4_35

Song, Y., Sohl-Dickstein, J., Kingma, D. P., et al. 2021, Score-Based Generative Modeling through Stochastic Differential Equations, arXiv, doi: 10.48550/arXiv.2011.13456

Spergel, D., Gehrels, N., Baltay, C., et al. 2015, arXiv e-prints, arXiv:1503.03757, doi: 10.48550/arXiv.1503.03757

Sui, C., Zhao, X., Jing, T., & Mao, Y. 2023, arXiv e-prints, arXiv:2307.04994, doi: 10.48550/arXiv.2307.04994

Sutter, P. M., Lavaux, G., Wandelt, B. D., & Weinberg, D. H. 2012, , 761, 44, doi: 10.1088/0004-637X/761/1/44

Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. 2016, Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. https://arxiv.org/abs/1602.07261

Tavaré, S., Balding, D., Griffiths, R., & Donnelly, P. 1997, Genetics, 145, 505—518. https://europepmc.org/articles/PMC1207814

Tegmark, M., Taylor, A. N., & Heavens, A. F. 1997, The Astrophysical Journal, 480, 22–35, doi: 10.1086/303939

Teyssier, R. 2002, Astronomy & Astrophysics, 385, 337, doi: 10.1051/0004-6361:20011817

Trotta, R. 2017, ArXiv e-prints. https://arxiv.org/abs/1701.01467

Tsaprazi, E., Jasche, J., Lavaux, G., & Leclercq, F. 2023, arXiv e-prints, arXiv:2301.03581, doi: 10.48550/arXiv.2301.03581

Ueda, H., & Itoh, M. 1997, Publications of the Astronomical Society of Japan, 49, 131, doi: 10.1093/pasj/49.2.131

Vaart, A. W. v. d. 1998, Asymptotic Statistics, Cambridge Series in Statistical and Probabilistic Mathematics (Cambridge University Press), doi: 10.1017/CBO9780511802256

van de Weygaert, R., Jones, B. J., & Martínez, V. J. 1992, Physics Letters A, 169, 145, doi: https://doi.org/10.1016/0375-9601(92)90584-9

Veličković, P., Cucurull, G., Casanova, A., et al. 2018, Graph Attention Networks. https://arxiv.org/abs/1710.10903

Verde, L., Wang, L., Heavens, A. F., & Kamionkowski, M. 2000, Monthly Notices of the Royal Astronomical Society, 313, 141, doi: 10.1046/j.1365-8711.2000.03191.x

Villaescusa-Navarro, F., Wandelt, B. D., Anglés-Alcázar, D., et al. 2020a, Neural networks as optimal estimators to marginalize over baryonic effects. https://arxiv.org/abs/2011.05992

Villaescusa-Navarro, F., Hahn, C., Massara, E., et al. 2020b, The Astrophysical Journal Supplement Series, 250, 2, doi: 10.3847/1538-4365/ab9d82

Villaescusa-Navarro, F., Anglés-Alcázar, D., Genel, S., et al. 2021, The Astrophysical Journal, 915, 71, doi: 10.3847/1538-4357/abf7ba

Villanueva-Domingo, P., & Villaescusa-Navarro, F. 2022, The Astrophysical Journal, 937, 115, doi: 10.3847/1538-4357/ac8930

von Wietersheim-Kramsta, M., Lin, K., Tessore, N., et al. 2024, KiDS-SBI: Simulation-Based Inference Analysis of KiDS-1000 Cosmic Shear. https://arxiv.org/abs/2404.15402

Wagstaff, E., Fuchs, F. B., Engelcke, M., Posner, I., & Osborne, M. 2019, On the Limitations of Representing Functions on Sets. https://arxiv.org/abs/1901.09006

Wandelt, B. D. 2022, private communication

White, M. 2014, Monthly Notices of the Royal Astronomical Society, 439, 3630, doi: 10.1093/mnras/stu209

Winkler, C., Worrall, D., Hoogeboom, E., & Welling, M. 2023, Learning Likelihoods with Conditional Normalizing Flows, arXiv, doi: 10.48550/arXiv.1912.00042

Xu, B., Wang, N., Chen, T., & Li, M. 2015, Empirical Evaluation of Rectified Activations in Convolutional Network. https://arxiv.org/abs/1505.00853

Xu, K., Hu, W., Leskovec, J., & Jegelka, S. 2019a, How Powerful are Graph Neural Networks? https://arxiv.org/abs/1810.00826

Xu, X., Cisewski-Kehe, J., Green, S., & Nagai, D. 2019b, Astronomy and Computing, 27, 34, doi: 10.1016/j.ascom.2019.02.003

Yang, D., & Yu, H.-B. 2022, A graph model for the clustering of dark matter halos, arXiv, doi: 10.48550/ARXIV.2206.05578

Yang, L., Zhang, Z., Song, Y., et al. 2024, Diffusion Models: A Comprehensive Survey of Methods and Applications, arXiv, doi: 10.48550/arXiv.2209.00796

Zaheer, M., Kottur, S., Ravanbakhsh, S., et al. 2018, Deep Sets. https://arxiv.org/abs/1703.06114

Zel'Dovich, Y. B. 1970, Astronomy and Astrophysics, 500, 13

Zhou, J., Cui, G., Hu, S., et al. 2020, AI Open, 1, 57, doi: https://doi.org/10.1016/j.aiopen.2021.01.001

Zürcher, D., Fluri, J., Sgier, R., et al. 2022, Monthly Notices of the Royal Astronomical Society, 511, 2075, doi: 10.1093/mnras/stac078