Machine Learning applied to threat detection and galaxy formation

MAKUN SINGH MADAR

Machine learning applied to threat detection and galaxy formation

Makun Madar

A thesis presented for the degree of
Doctor of Philosophy

Institute for Computational Cosmology (ICC)
Department of Physics
The University of Durham
United Kingdom
16th May 2024

Machine learning applied to threat detection and galaxy formation

Makun Madar

Abstract

We present findings from several projects across the industrial and academic domains, with a common thread of using machine learning techniques. This work sees a collaboration effort between the security, medical and nuclear imaging company Kromek Group PLC, and the Institute of Computational Cosmology at Durham University. Within the three industrial projects, we present novel threat detection techniques in security and medical imaging. My research and development work on these projects has been utilised and implemented into products in specific fields. The academic project has a global aim concerning predictions for the recently launched *Euclid* satellite.

Kromek related - redacted

The academic phase of this thesis sees the prediction of Hα number counts and clustering bias for calibration of the *Euclid* survey using updated observations. We compare two approaches to the generation of number counts and luminosity functions using the `GALFORM` semi-analytical galaxy formation model, the full lightcone method and the less computationally intensive interpolative method. We find significant improvements in computational speed using the interpolative method over the lightcone method, without sacrificing accuracy. We then take our findings to generate 3000 `GALFORM` models to use to train a machine learning algorithm (again an artificial neural network) to create an emulator. We then use this emulator in an MCMC parameter search of an eleven-dimensional parameter space, to find a best-fitting model to a calibration dataset that includes local data, and, for the first time, higher redshift data, namely to the number counts of Hα emitters relevant to the *Euclid* mission.

Supervisors: Prof. Carlton Baugh and Dr Difu Shi

Acknowledgements

I would like to express my sincere appreciation to Prof. Carlton Baugh and Dr Difu Shi (Steve). Thank you for your abiding patience and support, along with your invaluable insight and feedback over the past four years. Although there was a lot I struggled with, I hope my perseverance has not made you doubt taking me on. I promise I didn't apply just to eat my weight in free food. To Steve, I wish you a long and happy marriage.

I would also like to express gratitude to the Kromek team, specifically Pete, Ra'ad, Dave, Rob and Tom. Thank you for your positive energy despite spending most of our time together in lockdown, you all made me feel like a well-valued team member. Thank you Ra'ad for your incredible assistance with my thesis write-up where you took time out of your busy schedule to talk me through improvements. I wish you all the best in future projects.

I would like to give a special thanks to Ed whose prior work on neural networks and emulators provided critical knowledge for the development of my work. A large portion of my work comes from imitating your research and I hope to follow in your footsteps.

An important member of the department to whom everyone should give thanks is Shufei, who continuously goes above and beyond to make sure everyone's daily experience is as good as it can be. You are a great friend to all of us and I wish your family the very best. Also, thank you for letting me eat all the leftover food.

Thank you to the ICC for providing ample opportunities during my research period. Thank you to COSMA support team for maintaining the essential electronic infrastructure and responding to my queries, I apologise for clogging up the COSMA5 queue with my 3000 `GALFORM` runs.

Of course, I would be remiss in not thanking my family members who have provided unconditional love and support throughout my whole life, none more so than the last four years. To my mother and my sister, there were times when I didn't believe in myself but I am so grateful to have you there believing for me. Extra thanks to my mother whose spare room I locked myself in for months to finish this thesis.

Finally, I would like to thank the amazing people I have met and had the privilege of calling my friends during my time at Durham. To my friends in the department, Zoe and Sarah (room 101), Suttikoon, and Emmy (to name just a few) you have made me look forward to walking into the building every day. Beni and Tilly, I am excited to see what the next chapter for you two holds.

To Mark and Hannah (and Honey), you have both played a very important part in establishing Durham as my home, and I am honoured that we have become so close. Thank you for your friendship and support, especially during lockdowns. I promise we won't become strangers.

Thank you to the Durham athletics group, specifically Max, Julie, Jacqui, Conrad and Samantha. You've all had to deal with me coming to training moody, tired, and stressed (not to mention injured a lot). I hope I didn't confuse you all too much when talking about my work.

The most special person I met while pursuing my PhD was Kajal, you have been kind, patient, and supportive to me from the moment we met in Newcastle. Thank you for everything, and I'm excited for our future together.

Contents

Declaration

The work in this thesis is based on research carried out at the Institute for Computational Cosmology, Department of Physics, University of Durham, England. No part of this thesis has been submitted elsewhere for any other degree or qualification, and it is the sole work of the author unless referenced to the contrary in the text.

The work presented in the first part of the thesis is subject to a non-disclouse agreement.

The material in the second half of the thesis will form the basis of a first author paper (Madar et al.), which we expect to submit shortly.

List of Figures

List of Tables

Nomenclature

AI	artificial intelligence
ML	machine learning
STFC	Science and Technology Facilities Council
DM	dark matter
ELG	emission line galaxy
BAO	Baryonic Acoustic Oscillations
GC	galaxy clustering
SFR	star formation rate
LSS	large-scale structure
HST	Hubble Space Telescope
MCMC	Monte Carlo Markov Chain
DL	deep learning
SAM	semi-analytical model
LF	luminosity function
CASE	Collaborative Awards in Science and Engineering

Preface

The thesis presented here represents the culmination of several years of dedicated research and exploration in the fields of physics and astronomy. Undertaking a doctoral journey is a profound intellectual and personal endeavour, and this document is the manifestation of countless hours of inquiry, reflection and collaboration.

The PhD was conducted under the auspices of the Science and Technology Facilities Council (STFC) Collaborative Awards in Science and Engineering (CASE) studentship. This studentship program uniquely integrates industrial and academic perspectives by collaborating with non-academic partners on projects that fall within the STFC core science program (such as astronomy) and aims to apply technologies or techniques developed within the industrial settings into the academic environment, or vice versa. In my case, the industrial partner was Kromek Group PLC, a key player in the nuclear, security and medical imaging solutions sectors. The central theme of this research lies in the utilization of machine learning and intelligent algorithm methodologies to address challenges and opportunities in both academic and industrial contexts.

The STFC CASE studentship was structured to encompass two distinct yet somewhat interconnected phases, splitting the duration of my PhD in half. Starting with the industrial phase, I immersed myself in the practical implementation of machine learning for computer vision tasks to address specific challenges identified by Kromek, collaborating with their artificial intelligence (AI) team. Due to the nature of industry deadlines, I was involved in multiple different projects during my placement at Kromek, however, I was unable to see a whole project from start to finish due to my time constraints. Transitioning to the academic phase, the focus shifted towards the applications of machine learning in the field of computational cosmology. This phase utilised the knowledge gained during my industrial phase while also exposing me to new concepts and practices I was unable to obtain within an industrial setting. The collaboration with Kromek not only enriched the research process with industry insights but also contributed to the development of machine learning solutions with immediate practical impact. The academic project exploited the expertise I had gained at Kromek in the design and application of machine learning methods. Here my main contributions were to devise an emulator of a complex scientific simulation code, `GALFORM`, and to use this in an extensive exploration of the model parameter space.

In presenting his work, I hope that it will inspire further inquiry and innovation in the intersection of academia and industry. I am immensely grateful for the opportunity to engage in this unique experience, and I look forward to the continued exploration of machine learning in both

academic and practical realms.

This thesis is split into two halves, with the first half presenting the work from my time at Kromek, and the second half presenting the academic work with Durham University and the ICC.

Machine learning background

We live in a data-rich age, where large portions of our lives are played out digitally. It is no surprise that the amount of data generated is increasing day by day. According to the latest estimates, in the year 2023, on average 328.77 million terabytes (328.77×10^9 Gigabytes) of data were created each day (Statista, 2023) (where the word *created* refers to newly generated, captured, copied or consumed data). This number has grown year-on-year since 2010 such that 90% of the world's data was generated in the last two years. In the year 2010 just 2 zettabytes (2×10^{12} Gigabytes) of data was generated, in 2023 this number rose $60\times$ to 120 zettabytes and this yearly amount is expected to increase by over 150% in the year 2025. Therefore, there is an urgent need to extract useful knowledge and insights from this data. This has led to the rise of AI, one of the key technologies of the Fourth Industrial Revolution (or Industry 4.0) (Ross & Maynard, 2021), growing rapidly in recent years in the context of data analysis and computing. Another factor in the rise of AI is computing power. Famously, Moore's Law states that the number of transistors in computer chips doubles approximately every two years (Moore, 1965). Current data suggests this law is still holding up today, and on top of this the energy use of computers has halved every 1.5 years over the last 60 years, and the price of computer memory and storage is falling every year. Coupling the advances in computing with the amount of data generated means AI has become very widespread, including in research.

It has become common to use the terms AI, machine learning (ML) and deep learning (DL) interchangeably but there are key differences that distinguish these terms. Starting with AI, this is a loose term that describes any intelligent program which aims at imitating human behaviour and cognitive functions in problem-solving and learning. There is no generally accepted definition of the concept of AI (Russell & Norvig, 2010). ML, meanwhile, is a subset of AI which uses algorithms trained on data to learn and enhance solutions of a task automatically, i.e. without specific knowledge of the underlying model behind the data being analysed. Samuel (1959) states that ML algorithms enable computers to learn from data, and even improve themselves, without being explicitly programmed. Among the different ML algorithms, DL is a subset of ML that uses deeper, more complex algorithms to recognize patterns in data. In today's world, when AI is referenced it is often specifically referring to DL.

The first manually operated, general-purpose computer system was the Electronic Numerical Integrator and Computer (ENIAC) in 1946 (Goldstine & Goldstine, 1996). This was the spark for a revolution in computer science that has led to the state of intelligent algorithms today. The term

"machine learning" was coined by Samuel (1959) while programming a computer to play a better game of checkers than the person who wrote the program. The origin of ML in the modern sense is usually associated with Frank Rosenblatt, who developed algorithms based on the work of the human nervous system. He created a machine for recognizing the letters of the alphabet called the perceptron in 1957 (Rosenblatt, 1958). It became the prototype for modern artificial neural networks (ANN).

We can divide ML and DL into three forms: supervised, unsupervised, and reinforcement learning. In supervised learning, a program is fed labelled data and an algorithm is trained on the labelled dataset and then tested to see if it can correctly apply the labels to new data (Sheikh et al., 2023). This type of learning is the major variant we will employ throughout this thesis. Unsupervised learning has no training step. The algorithm is trained to search for patterns within the data by itself, such as clusters of characteristics in the training data that will form clusters in the future. Supervised learning is ideal when researchers know what is being searched for. If, however, it is not known what patterns are hiding within the data and we want to uncover hidden connections, then unsupervised learning is more appropriate. Reinforcement learning is an approach whereby an algorithm is trained by being rewarded for following certain strategies and decisions (Arulkumaran et al., 2017). As mentioned, we will focus on supervised learning in this thesis. Within supervised learning, there are two main classes of ML and DL algorithms: classification and regression. In short, the most significant difference between regression and classification is that regression predicts a continuous quantity/ set of quantities based on an input, while classification predicts discrete class labels. The main goal of a regression model is to estimate a mapping function based on the labelled input and output datasets provided during the training phase. Classification is a predictive model that also approximates a mapping function from a set of input variables to identify discrete output variables, which can be labels or categories. We will explore specific classification algorithms for the Kromek section of this thesis in (*Kromek related - redacted*), and regression models will be the main focus for the academic part of this thesis in chapter 8.

DL is not without its challenges. For starters, it is extremely data-hungry, demanding extensively large amounts of data to achieve a well-behaved performance model. A general rule of thumb is an increase in data relates to a more reliable performance model. However, not every situation has the benefit of large data sets available, and sometimes there is a shortage of data. In these cases augmentation could be employed to artificially scale up the data sets available, for example for image processing tasks, or fine-tuning existing algorithms. The data available may be imbalanced where the data used for training a model is heavily biased to one type. For example, in some sequential biological data, there may be an abundance of one type of sample over another. In this scenario, down-sampling the larger classes or up-sampling the smaller classes would help, or using a relevant loss evaluation (the definition of loss can be found in *Kromek related - redacted*). Another problem that arises from the more complex DL algorithms is a vast number of parameters can lead to overfitting. This is when a model is too specific to the data it has trained on and fails on data it has not seen before, therefore it needs to be more general. This can also come from insufficient training. There are remedies for this e.g. Zhang & Sabuncu (2018); Xu et al. (2019, 2020); Sharma et al. (2020).

It is common to think of ML and DL techniques as black boxes that are difficult to interpret. However, they are interpretable. However, there are examples of interpreting models to obtain

patterns recognized by the model, for example in bioinformatics (Li et al., 2019). For reviews on DL, including its challenges see Alzubaidi et al. (2021); Sheikh et al. (2023).

In today's world, we encounter all kinds of applications of AI in our everyday lives, for predictive analysis, image processing, language and speech, and for the performance of physical tasks. In this thesis, predictive analysis and image processing are the most prominent applications. The ability to use data to make better-informed estimates about the future has huge potential in a vast array of commercial and research areas. Google's DeepMind has developed an AI system that takes climate forecasts and data to predict the inflow of energy from wind farms 36 hours in advance (DeepMind, 2019). Computer vision relates to image-based recognition, automating the observation, analysis and interpretation of visual information. Examples of where this is deployed are in clinical medicine in the form of image-based diagnostics, with research finding evidence that DL systems perform similarly to that of a human medical professional (Yu et al., 2018; Liu et al., 2019). It is no surprise that ML techniques are used in astronomy both for computer vision and predictive analysis. Baron (2019) overviews the practicalities of ML in astronomy including specific algorithms and applications.

In this thesis, I will present a variety of machine-learning concepts and show how they can be applied in both industrial and research tasks.

Part I

Kromek industrial placement

Chapter 1

Kromek related - redacted

Chapter 2

Kromek related - redacted

Chapter 3

Kromek related - redacted

Chapter 4

Kromek related - redacted

Chapter 5

Kromek related - redacted

Part II

Academic work

Chapter 6

Introduction

The science goal of my academic research project is to predict the abundance and clustering of the galaxies that will be mapped in the redshift survey to be carried out by the European Space Agency's (ESA) *Euclid* mission. These quantities are the starting point for forecasting the accuracy with which the various cosmological probes will be able to constrain the dark energy equation of state. Even though *Euclid* has now launched and the main survey has started, it may still be necessary to change the survey strategy. The predictions presented here will allow the impact of any such changes on the mission objectives to be quickly assessed.

This chapter sets the scene for this part of the thesis and introduces a range of concepts. The chapter is split up into the following parts: we discuss the scientific background and characteristics of *Euclid* mission in §6.1, as well as presenting some background on the Hα emission line in §6.2; in §6.3 we give an overview of the `GALFORM` semi-analytical galaxy formation model and the N-body simulation that we use throughout this part of the thesis, and in §6.4 we present the currently available observations of the abundance of Hα galaxies over the redshift window that is relevant to *Euclid*.

6.1 ESA's *Euclid* mission: the emission line galaxy redshift survey

Euclid (Laureijs et al., 2011; Racca et al., 2016) is a mission designed to understand the origin of the Universe's accelerating expansion by surveying just over 13000 deg^2 of the sky. Launched in July 2023, the payload consists of a 1.2m Korsch telescope that directs light to two instruments, the visual instrument (VIS) camera and the near-infrared instrument (NISP) which contains a slitless spectrometer, both with a field of view of $\sim$0.5 deg^2. *Euclid* will perform two main wide-field surveys with these instruments. These surveys will be used for a variety of cosmological probes. There will also be a deep survey of around 50 sq degrees which will have a significant legacy value.

The VIS instrument will image the sky in a wide filter, providing shape measurements of the accuracy previously obtained with the Hubble Space Telescope but for more than two billion galaxies. These shape measurements will allow weak gravitational lensing to be measured, which depends on the clustering in the dark matter distribution. When combined with photometric redshift estimates, the clustering in the matter distribution can be measured in a series of slices (tomography); the evolution of the matter fluctuations between these shells depends on the dark

energy. This cosmological probe potentially has the most constraining power on the dark energy equation of state, but will not be discussed further in this thesis.

The slitless spectrometer is predominantly sensitive to Hα ELGs over the redshift range redshift $0.9 \leq z \leq 1.82$, with the aim of reaching a flux limit of $2{\times}10^{-16}$ erg s^{-1} cm^{-2} (Euclid Collaboration et al., 2022). NISP should be able to determine spectroscopic redshifts for at least 1700 galaxies deg^{-2} on average according to Pozzetti et al. (2016). (This is one of the numbers that we will revisit in this part of the thesis.) There is a compromise to be made in terms of going deeper and potentially increasing the surface density of galaxies, but at the expense of a likely reduction in the redshift measurement success rate, due to the confusion in associating overlapping spectra with galaxies. The redshift success rate is also likely to be density-dependent, which could lead to an impact on the measured clustering. To help identify spectra of galaxies, two "exposures" of each field will be made with the spectrograph, following a rotation, and reference will be made to galaxy positions in the H-band image of the same part of the sky. The spectral resolution of a spectrograph is a measure of its ability to resolve features across a wavelength range, given by $\lambda/\Delta\lambda$, the finer the resolution, the narrower the wavelength range. Given the spectral resolution of *Euclid*, of around $R \sim 300$, the emission will be measured from a combination of the Hα ($\lambda = 6563$ Å) and N[II] ($\lambda = 6584$ Å) lines; resolving these lines would require $R > 6584/21 = 313$.

A number of cosmological probes can be applied to the redshift survey to investigate the nature of dark energy, dark matter and gravity using observational tracers to measure signatures of these quantities and forces imprinted on the geometry of the universe and on the cosmic structure formation history. One of the primary cosmological probes that will be used on *Euclid*'s redshift survey is galaxy clustering (GC) using the Baryonic Acoustic Oscillations (BAO) feature (Cole et al., 2005; Eisenstein et al., 2005). BAOs arise from matter sound waves caused by a gas pressure differential in the early universe (before recombination) between the highly dense plasma collapsing due to gravitational forces, and the matter heating up causing outward pressure, leading to oscillations. The maximum distance that these sound waves can travel before matter-radiation decoupling is the horizon scale at the epoch of decoupling called the sound horizon scale. This scale has been accurately measured by the Planck cosmic microwave background (CMB) mission to around 0.3 per cent accuracy (Planck Collaboration et al., 2020). The BAO fluctuations in the matter (and galaxy) clustering are much lower amplitude than, for example, in the CMB temperature fluctuation spectra, because for the former the fluctuations are damped by the coupling of the baryons to the dark matter (through gravity). Nevertheless, the BAO scale can be measured with a large enough redshift survey, that is with bin sizes large enough to encompass the $\sim$100 Mpc BAO scale and a signal-to-noise strong enough to overcome the shot noise. The BAO acts as a standard ruler, and so its apparent size in the galaxy distribution acts to measure the distance-redshift relation, which constrains dark energy since the relation is cosmology dependent (for a review of BAOs as a standard ruler see Bassett & Hlozek 2010).

Galaxies are used as tracers of the BAO and so quantifying this scale in the three-dimensional galaxy distribution in terms of its imprint on the power spectrum (or correlation function) becomes important for probing the cosmological expansion and the growth of structure across several redshift bins over the time interval when dark energy becomes dynamically important. *Euclid* will accurately determine the relationship between the spatial distribution of galaxies and the underlying dark matter density field, known as the galaxy bias. This relationship will be determined by *Euclid*

by combining the fluctuation maps of the luminous and dark matter distributions produced using the redshift survey and weak lensing probes (e.g. Reyes et al., 2010; Bernstein & Cai, 2011). As mentioned, the weak lensing tomography from VIS and the NISP measurements of the BAO are two different cosmology probes. If individually they reach the same constraints on the cosmology, then the systematic errors are well understood and under control. The sample variance errors on these measurements are small due to the large sample size, they will be comparable to the systematic errors hence the need to minimise the systematic errors. The galaxy bias affects the signal-to-noise of the correlation function/power spectrum measurement which impacts the extraction of the BAO scale. The clustering bias is defined as the square root of the ratio of the galaxy correlation function to the correlation function of the dark matter (Kaiser, 1984). Galaxies are assumed to form within dark matter halos (White & Rees, 1978), and so galaxies can be used as a proxy in studying the location of dark matter. However, estimating a region's dark matter density by multiplying the region's galaxy abundance by a single ratio is merely a rough estimate, and the clustering ratio between galaxies and dark matter could vary with scale. We expect the scale dependence of galaxy bias to be fairly weak on the BAO scale (Angulo et al., 2008, 2014) due to the relationship between galaxies and the underlying dark matter density field tending towards linearity as the fluctuations in the density field are getting smaller.

Euclid aims to map the angular diameter distance, $D_A(z)$, and the Hubble parameter, $H(z)$, as a function of redshift. The combination of radial and transverse distance measurements depends on the assumed cosmology. Euclid will measure the BAO signal in the power spectrum of galaxies (Seo & Eisenstein, 2007) in redshift shells of volume V. The primary concern is the error on the clustering measurement in each redshift shell, encapsulated in the expression for fractional error on the power spectrum derived by Feldman et al. (1993):

$$\frac{\sigma}{P} \propto \frac{1}{\sqrt{V}} \left(1 + \frac{1}{\bar{n}P}\right), \tag{6.1}$$

where σ is the error on the power spectrum P, V is the volume of the redshift shell and $\bar{n}$ is the number density of galaxies within the shell. The main determinant of the fractional error on a clustering measurement is the volume, so maximizing the survey volume increases the number of independent k modes in the power spectrum. The $1/\sqrt{V}$ term is similar to a Poisson noise on the mode counting or a density of states argument, i.e. how many wavevectors k are sampled by the volume? However, the power spectrum is measured using a finite number of galaxies, therefore there is an associated discreteness noise known as shot noise (represented by the second term, $1/\bar{n}$, in Eqn. 6.1). Due to the flux limit of surveys, the number density of galaxies drops rapidly with increasing redshift, increasing the shot noise. Once the shot noise is comparable to the power spectrum amplitude, it is difficult to measure the signal from galaxy clustering. If $\bar{n}P \gg 1$, then the shot noise term becomes much smaller and the full benefit of adding a shell of volume V is felt in the measurements. To achieve this, a tracer is required with either a large bias factor (or equivalently a large power spectrum P) or a high number density $\bar{n}$. If $\bar{n}P \ll 1$, the clustering of galaxies in this shell contributes nothing to the statistical power of the survey due to the shot noise term dominating the error. Increasing the survey volume by adding a new shell in this regime will not contribute any new information.

Hence to determine how well *Euclid* will be able to measure the BAO (and other clustering probes) we need to predict the number of objects *Euclid* will see in the redshift survey and the

clustering bias of these objects, so we can predict the errors in the power spectrum measurement for each redshift bin. As mentioned, *Euclid* will use Hα emission line galaxies as tracers of the large-scale structure. The emission line Hα is used as its signal lies within the redshift window that *Euclid* operates. Furthermore, Hα traces the outer regions of galaxy clusters where the clustering signal is weaker, therefore, more object detections are required to offset the weaker clustering. In this thesis, we model the number counts and clustering of Hα emitters using the semi-analytic model `GALFORM` (refining the work conducted by Orsi et al. 2010 by using up-to-date Hα observations and using these to calibrate the model parameters). We wish to develop an automatic way of constraining `GALFORM` model parameters to achieve the objective of this work. It is worth noting that the calculations conducted for *Euclid* can be readily adapted for other wide-field galaxy surveys, such as the Nancy Grace Roman Space Telescope (formerly WFIRST) (Green et al., 2012; Spergel et al., 2015; Bailey et al., 2023) which, when launched, will work in synergy with Euclid's Hα redshift survey.

We discuss the challenge of making predictions for *Euclid*'s redshift survey in two chapters. First in chapter 7 we discuss techniques for mock galaxy catalogue generation where we compare the commonly used lightcone method with an interpolative method to compute the number counts and redshift distribution of emission line galaxies. Then in chapter 8 we build an ML algorithm to emulate `GALFORM` outputs in order to efficiently sample a large subset of the parameter space to find an improved model for emission line galaxy catalogue predictions. Finally in chapter 9 we discuss the impact of our academic work and look to future projects that could build on our results.

6.2 Hα emission line galaxies

Some of the most utilised and observed emission lines in astrophysics are the Balmer series lines that arise from recombination to the $n = 2$ level of hydrogen, the most abundant element in the Universe. In particular, the transition from $n = 3$ to $n = 2$ produces the Hα emission line. Hα is located at the rest-frame wavelength of $\lambda = 6563$Å, in the optical, and its emission is powered by hydrogen ionizing photons from hot, massive young stars (Faisst et al., 2019; Kewley et al., 2019). Because the stars that are hot enough to produce the ionising radiation are short-lived, the emission lines act as a probe of the current star formation activity in a galaxy. Therefore, Hα sheds light on the variability of the star formation rate (SFR) in galaxies by tracing the conditions in the ionized gas in star-forming HII regions. For galaxies at $z > 0.5$, this line is redshifted out of the optical and into the NIR section of the electromagnetic spectrum (Thompson et al., 1996; McCarthy et al., 1999; Hopkins et al., 2000; Shim et al., 2009). Hα can measure the SFR locally and out to a redshift of z$\sim$2.5. Compared to other emission lines such as Lyα, O$_{II}$, O$_{III}$, Hβ or Hγ, Hα has a longer wavelength and therefore is not as attenuated by dust, or as sensitive to metallicity and the ionization state of the gas. Hence, Hα is a more direct tracer of actively star-forming galaxies (e.g., see the review of Kennicutt Jr, 1998) and is a robust tracer choice of the large-scale structure (LSS) of the Universe. It is worth noting that dust attenuation from emission lines can be greater than the attenuation experienced by the stellar continuum; this is because emission lines can be attenuated by the gas cloud that new stars are made in as well as the diffuse dust in a galaxy.

Historically, measuring the abundance of Hα emitters by spectroscopy could only be done at low redshift through ground-based spectroscopic surveys working in the optical (e.g. Gallego

et al., 1995; Tresse et al., 2002). Even once NIR spectrographs were developed, the intense airglow at ground level made NIR spectroscopic searches for emission line galaxies at higher redshifts impractical. However, it is feasible to target Hα emitters using slitless spectroscopy onboard space-based telescopes, such as onboard *Hubble Space Telescope* (HST) or *Euclid*. *Euclid* specifically will observe a patch of the sky and take two spectra readings of objects with a flux limit of $f \geq 2\times10^{-16}$ erg s^{-1} cm^{-2}, with a rotation of the NISP spectrograph (Euclid Collaboration et al., 2022). It will then use H-band imaging to match the spectra to galaxies in the H-band image. In §6.4 we give a brief description of alternative Hα surveys (HiZELS and WISP), explaining their detection methodologies.

6.3 Theoretical Model

In this section, the semi-analytical galaxy formation model `GALFORM` is briefly introduced (§6.3.1) in conjunction with the dark matter-only N-body simulation in which it is implemented in this thesis (§6.3.2). The basic physical model of cosmology employed is the ΛCDM model that describes the primary components of the Universe as dark matter, dark energy, and baryonic matter. Within this model, galaxies form through a hierarchical process, from small density fluctuations in the early Universe that grow via gravitational instability to form dark matter halos. These halos merge, attracting baryonic matter via gravity. Galaxies are believed to form within dark matter halos as gas cools (White & Rees, 1978).

6.3.1 `GALFORM`

`GALFORM` is a physically motivated semi-analytical model for hierarchical galaxy formation (Cole et al., 2000; Bower et al., 2006; Lacey et al., 2016), modelling *ab initio* from the distribution of cold dark matter (DM) and following the evolution of the baryonic component using a set of coupled differential equations. `GALFORM` populates the DM haloes at the earliest branches of the merger tree with hot baryonic gas and models the main physical processes governing the formation and evolution of galaxies: (i) the collapse and merging of DM haloes, (ii) the shock-heating and radiative cooling of gas inside DM haloes, which leads to the formation of galactic discs, (iii) quiescent star formation in galactic discs, (iv) feedback from SNe, active galactic nuclei (AGN), and photo-ionization of the inter-galactic medium, (v) chemical enrichment of stars and gas and (iv) dynamical friction driven by mergers of galaxies within DM haloes, forming spheroids and triggering starbursts. Full descriptions of these physical processes are given in a series of papers: Cole et al. (2000); Bower et al. (2006); Lacey et al. (2016) as well as the reviews by Baugh (2006) and Benson (2010).

The star formation histories are convolved with a simple stellar population (SSP) model, which gives the light emitted as a function of age by a population of stars produced with a given stellar initial mass function and metallicity, allowing for the prediction of a composite spectral energy distribution (SED) for each galaxy (see the review by Conroy 2013). This can be sampled with various filters to predict the rest or observer frame magnitudes in the UV, optical, NIR and far-infrared bands. Dust is assumed to be mixed with the stars in the disk of a galaxy in two phases: in clouds and in a diffuse component (Granato et al., 2000). Properties of the dust are assumed and

combined with the predicted scalelengths of the disk and bulge allowing for the optical depth and attenuation of the starlight to be calculated as a function of wavelength.

GALFORM distinguishes between central and satellite galaxies within their host dark matter halo, with some of the physical processes being affected by this designation. Central galaxies are placed at the centre of most massive sub-halo and are the focus of all the gas that is undergoing cooling. Halo merger events choose the central galaxy of the main (most massive) progenitor halo as the central galaxy of the descendant halo with other galaxies becoming satellites. In the default gas cooling model (see Font et al. 2008 for an alternative model), satellite galaxies are stripped of their hot gas as soon as they become satellites, hence quenching any further cooling and stopping any long-term star formation. The satellite is assumed to follow the motion of the sub-halo. Eventually, the sub-halo may no longer be resolved within the simulation; some authors refer to this as an orphan galaxy. When this happens we calculate an analytical timescale for the satellite galaxy to merge with the central (Simha & Cole, 2017). Merger time scales are calculated based on the initial energy and angular momentum of the satellite's orbit at the point that the sub-halo is lost (these distributions are measured from N-body simulations and sampled by the model), and the mass of the satellite and halo hosting both the satellite and central galaxy. It is expected that after this time the effects of dynamical friction will have caused the satellite to merge with the central galaxy. However, the merger time scale of a satellite is calculated every time a satellite's host halo merges into a sub-halo of a more massive halo (Cole et al., 2000; Simha & Cole, 2017).

Here we give an overview of the processes in GALFORM that are relevant to this thesis.

6.3.1.1 Quiescent star formation in discs

The quiescent mode of star formation takes place in the disk following the accretion of cooled gas from the hot halo. The star formation rate (SFR) in the disk is calculated using the empirical law inferred from observations by Blitz & Rosolowsky (2006) (as implemented in GALFORM by Lagos et al. 2011; see also Fu et al. 2010 and Popping et al. 2014 for the incorporation of similar schemes into other semi-analytical models) which is based on observations of nearby star-forming disc galaxies. The SFR is assumed to be proportional to the mass of the molecular component of the gas in the disk $M_{\rm mol,disk}$

$$\psi_{\rm disk} = \nu_{\rm SF} M_{\rm mol,disk}, \tag{6.2}$$

where $\nu_{\rm SF}$ is the value of the SFR coefficient, which controls the rate of conversion of the molecular gas into stars in quiescent galaxy disks. This is an adjustable parameter set within the range inferred from observations by Bigiel et al. (2011). The mass of molecular gas depends on the gas pressure in the mid-plane of the disk.

6.3.1.2 Supernova feedback

Supernova explosions eject gas from galaxies and their host dark matter halos. These ejections are mainly due to type II supernovae from the deaths of short-lived, massive stars. The model therefore assumes the rate of gas ejection due to supernova feedback is proportional to the instantaneous SFR ψ, with a mass loading factor that is dependent on the galaxy circular velocity, $V_{\rm c}$, as a power

law:

$$\dot{M}_{\rm eject} = \left(\frac{V_{\rm c}}{V_{\rm SN}}\right)^{-\gamma_{\rm SN}} \psi, \tag{6.3}$$

where $\gamma_{\rm SN}$ and $V_{\rm SN}$ are adjustable parameters. We can further split the $V_{\rm SN}$ term into $V_{\rm SN,\,disk}$ and $V_{\rm SN,\,burst}$ to distinguish the feedback contributions in quiescent star formation in disks from star formation in bursts. Most previous studies have assumed that these two velocity normalisation parameters are equal (e.g. Gonzalez-Perez et al. 2014 and Lacey et al. 2016). However, recent versions of the model have relaxed this restriction (e.g. Benson & Bower, 2010; Elliott et al., 2021).

Gas ejected from the galaxy due to SN feedback is assumed to gather in a reservoir beyond the virial radius of the host dark matter halo. The gas gradually returns to the hot gas reservoir within the virial radius at a rate of

$$\dot{M}_{\rm return} = \alpha_{\rm ret} \frac{M_{\rm res}}{\tau_{\rm dyn,\,halo}}, \tag{6.4}$$

where $\tau_{\rm dyn,\,halo}$ is the halo dynamical time, $M_{\rm res}$ is the mass of the reservoir beyond the virial radius, and $\alpha_{\rm ret}$ is an adjustable free parameter (Bower et al., 2006).

6.3.1.3 Galaxy mergers

It is assumed when galaxies merge there may be a burst of star formation and destruction of the galactic disks. To define the types of mergers we set two thresholds, $f_{\rm ellip}$ and $f_{\rm burst}$. These thresholds are compared to the baryonic masses of the central galaxy, $M_{\rm b,cen}$, and the merging satellite galaxy, $M_{\rm b,sat}$ through the ratio $M_{\rm b,sat}/M_{\rm b,cen}$. For the case where $M_{\rm b,sat}/M_{\rm b,cen} \geq f_{\rm ellip}$, the merger is classified as a *major* merger. After a major merger, the disk component of the primary galaxy is destroyed and forms a spheroid. We assume the cold gas in the disk is used up in a burst of star formation which also adds stars to the spheroid. The case for which $M_{\rm b,sat}/M_{\rm b,cen} < f_{\rm ellip}$ is designated as a *minor* merger. Following a minor merger, the disk survives. For the cold gas in the disk to be consumed in a starburst after a minor merger, the following condition must be met, $M_{\rm b,sat}/M_{\rm b,cen} \geq f_{\rm burst}$. Both $f_{\rm ellip}$ and $f_{\rm burst}$ are treated as free parameters. We use the new prescription of Simha & Cole (2017) to compute the time for a galaxy merger to take place.

6.3.1.4 Disk instabilities

Disk instabilities can trigger star formation. When a galaxy is dominated by rotational motion the disk is unstable to bar formation through sufficient self-gravitation. We assume that disks are dynamically unstable to bar formation if the following condition is met (Efstathiou et al., 1982)

$$F_{\rm disk} \equiv \frac{V_{\rm c}(r_{\rm disk})}{(1.68 G M_{\rm disk}/r_{\rm disk})^{1/2}} < F_{\rm stab}, \tag{6.5}$$

where $M_{\rm disk}$ is the total disk mass and $r_{\rm disk}$ is the disk half-mass radius. The quantity $F_{\rm disk}$ describes the contribution of disk self-gravity to its circular velocity, with larger values equating to lower self-gravity and greater disk stability. Predictions of $F_{\rm disk}$ have produced varied results from different methods; Efstathiou et al. (1982) found $F_{\rm disk} \approx 1.1$ for a suite of exponential stellar disk models, while Christodoulou et al. (1994) found $F_{\rm disk} \approx 0.9$ for gaseous disks. For a completely self-gravitating stellar disk, $F_{\rm disk} = 0.61$. $F_{\rm stab}$ is a model parameter.

If the disk instability condition $F_{\rm disk}$<$F_{\rm stab}$ is met we assume the disk forms a bar which evolves into a spheroid (Combes et al., 1990; Debattista et al., 2006). We assume that an unstable disk is disrupted by bar instabilities on a sub-resolution timescale thus all the mass is instantly transferred to the spheroid and any gas present is used in a burst of star formation.

6.3.1.5 Starbursts

We assume the rate at which bursts of star formation form stars in a hot spheroid is

$$\psi_{\rm burst} = \nu_{\rm SF,burst} M_{\rm cold,burst} = \frac{M_{\rm cold,burst}}{\tau_{*\ \rm burst}}, \tag{6.6}$$

where the timescale $\tau_{*\ \rm burst}$ is

$$\tau_{*\ \rm burst} = \max[f_{\rm dyn}\tau_{\rm dyn,bulge}, \tau_{*\ \rm burst,min}]. \tag{6.7}$$

Here the bulge dynamical time is defined in terms of the half-mass radius and circular velocity of the bulge, $\tau_{\rm dyn,bulge}$=$r_{\rm bulge}/V_c(r_{\rm bulge})$. We treat $\tau_{*\ \rm burst,min}$ as an adjustable parameter, and we fix $f_{\rm dyn}$=20 (Lacey et al., 2016).

6.3.1.6 SMBH growth and AGN feedback

Supermassive black holes can inject energy into the halo gas disrupting the gas cooling. Multiple instances can lead to black hole growth; hot halo accretion, BH-BH mergers, and starbursts (Bower et al., 2006; Fanidakis et al., 2011; Griffin et al., 2019). In the starburst scenario, mass accreted onto the SMBH is a constant fraction of the mass of stars formed, $f_{\rm SMBH}$, where $f_{\rm SMBH}$ is an adjustable parameter. AGN heating of the hot gas halo is assumed to occur if two conditions are met: (1) the gas halo is in quasi-hydrostatic equilibrium, i.e. meeting the condition

$$\tau_{\rm cool}/\tau_{\rm ff} > 1/\alpha_{\rm cool}, \tag{6.8}$$

where $\tau_{\rm cool}$ is the gas cooling time, $\tau_{\rm ff}$ is the free-fall time, and $\alpha_{\rm cool}$ is an adjustable parameter; (2) The AGN power required to balance the radiative cooling luminosity is less than the fraction of Eddington luminosity of the SMBH, $f_{\rm Edd}$.

6.3.2 N-body simulation

When `GALFORM` is coupled with an N-body simulation, the simulation is used to provide merger histories for the dark matter halos required by the model and the semi-analytical model provides predictions for the spatial distribution of galaxies (Benson et al., 2000).

We run `GALFORM` on the Plank Millennium N-body simulation (hereafter PMILL) (Baugh et al., 2019), the latest in the 'Millennium' series of simulations carried out by the Virgo Consortium. The PMILL has a similar but 1.43× larger volume (after accounting for differences in the Hubble parameters) than the simulation described by Gao et al. (2009) (hereafter WM7). The PMILL parameters correspond to the best-fitting cold dark matter model for the first year Planck cosmic microwave background data and measurements of the large-scale structure in the galaxy

distribution (Planck Collaboration et al., 2014); these parameters have been updated in the final Planck dataset (Planck Collaboration et al., 2020). The specifications and cosmological parameters used correspond to a flat universe, and are listed here for completeness:

(i) Present day matter density, $\Omega_{\rm M}$ (in units of critical density): 0.307

(ii) Baryon density parameter, $\Omega_{\rm b}$: 0.04825

(iii) Spectral index of the primordial density fluctuations, $n_{\rm spec}$: 0.9611

(iv) Reduced Hubble parameter, $h = H_0/(100$ km s^{-1} Mpc$^{-1})$: 0.6777

(v) Normalisation of the density fluctuations at the present day, σ_8: 0.8288

(vi) Simulation box length, $L_{\rm box}(h^{-1}$ Mpc): 542.16

(vii) Number of particles, $N_{\rm p}$: 5040^3

(viii) Particle mass, $M_{\rm p}$ $(h^{-1}$ M$_\odot)$: 1.06×10^8

(ix) Halo mass limit corresponding to 20 particles, $M_{\rm p}$ $(h^{-1}$ M$_\odot)$: 2.12×10^9.

These cosmological parameters listed for PMILL are the same parameters as those used in the EAGLE simulations of Schaye et al. (2015).

The use of 128 **billion** particles (5040^3) to represent the matter distribution gives the PMILL an order of magnitude better mass resolution than the MSI or WM7 simulations. Coupling this with the volume used, PMILL has an intermediate resolution between MSI (Springel, 2005) and MSII (Boylan-Kolchin et al., 2009). The number of outputs also increases from MSI to PMILL, with PMILL storing 271 redshifts of halos and subhalos compared with only 60 used in MSI.

The friends-of-friends (FoF) percolation algorithm is first run on the PMILL output snapshots to identify structures that can be considered as virialised dark matter halos. SUBFIND (Springel et al., 2001) is then run on the FOF particles to identify subhalos, self-gravitating over-densities within the FoF halos, to construct the dark matter halo merger trees using the DHALOS algorithm described in Jiang et al. (2014) (see also Merson et al. (2013)). The (sub)halos are retained if they have a minimum particle count of 20, which corresponds to a minimum halo mass of 2.12×10^9 h^{-1}M$_\odot$. The full particle data is only stored for a selection of redshift snapshots for storage efficiency. If the full particle and halo data is stored at all 271 redshift outputs the dataset size would be 1Pb. We run GALFORM on the PMILL simulation using the COSMA-5 machine of the DiRAC installation at Durham.

6.4 A summary of existing Hα observations

As described in §6.1, *Euclid's* wide field survey will observe Hα ELGs over the redshift range of $0.9 \leq z \leq 1.82$ down to a flux limit of 2×10^{-16} erg^{-1}cm^{-2}s^{-1}. Several prior observations have complemented *Euclid's* survey from ground- and space-based satellites, which have been used to predict how many ELGs Euclid will see. We review these here to give the context for the Hα data that we will use to calibrate a new GALFORM model.

The Hi-Z Emission Line Survey (HiZELS) (Geach et al., 2008; Sobral et al., 2009, 2012, 2013) is a ground-based, narrow-band NIR survey using observations obtained with the Wide Field CAMera (WFCAM; Casali et al. 2007) on the United Kingdom Infrared Telescope (UKIRT) that surveys emission-line objects over several square degrees of the Cosmological Evolution Survey Field (COSMOS; Scoville et al. 2007), sampling Hα star-forming galaxies at $z = 0.4, 0.84, 1.47$ and 2.23 using broad- and narrow-band filters in the z', J, H and K bands. Samples of candidate Hα emitters at various redshifts are selected using a combination of broad-band and narrow-band colours (colour-colour selections) and photometric redshifts (where available). The narrow-band filters trace Hα at the specified redshifts, while the broad-band photometric imaging is used to estimate and remove the background contribution to the continuum. This selection criterion is coupled with a spectroscopic follow-up. Objects with lines with a rest-frame equivalent width (EW) of at least EW_0=25 Å are retained to guarantee a clean selection of line emitters and ensure the samples of Hα emitters are selected down to the same restframe EW allowing for evolution across cosmic time to be quantified. Additionally, spectroscopically confirmed sources are included. See Table 3 of Sobral et al. (2013) for the number of sources, including spectroscopically confirmed ones, for each field at each redshift surveyed.

Space telescopes such as the Hubble Space Telescope (HST) have also been used to measure the Hα emission line number counts and luminosity functions. Colbert et al. (2013) presented near-infrared emission line counts and luminosity functions from the *Hubble Space Telescope* Wide Field Camera 3 (WFC3) Infrared Spectroscopic Parallels (WISP) program (Atek et al., 2010, 2011) with both the G102+G141 slitless grism filters. A grism is the combination of a diffraction grating and a prism, used to separate otherwise overlapping orders of diffraction spectra and increase the resolving power. This survey selects emission line galaxies with EW greater than 40 Å for a faint flux limit of $(3-5) \times 10^{-17}$ erg^{-1}cm^{-2}s^{-1} which is deeper than the expected flux limit of *Euclid*. They present projections for *Euclid*-like cumulative number counts between the redshift range $0.7 \leq z \leq 1.5$ for a survey size of 10 000 deg^2 for an Hα flux of 2×10^{-16} erg^{-1}cm^{-2}s^{-1}. To address false detections, visual inspection and comparison from team members are used, leading to no consensus being reached for just 5% of emission line objects. These uncertain emission lines were removed from the final sample, and errors involving visual inspection were accounted for in the completeness correction derivation. Colbert et al. (2013) quotes a completeness of 90%-95% even for high signal-to-noise ratio lines (S/N $>$ 100). In total, WISP extracts 1960 emission lines with Hα being detected in 996 galaxies and 226 galaxies with an overlap of both Hα and [OIII] across their 29 fields covering 0.037 deg^2. They find their high-redshift (z=0.9-1.5) counts are in agreement with the high-redshift (z=1.47) HiZELS survey but underpredict their z=0.84 results with their low-redshift luminosity function (z=0.3-0.9). The factors explaining the discrepancy between the two surveys could include differences in completeness methods, sample selection and sample variance from the different survey sizes.

An update to the observations carried out by Colbert et al. (2013) was provided by Bagley et al. (2020) who combined the spectroscopic data from three HST grism programs: the WISP survey, 3D-HST (Brammer et al., 2012; Skelton et al., 2014; Momcheva et al., 2016), and A Grism H-Alpha SpecTroscopic Survey (AGHAST) (Weiner, 2009). All of these programs utilise the near-infrared slitless spectroscopic observations using the WFC3 IR grisms: G102 and G141. The observations compiled by Bagley et al. (2020) cover a total area of 0.56 deg^2 which is approximately equal to

Euclid's NISP field of view. The completeness corrections from Colbert et al. (2013) were adopted for the 3D-HST+AGHAST catalogues, while simulations were used to determine the completeness of the updated line-finding procedure that created the WISP catalogue. On top of this, an EW lower limit of 40 Å and S/N > 5 is used. From the full WISP+3D-HST catalogue, a wide sample of ELGs is selected to match the *Euclid*-wide survey. Sources are selected with $f \geq 2\times10^{-16}$ erg^{-1}cm^{-2}s^{-1} and $\lambda_{\rm obs} \geq 1.25$ μm, resulting in a Hα coverage from $0.9 \leq z \leq 1.6$ for the wide field. Note, that given the spectral resolution of the Euclid spectra, Hα+[N II] will be blended for most source spectra, Bagley et al. (2020) does not correct the observed Hα for the contribution from [N II]. The wide survey here consists of 3266 Hα+[N II] emitters deg^{-2} below redshift z $\leq$ 2.5 when correcting for incompleteness. The errors on their number counts come from a Monte Carlo process creating 200 realizations of the full WISP+3D-HST emission-line catalogue and measuring the distribution on the number counts.

Pozzetti et al. (2016) constructed empirical models of the Hα luminosity function spanning the range of redshifts and line luminosities relevant to the redshift surveys proposed by Euclid and other upcoming space missions such as the Nancy Grace Roman telescope. They consider various surveys but prioritise those that are within the sensitivity ranges of Euclid and Nancy Grace Roman, these being the HiZELS survey, WISP survey and HST+NICMOS survey (Shim et al., 2009). They construct three models for the Hα luminosity function evolution by applying fits to the observations, smoothing between the observed data points. It is worth emphasising that the models produced by Pozzetti et al. (2016) were not physically motivated. The first uses a Schechter (1976) parametrization for the luminosity functions and an evolutionary form similar to Geach et al. (2010). The second model adopts a similar approach with the Schechter function for the LFs but modifies the evolutionary form for $L_\star$. The final model is a combined fit to HiZELS, WISP and NICMOS using an Monte Carlo Markov Chain (MCMC) to explore the Schechter function parameter space. The fit parameters can be found in Table 2 of Pozzetti et al. (2016).

A visualisation of a selection of the observed luminosity function estimated can be found in Fig. 7.1 in the next chapter. Here we compare the latest ELG number counts from Bagley et al. (2020) with the empirical models from Pozzetti et al. (2016) in Fig. 6.1, which is extracted from the panels that make up fig. 7 from Bagley et al. (2020). The Bagley et al counts agree best with the 'pessimistic' empirical model 3 from Pozzetti et al. (2016). Covering the redshift range $0.9 < z < 1.8$ to a flux limit of 2×10^{-6} erg s^{-1} cm^{-2}, the Pozzetti et al. predictions of the number of Hα emitters/deg^2 have a large uncertainty. They quote a predicted range of 2000-4800 Hα emitters/deg^2 (which is equivalent to 30-72 million emission sources mapped by a 15,000 deg^2 *Euclid* wide survey). A new prediction is needed with a smaller uncertainty to allow more useful forecasts to be made for *Euclid*. The GALACTICUS lines in Fig. 6.1 are from a similar model to the `GALFORM` model used here. The different curves show different assumptions about dust extinction.

The most recent 'new release' of `GALFORM` was produced in Lacey et al. (2016) which extends on previous versions of `GALFORM` (e.g. Cole et al., 2000; Baugh et al., 2005; Bower et al., 2006; Font et al., 2008; Lagos et al., 2011; Gonzalez-Perez et al., 2014) using the Millennium-WMAP7 cosmological N-body simulation (Komatsu et al., 2011; Merson et al., 2013; Jiang et al., 2014). Here, the primary observational constraints for the model were: the optical and near-IR luminosity function at $z = 0$, HI mass function at $z = 0$, morphological fractions at $z = 0$, black hole - bulge

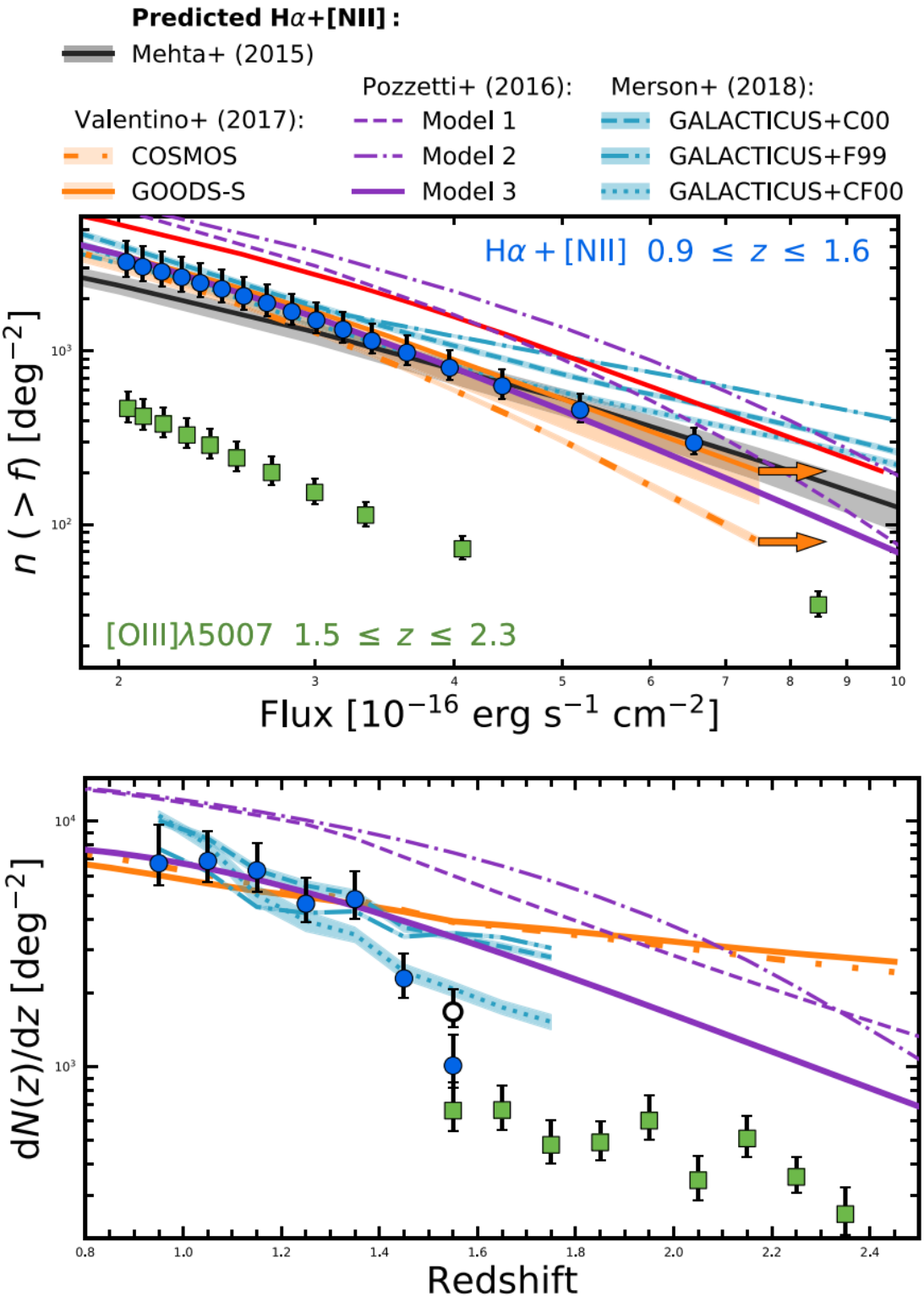


Figure 6.1: Number counts (top) and redshift distribution of ELGs, adapted from fig. 7 of Bagley et al. (2020). The blue symbols show the Hα + [NII] data from Bagely et al. The purple lines show the three empirical models from Pozzetti et al. (2016). The cyan lines show the predictions of another semi-analytical model, with the different lines showing different assumptions about dust extinction. The other lines and symbols will not be discussed.

mass relation at $z = 0$, evolution of near-IR luminosity function, sub-mm galaxy number counts and redshift distributions, far-IR number counts, and far-UV luminosity functions of Lyman-break galaxies, along with several secondary observational constraints. The best-fitting model to the constraints given was found by running small, targetted grids of models and visually comparing the results, rather than using automated procedures, such as MCMC (see e.g. Elliott et al. 2021). The parameters varied and their ranges are given in Table 1 of Lacey et al. (2016). These parameter ranges are set through a mixture of theoretical considerations and independent observational constraints. This model has great success modelling the number counts of sub-millimetre galaxies at $z \sim 1$ - 3 and the evolution of the bright end of the rest-frame K-band luminosity function of galaxies at $z \sim 0$ - 3. However, it is a generalized model and as such its parameters are not tuned to accurately recreate the number count for Hα emission line galaxies within the parameters for the *Euclid* wide-field survey, as we will see in the next chapter.

In the following chapters, we construct a new semi-analytical model for *Euclid*-like predictions, improving on the work conducted by Pozzetti et al. (2016) by using more up-to-date observational constraints from Bagley et al. (2020), and, for the first time, employing these directly in the calibration of `GALFORM`. This will allow us to make an explicit connection between the calibration data and the range of acceptable models. Furthermore, compared with empirical models which only give the abundance of ELGs, our model will also predict the clustering of ELGs and can be used to make predictions for other experiments, such as *WFIRST*.

Chapter 7

Testing methods for predicting the number counts of Hα emitting galaxies

7.1 Motivation

In the next chapter, we will perform an MCMC search over an 11-dimension parameter space, calculating the number counts of Hα emitting galaxies for each set of parameters. This exercise could involve somewhere in the order of 200,000 calculations of the counts, e.g. if we use 20 MCMC chains of 10,000 steps each in the parameter search. In model calibration exercises like the one carried out by Elliott et al. (2021), the predictions for the target datasets can be calculated from a single output redshift, using a sample of halos from the full N-body simulation. For the number counts, many output redshifts are required. Due to the way the `GALFORM` code is structured, each output redshift requires a separate run of the model, so the computational overhead is increased by a factor equal to the number of redshifts needed. `GALFORM` was first developed for single redshift research. `GALFORM` does not store the spectrum of the galaxy, instead it stores the mass-to-light ratios in a set of pre-specified bands. The definition of the mass-to-light ratio in the observer frame is redshift-dependent. At the time this choice was made on memory grounds because we expected to be looking at a few filters rather than storing the whole spectrum that could be running to thousands of numbers. Even if we use an emulator to search the model parameter space (as in the next chapter), a large number of expensive full runs of the model could be needed to train an emulator for the number counts.

In this chapter, we compare two methods for calculating the number counts of galaxies. Both methods are approximate. The first method is to construct a lightcone mock catalogue from the output snapshots of the simulation, once they have been populated with galaxies (see Merson et al. 2013). This method uses *all* of the simulation outputs within a specified redshift range. This approach is approximate because we generally do not see the full simulation volume at a given redshift in the observer's past lightcone; instead the solid angle specified cuts through a fraction of the full simulation volume. Furthermore, we do not need to run the `GALFORM` model for all of the files that make up a snapshot (see later for a discussion of how the files are organised); we can run a sparse sampling of the 'full' galaxy density and adjust for this sampling when computing

the number counts. The lightcone method could be considered as a Monte Carlo approach which makes a realisation of the number counts. The second approach involves estimating the luminosity function at a select number of redshifts, and then using this to compute the number counts. This method is approximate for two reasons: 1) we may not use the full simulation to estimate the luminosity function, which could make the estimate inaccurate for rare, bright galaxies. 2) The method uses interpolation to generate the luminosity function in between the chosen output snapshots. If the luminosity function evolves in a complicated way e.g. if it goes up and down between the available snapshots, this behaviour will not be captured by the interpolation scheme adopted. We call this second approach the interpolative method. This method is much cheaper computationally than the lightcone approach.

Here we present calculations of the number counts of Hα emitting galaxies made using both approaches. We treat the lightcone calculation as the benchmark and degrade the inputs (i.e. number of subvolumes and output redshifts used) in the interpolative method to test what is the minimum computational expense that we need to incur to make an accurate prediction.

7.2 Number counts prediction methods

Here we outline the two methods used to generate number count predictions; the lightcone and interpolation methods. We adopt the Lacey et al. (2016) version of the GALFORM semi-analytical model for galaxy formation implemented in the PMILL N-body simulation, as recalibrated by Baugh et al. (2019), and described in chapter 6.

7.2.1 Building a lightcone catalogue to obtain the number counts

We first run the GALFORM model on halo merger trees from the PMILL simulation (described in Chapter 6) to generate a galaxy population that is used to build a lightcone catalogue. GALFORM is designed so that we have to run the model for every epoch available from the N-body simulation from redshift $z = 0$ up to a specified maximum redshift. The properties of the galaxy population in the simulation box are stored at a discrete set of fixed snapshot epochs within the redshift range of interest. The lightcone is built by interpolating the galaxy magnitudes and positions between the values of all the discrete redshifts available, using the redshift at which the galaxy crosses the observer's lightcone, following the procedure described in Merson et al. (2013). We define the observer position inside the simulation box and choose a line-of-sight direction for the mid-point of the survey, and a solid angle. It is good practice to avoid a line-of-sight that does not coincide with one of the axes of the simulation box. Due to the size of the P-MILL simulation box, using this volume on its own we would only be able to probe to redshifts up to $z \approx 0.19$. Hence, to cover the volume sampled by *Euclid* we tile replicate the simulation box to fill the much larger volume needed, exploiting the periodic boundary conditions of the simulation. There are no translations or rotations, structures are preserved and repeated on box size scales and interpolated smoothly between the snapshots. Galaxies are placed according to the epoch at which they first cross the observer's lightcone, which is equivalent to the location at which the light emitted from the galaxy reaches the observer at $z = 0$. Different interpolation procedures are applied for central and satellite galaxies to minimise the artificial jumps in the correlation function measured from the

lightcone. Central galaxies are assumed to be at the centre of mass of the host dark matter halo, so tracking the motion between snapshots can be done by a simple linear interpolation. Satellite galaxies follow a more complex path, as they can enter the observer's lightcone either before or after their associated central galaxy. Therefore, a more sophisticated treatment is required to compute its position about the central (see Fig. 2 of Merson et al. (2013)).

We follow Merson et al. (2013) and Manzoni et al. (2023) for the handling of galaxy properties between the distinct snapshots using the procedure by Kitzbichler & White (2007). For any galaxy that enters the lightcone, we assign the intrinsic property it had at the snapshot immediately before the epoch, z, at which it entered the lightcone, i.e. the snapshot i with the smallest redshift, z_i, for which $z_i > z$. Using the methods described above, we build a mock catalogue which covers approximately 100 deg^2 for the redshift range $0 < z < 2.23$, which extends to the highest redshift at which we have H-α observations to compare against.

We compute the luminosity function of Hα ELGs, number counts and redshift distributions separately from the lightcone mock by simply counting the number of galaxies in a volume and creating a histogram – this can be viewed as a Monte Carlo realisation of the number counts. For the cumulative number counts and redshift distribution, we consider galaxies inside the lightcone that have a redshift of $0 < z < 2.23$ from the observer and create the respective histogram plots that trace the distributions per square degree. When calculating the number counts here, we take into account factors such as bin size, viewing radius and the total number of N-body snapshot files (or subvolumes) used when creating the lightcone mock (which is explained further in §7.3.1).

For the luminosity function, we process the galaxies in the lightcone that fall within an annulus about a desired redshift, z. Lightcone galaxy mocks output the properties of galaxies relative to the observer, therefore we must convert the Hα emission line flux into luminosity using:

$$d_{\mathrm{L}}(z) \equiv \sqrt{\frac{L}{4\pi\, S}}, \tag{7.1}$$

as the redshift of the source is known. We then count the galaxies within a thin shell between redshifts z and $z + \delta z$ and volume

$$\Omega_{\mathrm{s}} \int_{z}^{z+\delta z} \mathrm{d}z \frac{\mathrm{d}V}{\mathrm{d}z}, \tag{7.2}$$

where Ω_{s} is the solid angle of the survey from the view radius, and $\mathrm{d}V/\mathrm{d}z$ is the cosmological volume element which depends on the geometry of the model universe. Using this, we can create a luminosity function histogram at any specified redshift.

7.2.2 Using interpolation to compute the number counts

Similar to the lightcone method described above, we run `GALFORM` on the PMILL to generate a galaxy population. However, we do this at a handful of discrete redshift snapshots that cover the desired redshift range. We use a fraction of the total number of halos available in the PMILL simulation just as we do for lightcone building. The number counts of galaxies can be computed from the galaxy luminosity function, as a complementary approach to using the lightcone output.

The differential number counts of galaxies in the magnitude range m to $m + \delta m$ are given by

$$\frac{\mathrm{d}N(m)}{\mathrm{d}m} = \Omega_{\mathrm{s}} \int_{0}^{\infty} \mathrm{d}z \frac{\mathrm{d}V}{\mathrm{d}z} \int_{L_1}^{L_2} \mathrm{d}L\, \phi(L, z), \tag{7.3}$$

where $\Omega_{\rm s}$ is the solid angle of the survey, $\mathrm{d}V/\mathrm{d}z$ is the cosmological volume element, which depends upon the geometry of the universe, and $\phi(L,z)$ is the galaxy luminosity function at redshift z. The integral over luminosity gives the space density of galaxies at redshift z in the magnitude range m to $m+\delta m$, by integrating over the luminosity function at this redshift. The luminosity limits L_1 and L_2 correspond to the apparent magnitudes m and $m+\delta m$ respectively. The apparent magnitude m is related to the absolute magnitude M through the distance modulus:

$$m = (M - 5\log_{10} h) + 5\log_{10}\left(d_{\rm L}/h^{-1}{\rm Mpc}\right) + k(z) + 25. \tag{7.4}$$

Luminosity and magnitude are related through the standard definition:

$$M_1 - M_2 = -2.5\log_{10}\left(\frac{L_1}{L_2}\right). \tag{7.5}$$

In practice, we estimate the luminosity function from the GALFORM model output at a handful of redshifts, z, over the redshift range of interest. For a general calculation of the galaxy number counts, the redshift will in effect vary from 0 to ∞. In our application to the number counts of emission line galaxies seen by a particular instrument, the redshift range is set by the sensitivity of the instrumentation. In the integral to compute the number counts, the luminosity function can be obtained at any redshift by a linear interpolation in redshift between these outputs. Typically the number counts are dominated by the luminosity function around the exponential break, so we need to ensure that the model predictions are robust around this luminosity range.

7.3 Results

7.3.1 Lightcone and integration comparison

For both methods of mock catalogue construction, we use 11 PMILL files or 'subvolumes' out of the 1024 in total used to store each simulation snapshot. The accuracy of the number counts calculation depends on how well we know the luminosity function around the break; we will see later in this chapter that the number counts calculation has converged using 11 subvolumes. Of course, our knowledge of the bright end of the LF may improve on processing a larger number of subvolumes, but the impact on the number counts predictions will be minimal. Hence we do not run all 1024 subvolumes to make predictions for the full simulation volume. Note that whilst we refer to the files the simulation halo merger histories are split into subvolumes, they correspond to a random sample of all the merger histories (without replacements). Hence, each subvolume covers the full simulation volume but at a reduced number density, roughly 1/1024 of the full number density. For this reason, the sampling noise associated with a single subvolume is much less than it would have been if the subvolume files represented spatial subvolumes with all of the halos (i.e. fully sampled) within the subvolume. Hence, even though we are using 1 per cent of all the merger histories available in the simulation (with our 11 subvolumes), we have a higher chance of picking up cluster mass halos than when using the spatial subvolumes employed in earlier simulations. The estimate using the luminosity function from 11/1024 subvolume files will be less noisy than the lightcone version because we are using the full simulation volume, whereas the lightcone only accesses a fractional of the full simulation volume. Using the term *subvolume* to refer to the files

in which the halo merger histories are stored can be confusing, as early developments of GALFORM did split the simulation up into spatial subvolumes. This has since changed to subsampling the merger histories, however, the subvolume name has stuck.

The mock catalogues are constructed over a redshift range that encompasses the redshift range over which *Euclid* is expected to detect Hα emission line galaxies (see previous chapter). For all lightcone-generated catalogues, model universes are simulated from present-day ($z = 0$) up to a maximum redshift, which here is set to be $z_{\rm max} = 2.23$. For the number counts results using the interpolative method we can choose individual redshift snapshots to generate mock catalogues. We choose snapshots that cover the redshift range $0.4 < z < 2.23$ using 9 redshift snapshots that are closest to redshifts at which the Hα emission line luminosity function was estimated from WISP (Colbert et al., 2013) and HiZELS (Sobral et al., 2009, 2013). These redshift snapshots are $z = 0.400, 0.583, 0.692, 0.843, 1.144, 1.460, 2.002$, and 2.223.

We begin by assessing the quality of the interpolated galaxy luminosity functions by comparing the GALFORM $z = 0.843$ Hα luminosity function with the output of linearly interpolating between the $z = 0.583$ and $z = 1.144$ luminosity functions. Each redshift snapshot luminosity function is simulated using 11 P-MILL (Baugh et al., 2019) subvolumes. We present these results in Fig. 7.1 along with the HiZELS $z = 0.843$ and WISP low redshift ($0.3 < z < 0.9$) surveys. We present the luminosity functions used for interpolation as red dotted and dashed lines for redshifts $z = 0.583$ and $z = 1.144$ respectively, and we show both $z = 0.843$ luminosity functions as solid lines. We see good agreement between the exact and interpolated luminosity functions up to the high luminosity end where the data becomes noisy. The interpolated luminosity function replicates the features of the exact output well, including the high luminosity end. From these results, we are confident that the interpolation method to estimate the GALFORM outputs works as intended. We also see, from the HiZELS and WISP, that the Lacey et al. (2016) model does not accurately replicate the observed Hα luminosity function, motivating the requirement to improve the model, which we present in chapter 8. This is not a detailed fitting of the observed data sets, but rather a test of the interpolation method. The HiZELS and WISP data act as a clarification that the results presented here are consistent.

To compare the interpolated Hα emission line galaxy luminosity function to the lightcone generated luminosity function, we analyse the results for a single redshift, choosing $z = 0.4$. For the interpolative method, we simulate 11 P-MILL subvolumes using GALFORM at the specified redshift and a luminosity function is calculated for each subvolume calculated from the galaxy counts in a simulation box of length 542.16 Mpc. A final luminosity function of the Hα emission line galaxies is taken as the summation of the 11 subvolume luminosity functions.

The lightcone luminosity function calculation is slightly more involved. Lightcone galaxies are generated using all redshifts continuously between 0 and the maximum redshift of interest. We use 11 GALFORM subvolumes for lightcone generation and create a lightcone by stitching together all snapshot redshifts from 0 to 2.23 (with a view to computing the number counts of galaxies that will be seen by Euclid). This requires us to specify a direction and solid angle for the lightcone. From the completed lightcone, we count the number of galaxies within a shell, dV, centred on redshift $z = 0.4$ with a thickness of $\Delta z = 0.1$ using a solid angle with an aperture radius of 10 degrees. From this, the luminosity function of the shell approximates the luminosity function of the lightcone at $z = 0.4$ for Hα emission line galaxies.

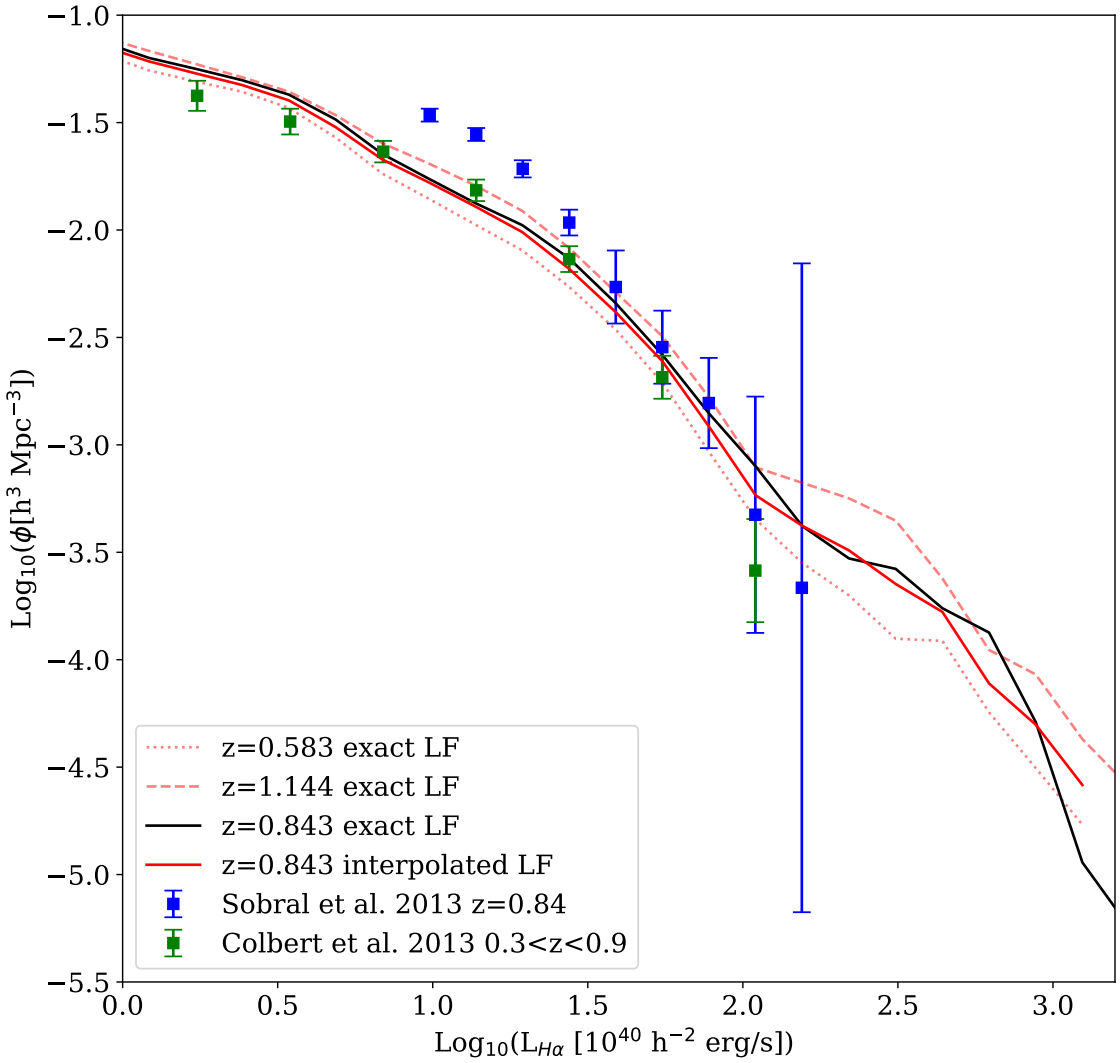


Figure 7.1: Comparing the exact (solid black line) and interpolated (solid red line) GALFORM Hα emission line luminosity function at $z = 0.843$. We interpolate between the $z = 0.583$ (red dotted line) and $z = 1.144$ (red dashed line) GALFORM Hα emission line luminosity functions (based on photo ionization models, see §6.2
). We use 11 subvolumes for each redshift snapshot seen. We compare the results with the equivalent Sobral et al. (2013) HiZELS $z = 0.84$ survey (blue points) and Colbert et al. (2013) WISP low redshift survey (green points).

The results of the two calculations are presented in Fig. 7.2, where we show the direct or interpolation method luminosity function as a solid line, and the lightcone luminosity function as a histogram. (Note in this example we use the luminosity function output directly at $z = 0.4$, rather than an interpolation.) We plot beyond the luminosity resolution limit of GALFORM which is just below $L_{H\alpha}$=$10^{-2} \times 10^{40}$erg s^{-1} to more completely show the trends of the luminosity function. We have also plotted the *Euclid* luminosity limit at redshift $z = 0.4$, calculated from the flux limit of 2×10^{-16}erg s^{-1} cm^{-2} (shown by the vertical dot-dashed line). The GALFORM resolution limit is well below the limit that is relevant for the *Euclid* selection. At higher redshifts, the *Euclid* flux limit will move to brighter luminosities, whereas the luminosity at which the resolution limit of GALFORM becomes apparent does not change much. Hence, it is more challenging to model galaxies visible to *Euclid* at low redshift. In this case, the model can resolve galaxies well below the *Euclid* flux limit, so the model predictions are complete and accurate at all redshifts. We see

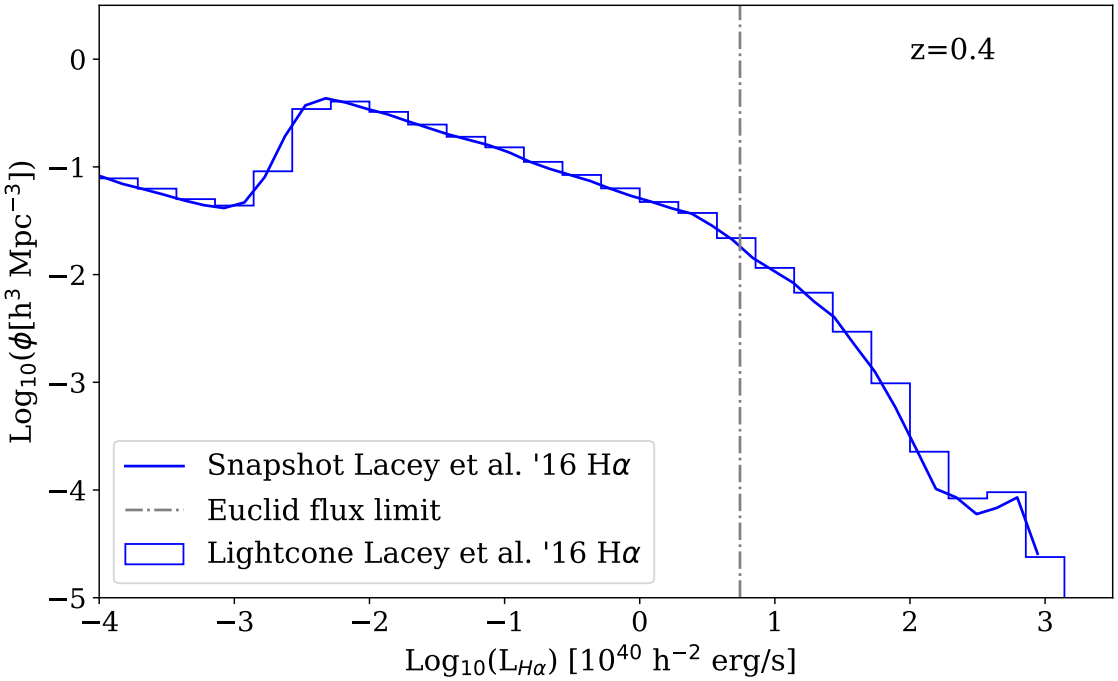


Figure 7.2: Comparing two different approaches to estimating the Hα luminosity function. The smooth solid blue curve shows the luminosity function generated from a single snapshot, and the blue histogram displays the luminosity function estimated from a shell of the lightcone. The lower bound of the luminosity axis corresponds to the resolution limit of the GALFORM simulation ($-3 < \mathrm{Log}_{10}(\mathrm{L_{H\alpha}})/10^{40}\mathrm{h}^{-2}\mathrm{erg/s} < -2$); here the luminosity function turns over artificially, whereas in a hierarchal model, without any mass resolution concerns, we would expect the number of emitters to keep increasing as the luminosity gets fainter. We also plot the Euclid flux limit as a vertical grey dashed line. For the lightcone LF, we have counted the galaxies in a shell with thickness $\Delta z = 0.1$.

a very good agreement between the two predictions. We notice some noise at the bright end of the luminosity function coming from the interpolation method. This is due to the relatively small number of subvolumes used; this noise would be reduced if more subvolumes were processed through GALFORM.

For the cumulative number counts and redshift distribution results, we require galaxy measurements across a range of redshifts. The number count predictions for the lightcone method are generated by selecting the galaxies in the lightcone volume between $0.4 \leq z < 2.23$ and binning them according to their fluxes. With the interpolation method, we have only 9 redshift snapshots available therefore we interpolate over the luminosity function estimated at each snapshot and use this to calculate the number counts of galaxies across the entire redshift range. A comparison of the cumulative number counts for Hα emitters is shown in Fig. 7.3. The interpolative number counts are shown as a solid blue line and the lightcone-generated number counts are shown as a dashed blue line. We calculate the cumulative number counts between a redshift range of $0.4 \leq z < 2.23$, corresponding to the redshift range of the Euclid wide field survey (Laureijs et al., 2011, 2012; Bagley et al., 2020) for Hα emitters. The two number count predictions are in very close agreement with one another. This version of GALFORM predicts 5130 Hα emitters deg^{-2} with a flux above the Euclid flux limit, $f \geq 2 \times 10^{-16}$ erg s^{-1} cm^{-2}, in the wide survey redshift range for both the interpolation and lightcone methods (seen by where the lines cross the y-axis at the flux limit in Fig. 7.3). The divergence seen at the bright flux end between the two methods is likely to be

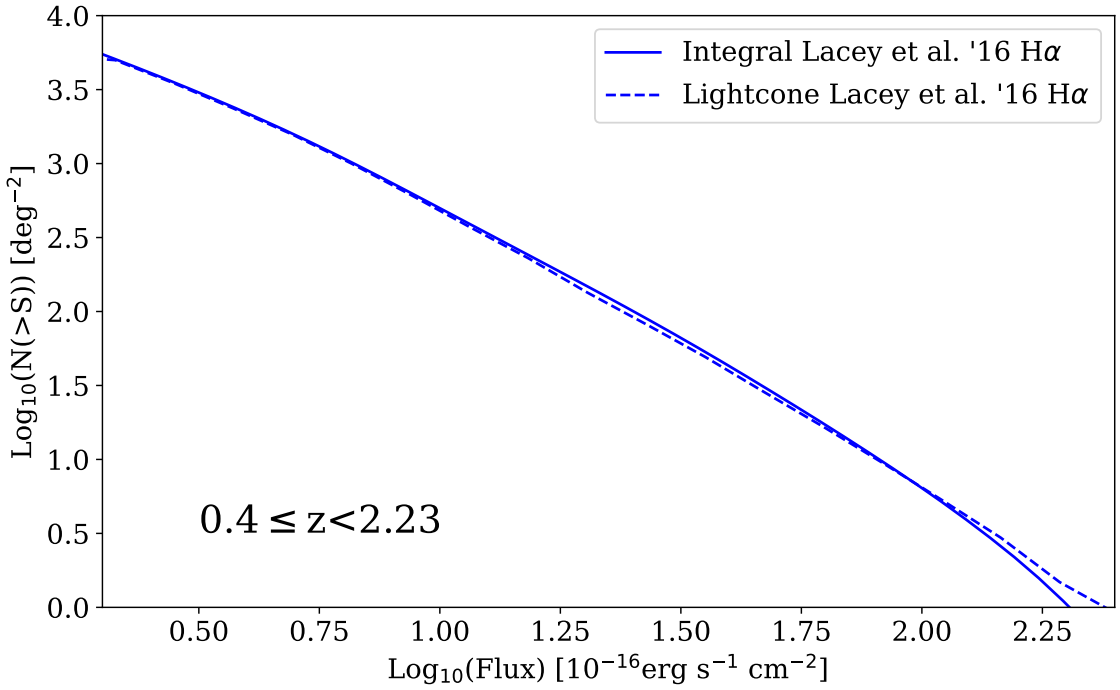


Figure 7.3: Cumulative number counts comparison plot between an interpolative method (solid blue curve) and the baseline lightcone method (dashed blue line) between a redshift range of $0.4 \leq z < 2.23$. We see the interpolation method replicates the lightcone method well up to high flux values.

related to the same noise at the bright end of the luminosity function in Fig. 7.2, due to the lack of galaxies measured in this flux range for the number of subvolumes we have used for this task.

The redshift distribution is calculated in the lightcone by selecting all galaxies within the lightcone that have a flux above the *Euclid* flux limit f, inside a volume of redshift range $0.4 \leq z < 2.23$ as viewed with a view radius of 10 degrees. We then plot a histogram as a function of redshift from the selected galaxies. Following the interpolative method to predict number counts, we interpolate between the single redshift snapshots and select all of the galaxies that have a flux above the *Euclid* limit, f. We use Eqn. 7.3 to compute the number density of galaxies visible to *Euclid* as a function of redshift (i.e. removing the integral over redshift, but integrating over all fluxes brighter than the Eucid limit). We present the redshift distribution results in Fig. 7.4 where we compare the two methods. We see good agreement between the two where the interpolative counts follow the same main features from the base lightcone counts. There is an argument to be made that there is some overestimation of the counts from the interpolative method, but as with the results from the luminosity function (Fig. 7.2) and cumulative number counts (Fig. 7.3), the deviation is low and within acceptable bounds. We are producing counts for all galaxies above the *Euclid* threshold, therefore there will be small variations in the number of bright galaxies for each volume causing noise in the results.

The primary advantage of using the interpolation method for number count predictions is to decrease the computational expense when generating mock catalogues. The lightcone method requires all snapshots from zero to the maximum redshift in our chosen range, in this case, $z_{max} = 2.23$. This equates to 135 snapshot outputs for the P-MILL N-body simulation. Whereas the alternative method requires a fraction of the snapshot outputs and applying a postprocessing

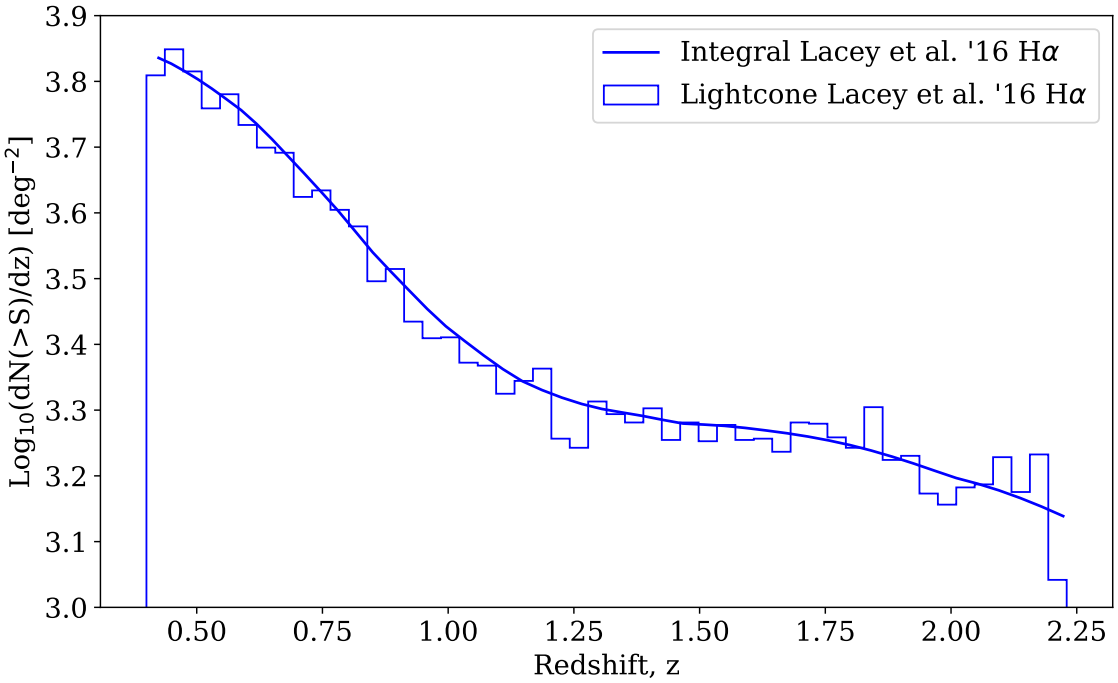


Figure 7.4: Comparison between the interpolation method (solid blue curve) and lightcone (blue histogram) for redshift distribution between the redshift range $0.4 \leq z < 2.23$ and for galaxies brighter than the Euclid flux threshold $f \geq 2 \times 10^{-16}$ erg s^{-1} cm^{-2}. There is good agreement between the two curves.

script to generate the number count predictions based on interpolation. We find a 97.3% decrease in running time when using the interpolation method when compared to the lightcone method generating the same number of subvolumes, 11, and redshift range.

7.3.2 How much can we reduce the computational cost?

We can find further performance increases for the interpolative number counts method by reducing the number of snapshot and subvolume outputs used. We have already discussed the speed increase obtained by moving from the lightcone to the interpolation method when we use the same number of subvolumes in both. Here, we aim to find the minimum number of snapshots and subvolumes used to generate the `GALFORM` luminosity functions used in the number counts that still replicate the lightcone results accurately. The fewer redshift snapshots and subvolumes we interpolate between, the fewer calculations are performed which increases computational efficiency when compared to running a full lightcone. When reducing the number of subvolume files used there are fewer galaxies available with which to make a reliable estimate of the luminosity function. Similarly, when reducing the number of snapshots used, the interpolation is over larger gaps in the redshift so there is less information available and the errors could be larger if the model luminosity function evolves in a complicated manner.

Beginning with the number of subvolumes and redshift snapshots used so far in this thesis, we gradually reduce these two quantities to their lowest values, that being one single subvolume and two redshift snapshots at $z = 0.4$ and $z = 2.23$. We cannot produce a mock catalogue from the interpolation of a single snapshot redshift output (which would be extrapolation). We can reduce

the number of subvolumes down to one subvolume, this generates a luminosity function based on $\sim 0.1\%$ of the available GALFORM DM merger tree history.

For each test, we run the following redshift snapshots and subvolume amounts:

(i) redshifts 0.400, 0.583, 0.692, 0.843, 1.144, 1.460, 1.60, 2.002, 2.223, with 11 subvolumes

(ii) redshifts 0.400, 0.692, 0.843, 1.460, 2.223, with 5 subvolumes

(iii) redshifts 0.400, 0.843, 2.223, with 3 subvolumes

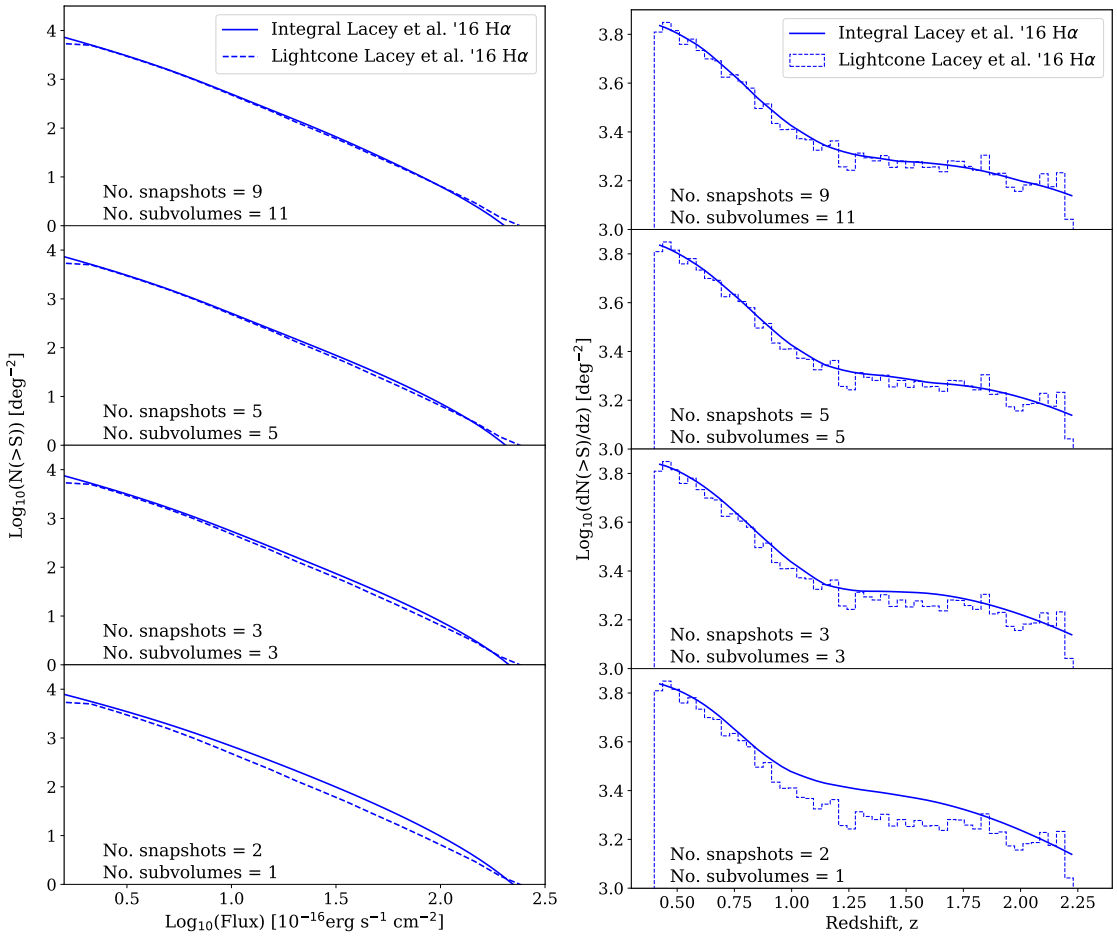
(iv) redshifts 0.400, 2.223, with 1 subvolume.

We present the results of reducing the number of redshift snapshot outputs against the base lightcone outputs in Fig. 7.5. In the left column, we show the cumulative number counts and in the right column, we show the redshift distribution. There is a gentle deviation from the lightcone results as the number of redshifts is cut down as we see the cumulative number counts and redshift distribution both tend towards over-estimation. For the cumulative number counts (Fig. 7.5(a)) the slope of the interpolation method gradually decreases with a decrease in the number of snapshots used, showing that there are overestimations in the number of higher flux emission line galaxies. In the redshift distribution plots (Fig. 7.5(b)) we notice the slope flattening compared to the base lightcone results, with the drop off in number counts up to $z \sim 1.00$ weakens. Surprisingly, the degradation of both results is subtle until the final set of tests (bottom row of Fig. 7.5) where we use the minimum number of redshift snapshots and subvolumes leading to a significant offset between the baseline and the reduced interpolative number counts.

We use a by-eye analysis to determine the most appropriate number of redshift snapshots and subvolumes for the interpolative method to accurately replicate the base lightcone results (which used 9 snapshots and 11 subvolumes). From our results, we conclude that using a set of 5 redshift snapshots and 5 subvolumes (ii) is an optimum solution to balancing accurate results when compared to a lightcone, and computational efficiency.

7.4 Conclusions

In this chapter, we have discussed an efficient method of generating Hα luminosity function and number counts outputs using GALFORM. By utilising interpolation, we can output accurate calculations that match the traditional lightcone method, but with less computational power as fewer redshift snapshots and subvolumes are required for the results. We go further by finding at what point the predictions from the two methods deviate by gradually reducing the number of redshift snapshots and subvolumes used to produce the interpolative predictions. As the number of snapshots and subvolumes reduces, the interpolative method overestimates the cumulative number counts and redshift distribution of emission line galaxies when compared to the base lightcone results. Surprisingly, the level of overestimation against the lightcone base number counts was relatively subtle until the minimum number of redshift snapshots and subvolumes was used (bottom plots of Fig. 7.5). When interpolating between the lowest number of snapshots, with minimum amounts of data, the interpolation method recovers some of the cumulative number counts and



(a) A subfigure (b) A subfigure

Figure 7.5: Results from reducing the number of redshift snapshots and volumes for the interpolative method when compared to the baseline lightcone method for Hα galaxy number counts (a) and redshift distribution (b). The sets of redshifts used in each test are given in §7.3.1.

redshift distribution trends. The over-prediction comes from noise associated with a lower number of samples and redshifts to interpolate between. We find using 5 redshift snapshots between the redshift range $0.4 \leq z < 2.23$ is an acceptable balance of replicating the lightcone number counts and conserving computational costs.

The combination of these results is particularly beneficial as we lead onto the next chapter, where we create almost 100,000 `GALFORM` models and use this interpolative method of generating an improvement on the Lacey et al. (2016) number counts predictions for a Euclid-like survey.

Chapter 8

A new GALFORM model calibrated to predict the number of Hα emitters

8.1 Introduction

Forecasting the expected performance of deep, wide-field cosmological surveys, such as the ESA-led *Euclid* mission (Laureijs et al., 2011; Racca et al., 2016) is essential. The ultimate figure-of-merit of various cosmological probes is dependent upon the observed number density and clustering strength of the type of galaxies being targeted. Therefore, prior knowledge of the number density and clustering of different galaxies, as a function of redshift, is crucial. This is still relevant for *Euclid* post-launch, as the performance of the various detectors is assessed in situ and changes are made to the survey strategy. For example, the mitigation of stray light on the VIS detector has required a tilt in the telescope at the expense of a small reduction in the total area to be scanned in the wide survey, bringing it closer to 13 000 sq deg; the impact of such compromises for the science objectives can be readily assessed if the characteristics of the galaxy population are known accurately. There are two routes to making this characterisation: exploiting existing studies of the target galaxy population or using physically motivated models.

Pozzetti et al. (2016) attempted to describe the Hα luminosity function estimates available at the time (and hence number counts) using empirical models (as described in §6.4). Three different empirical models were fit to the Hα luminosity functions measured by WISP, HiZELS and NICMOS. The resulting simple functional forms for the Hα LF and its evolution with redshift can be integrated numerically to get the number counts. The uncertainty at the time was considerable, with the predicted surface density of Hα emitters varying by more than a factor of two. The uncertainties are intrinsic to the different surveys used in this work, with one being ground-based and one being space-based. WISP has no pre-selection criteria and no S/N cut, which allows for more faint galaxy measurements but comes at the cost of a greater risk of objects being missed from completeness. Conversely, HiZELS does have a flux limit and S/N cut.

Recently, with the addition of further space based data, the situation has improved somewhat, and there have been many new efforts to estimate the number of Hα emitters that *Euclid* (and other upcoming similar surveys such as the *Nancy Grace Roman Telescope* (Green et al., 2012; Spergel et al., 2015; Bailey et al., 2023) are likely to observe (e.g. Colbert et al., 2013; Mehta et al., 2015; Pozzetti et al., 2016; Valentino et al., 2017; Merson et al., 2018; Zhai et al., 2019; Wang et al.,

2022; Zhai et al., 2021).* Bagley et al. (2020) constructed a new data sample of line emitters from several *HST* surveys and forecast the properties of Hα (and [OIII]) emission line galaxies for future surveys such as the *Euclid* and the *Nancy Grace Roman* missions. The new results from Bagley et al. (2020) show a clear preference for the so-called 'pessimistic' model 3 from Pozzetti et al. (2016), which predicted the lowest surface density of ELGs.

In addition to forecasting using available observational data, numerical simulations have been used to produce realistic mock catalogues. With a physical model, it is possible to predict the clustering of the galaxies as well as their abundance (see, for example, Orsi et al. 2010; Merson et al. 2019). Examples of such efforts include Merson et al. (2018), where the authors applied the `Galacticus` (Benson, 2012) semi-analytical model (SAM) of galaxy formation to produce galaxy catalogues and forecast the number densities of Hα emitters using a variety of dust attenuation models. Merson et al. (2018) predict 3900-4800 emitter per sq degree for the *Euclid* selection. However, in this case, *Galacticus* was calibrated to reproduce a variety of observational constraints with particular emphasis on the observations of the population in the local Universe, without any explicit reference to ELGs. This situation was rectified in Zhai et al. (2019), in which the Galacticus SAM was recalibrated using a new N-body simulation suite, UNIT (Chuang et al., 2019) and a different suite of calibration data. They did not limit their model calibration to the local universe, instead, they focused on observational data relevant to the Euclid and Nancy Grace Roman redshift range, in particular, choosing the Hα luminosity function from HiZELS (Geach et al., 2008; Sobral et al., 2009, 2013).

Efficient calibration and exploration of SAMs has been investigated in several previous works, typically in two forms: a direct exploration of the model parameter space, running the full simulation for each set of parameters, and emulation, in which the full calculation is mimicked by a cheaper process that outputs estimations of the full simulation using far less computational power. Despite SAMs being vastly cheaper to run than hydrodynamic simulations, direct exploration of their parameter space is still computationally expensive due to the vast number of model configurations required for a proper search. Unless the parameters are being turned to a small number of datasets, this search will take an excessive length of time. Direct exploration has been investigated in several previous papers. Kampakoglou et al. (2008) used the MCMC technique to calibrate a SAM to multiple datasets. MCMC was used again in Henriques et al. (2009) to calibrate their SAM (DLB07) to several datasets, where they found the choice of dataset the values of the best-fitting parameters, pointing to deficiencies in their model. Lu et al. (2011, 2012) constrained the parameter space for their SAM using Bayesian inference to achieve acceptable fits to the K-band luminosity function (LF), this was expanded to include the HI mass function in Lu et al. (2014). Ruiz et al. (2015) employed a stochastic technique called particle swarm optimization (Kennedy & Eberhart, 1995) to calibrate the SAM SAG (Springel et al., 2001; Cora, 2006; Lagos et al., 2008; Padilla et al., 2014; Gargiulo et al., 2015) to the K-band LF. This process can be visualised as a *swarm* of particles exploring iteratively the multidimensional parameter space, exchanging information as they do so. The second class of SAM calibration involves building a statistical emulator of the SAM which can be evaluated orders of magnitude faster than running the full SAM, the drawback with this method is it is approximate by nature. Bower et al. (2010) and Vernon et al. (2010) constructed a Bayesian approximation technique (as described in Goldstein & Wooff (2007)) to

*The Dark Energy Spectroscopic Instrument Survey targets [OII] emitters in its ELG survey; this line is shifted firmly into the visible part of the spectrum for the target redshift range.

the GALFORM model that can be rapidly evaluated at any point in parameter space to constrain the parameter space which can provide reasonable fits to the K- and b_J-band LFs. The work was extended in Benson & Bower (2010) to explore how adaptable this reduced parameter space was to fit further observational datasets, and in Rodrigues et al. (2017) to calibrate the GALFORM SAM to the galaxy stellar mass function in the local Universe. More recently, Elliott et al. (2021) used a deep learning algorithm to emulate GALFORM across a range of output statistics to promising accuracy. They were able to run many simple MCMC chains to explore the parameter space and investigated how calibration to different datasets constrained the model parameters. The emulation method of model calibration can cope with a big parameter space.

Here, we aim to improve on current calibrations of the GALFORM SAM to forecast the Hα number counts and galaxy bias as seen by the *Euclid* satellite by fitting to recent relevant Hα observations from Bagley et al. (2020). We emulate a version of GALFORM code implemented in the Planck Millenium N-body simulation (Baugh et al., 2019), which uses an improved galaxy merger scheme (devised by Simha & Cole (2017) and was first implemented in GALFORM by Campbell et al. (2015)). We focus specifically on using deep learning to build our emulator. Deep learning allows users to build flexible function approximators that can reveal non-linear relations within data without the need for a strongly pre-defined model. There have been many successful uses in astronomy (e.g. Ravanbakhsh et al., 2016; Schmit & Pritchard, 2018; Perraudin et al., 2019; He et al., 2019; Cranmer et al., 2019; Ntampaka et al., 2019; Zhang et al., 2019; de Oliveira et al., 2020). Here, we demonstrate the accuracy that deep learning algorithms have when emulating SAMs over a range of model outputs useful for number count predictions for *Euclid*. We use a relatively small number of training examples to achieve good accuracy when compared to other calibration methods previously outlined. As a deep learning emulator can be evaluated orders of magnitude more rapidly than running GALFORM in full, we can run many simple MCMC chains to explore the parameter space and identify the range of parameters that fit our calibration datasets. We achieve this by minimizing the absolute error between our emulator output and the observation datasets we wish to fit to, employing a heuristic weighting scheme to the different observational datasets which mimic the process employed by model practitioners. This automation of the calibration process will exhaustively search the parameter space of the model.

Although non-emulation approaches with MCMC and particle swarm optimization offer powerful ways to quantify parameter uncertainty and fit models to particular observables, they are limited in their exploring due to the significant computational expense that comes with running a full SAM for each parameter iteration. Previous approaches that use emulation have focused on reducing the parameter space based on measures of implausibility (that is, incorporating information about the emulator prediction, and the target data with their variances to rule out regions of parameter space). These iterative methods 'zoom in' on parameter space regions which could plausibly contain good fits to a predefined set of a small number of observables. Here, we aim to emulate the GALFORM model across the entire parameter space. Allowing for the exploration of the full parameter space means we can fit to a more diverse combination of observables. We take the method developed in Elliott et al. (2021) and modify it for a more specific need of galaxy number counts by aiming to fit recent redshift distribution observations as well as local Universe LFs. The work presented in this chapter is relevant as the Euclid satellite has launched, and the Nancy Grace Roman mission is coming up. The Hα data has been improved since Pozzetti et al.

(2016)'s study with the observations from the likes of Bagley et al. (2020) therefore we present automated techniques for model parameter space searches including different calibration data.

The layout of this chapter is as follows. In §8.2 we give a brief review of the deep learning approach and describe the emulator design, and in §8.3 we discuss how we find best-fitting parameters using MCMC. In §8.4 we outline the generation of training and testing data for our emulator and describe the observational constraints under consideration. In §8.5 we present our results. In §8.5.1 we review the generation of our training and testing data, in §8.5.2 we review the predictive performance of the emulator, in §8.5.3 we show the results of our model exploration and calibration and the results for our Euclid number counts and galaxy bias predictions. Finally, in §8.6 we review the merits of our methods and outline potential future avenues of work.

8.2 Deep learning emulator

Here we describe the construction of an efficient emulator of `GALFORM` using the `tensorflow` deep learning platform (Abadi et al., 2016). This is a supervised learning problem (also known as associative learning) in which the network is trained by providing it with inputs and matching output patterns. We define the input vector $\mathbf{x}$ to represent a set of `GALFORM` model parameters and predict an output vector $\mathbf{y}$, which consists of the binned statistical properties of the resulting synthetic galaxy population, for example, the galaxy luminosity function. The emulator aims to map the input vector $\mathbf{x}$ to the output vector $\mathbf{y}$ via an unknown function $\hat{f}(.)$ which replaces running the full `GALFORM` model at a fraction of the computational cost. The emulator allows us to thoroughly search a multi-dimensional model parameter space. The problem is one of regression where the outputs are binned floats rather than the probabilities that might be found in classification problems.

The general structure of a multi-layer neural network was shown in Fig.(*Kromek related - redacted*). The first layer (in black) is the input layer with a size equal to the number of entries or components in $\mathbf{x}$. In our case, this is the number of `GALFORM` input parameters or features used to make the predictions, with one neuron per feature. Note that these input parameters are the subset that is being varied; the full parameter space of the model is larger than the 11 parameters that we vary here, but the other parameters are held fixed (for the full list of parameters see Table 1 in Lacey et al. 2016). The final layer is the output layer with one neuron per prediction value. Here the number of output neurons is the total number of bins across all of the chosen statistics. The middle layers of the network are known as hidden layers. The neurons in these layers extract features for mapping an input to an output and the network is trained by evaluating the hidden layer neurons using labelled examples, i.e. with the output from runs of `GALFORM`. Networks with multiple hidden layers are known as deep learning networks. The connections between each neuron have an associated weight, w, and each neuron has a bias, θ. A neural network learns by adjusting these weights and biases from exposure to the training examples according to some learning rule. Each neuron is a simple mathematical function taking a vector of inputs and calculating an output. The i-th neuron in the j-th layer contains a vector of adjustable weights $\mathbf{w}_{ij}$ and an adjustable bias θ_{ij}. The vector $\mathbf{w}_{ij}$ contains all the weights linking a neuron i to each neuron in the previous layer, $j-1$. The data flow from the input to output neurons is strictly passed forward and every neuron in each layer is connected to every neuron in the following layer in what is known as a fully

connected network. Note there are no connections *within* a layer. The total input of neuron i in layer j is a function of the outputs from each neuron in layer $j-1$, $\mathbf{y}_{j-1}$, the neuron vector weights $\mathbf{w}_{ij}$, and the bias of the neuron θ_{ij}. An activation function $\mathcal{F}(.)$ takes the total input of the neuron to produce an output,

$$y_{ij} = \mathcal{F}(\mathbf{w}_{ij} \cdot \mathbf{y}_{j-1} + \theta_{ij}). \tag{8.1}$$

The activation function is often a non-decreasing function of the total input of the neuron, introducing non-linearity to the whole network and allowing for complex representations and functions to develop, which is not possible with a simple linear input-output model. The activation function transforms the output value of the neuron to within certain limits, modified based on the application of the model. If unrestricted by the activation function, the outputs of neurons can explode in magnitude in deeper networks. Generally, some sort of non-linear threshold function is used for this purpose, such as a sigmoid or hyperbolic tangent function. The outputs of the neurons, y_{ij}, are passed to the following layers of neurons repeatedly until the final layer is reached. The output from the final layer is the network predictions $\mathbf{y}$ from inputs $\mathbf{x}$. An activation function is still applied to the final layer but this is usually a linear function in the case of regression.

To adjust the weights assigned to hidden neurons we use the back-propagation learning rule (Rumelhart et al., 1986). During training the predictions from the output layer are compared to the true values and the errors between these two are back-propagated from the output layer to the hidden layers and their weights are adjusted accordingly to minimize an error function. We follow Elliott et al. (2021) by choosing to minimize the mean absolute error function (MAE) between the emulator's predictions of the `GALFORM` outputs and the true outputs

$$\text{MAE} = \frac{1}{n}\sum_{k=1}^{n} |\hat{\mathbf{y}}_k - \mathbf{y}_k|, \tag{8.2}$$

where $\hat{\mathbf{y}}_k$ is the emulator prediction for the k-th sample out of n and $\mathbf{y}_k$ are the values conputed by `GALFORM` for the same parameter values. The MAE is also known as the cost function and reveals how badly (or how well) the network is performing.

The neural network is trained iteratively over many epochs. One epoch is equivalent to the network cycling through every sample in the training set once, and the number of epochs used to train the network is a user choice. An optimizer algorithm is used to change the weights and biases of the neural network by seeking minima on the error surface space, often via a form of gradient descent. The optimizer also specifies the size of steps taken during the gradient descent towards the local minima, known as the learning rate. At the end of each epoch, the adjusted model is tested on a validation sample, which is a subset of the data that has not been used during the training to ensure the model is generalisable to completely unseen data. The number of training epochs is fixed by plotting the MAE against the epoch; this curve flattens off after some number of training epochs so that the precise choice of the number of epochs used is not important once this flat part of the MAE curve has been reached (see e.g. Fig. 8.1).

The final model set-up is tested on a hold-out set of samples to carry out a performance analysis on completely unseen data.

Table 8.1: The GALFORM parameter space investigated assuming a uniform range for each parameter. See § 6.3.1 for an explanation of how each process is modelled and the equations which involve each parameter. The first column gives the parameter name (and units if relevant), the second column gives the range over which the parameter is allowed to vary and the third column lists the process to which the parameter relates.

Parameter	Range	Process
$\nu_{\rm SF}$ [Gyr^{-1}]	0.1 - 4.0	Quiescent star formation
$V_{\rm SN,\,disk}$ [kms^{-1}]	10 - 800	SN feedback
$V_{\rm SN,\,burst}$ [kms^{-1}]	10 - 800	SN feedback
$\gamma_{\rm SN}$	1.0 - 4.0	SN feedback
$\alpha_{\rm ret}$	0.2 - 3.0	SN feedback
$\rm F_{stab}$	0.5 - 1.2	Disk instability
$f_{\rm ellip}$	0.2 - 0.5	Galaxy mergers
$f_{\rm burst}$	0.01 - 0.3	Galaxy mergers
$\tau^{*}_{\rm burst,min}$ [Gyr]	0.01 - 0.2	Starbursts
$f_{\rm SMBH}$	0.001 - 0.05	SMBH growth
$\alpha_{\rm cool}$	0.0 - 4.0	AGN feedback

8.2.1 Inputs and outputs

We aim to develop an emulator to map an input vector $\mathbf{x}$, which is the subset of GALFORM parameters that are allowed to vary, onto an output vector $\mathbf{y}$, which corresponds to the statistical galaxy properties we wish to predict. Our choice of the input parameters that are allowed to vary is made through a combination of physical motivation and informed choices from previous analyses (see § 8.4.2). These parameters and their ranges are shown in Table 8.1. We tune the emulator to predict three statistical galaxy properties calculated from the output of GALFORM to calibrate a model to make accurate predictions for *Euclid*. These statistics are the redshift distribution of Hα emitters between $0.69 \leq z \leq 2.00$, and the luminosity functions in the r and K-bands at $z = 0$ (see § 8.4.2 for more information about these datasets). Each dataset is weighted equally in the metric when the emulator is being constructed.

8.2.2 Network architecture

The final neural network architecture was determined by testing individual hyperparameter configurations one at a time. We take inspiration from Elliott et al. (2021) by starting with an architecture with two hidden layers, each hidden layer containing 512 neurons with the sigmoid activation function, and linear activations on the output layer. Here, we test modifying the choice of activation function, the width of the network (the number of neurons per layer), and the depth of the network (the number of hidden layers). All networks are trained with the same training dataset. We track the MAE loss against the validation dataset at each epoch during training and display the results in Figs. 8.1, 8.2 and 8.3. The network architecture that has the lowest validation MAE loss relative to the other architectures after fine-tuning is the one chosen moving forward. It is worth noting there is a caveat with these tests due to the stochastic nature of training a neural network; an identical network architecture trained on identical training data can display a small variability in its final validation score, so we take this into account when deciding the final network.

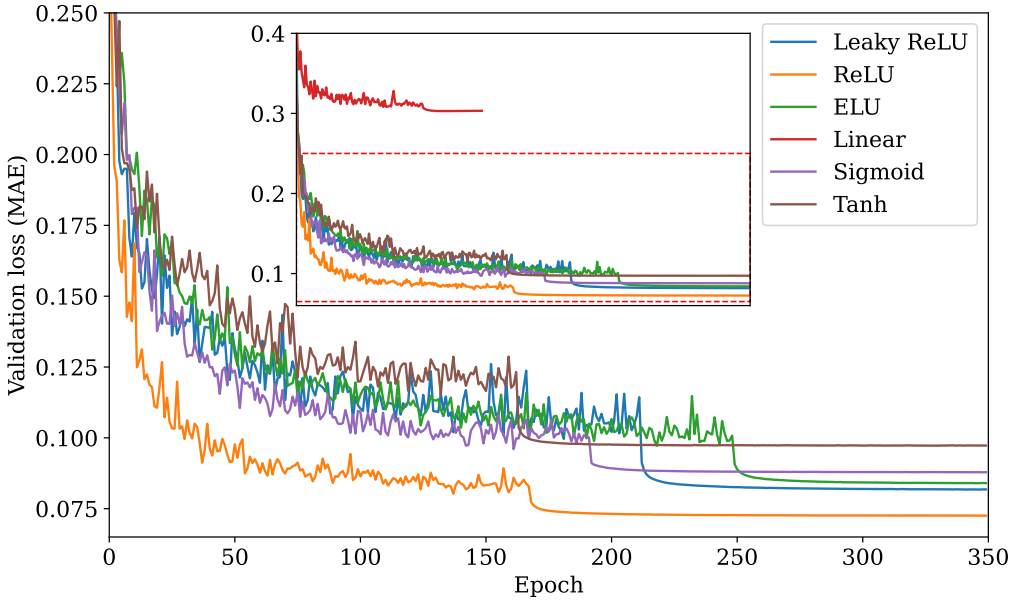


Figure 8.1: Testing a variety of activation functions for the hidden layers of the network architecture. We plot the MAE loss on the validation dataset against the training epoch. A different colour is used for each choice of activation unit, as indicated by the key. Each network has the same architecture of 2 hidden layers, with 512 nodes and a linear output activation function. We display a zoomed-out inset to show the poor loss attained with a linear activation function. The sudden drop in loss value exhibited in all cases, when the curves also appear to become smoother, is due to the fine-tuning stage of training (see text for further details).

Starting from the architecture used in Elliott et al. (2021), we then modify the activation functions, testing a linear function, Logistic Sigmoid, Tanh, Rectified Linear Unit (ReLU) (Nair & Hinton, 2010; Sun et al., 2015), Leaky ReLU (LReLU) (Maas et al., 2013; Xu et al., 2015), and Exponential Linear Unit (ELU) (Clevert et al., 2015), with the results displayed in Fig. 8.1 (for a full review of the many activation functions available see Dubey et al. (2022)). We found that modifying the activation function to a type of rectifier unit was the best option.

Going forward from this point, we test both the ReLU and LReLU activation functions, while modifying the width of our network but keeping the number of hidden layers at two. We vary the width between 200, 512, and 1000 neurons per hidden layer. We want to see if there is a positive trend in terms of a reduction in the MAE when increasing the number of neurons per layer. In Fig. 8.2 we plot the results from both the LReLU (solid line) and the ReLU (dotted) network activation functions. There are training speed benefits to using a thinner network: the percentage increase in training speed for the network to reach epoch 350, between the thinnest network (width 200) and the widest network (width 1000) is $\sim 190\%$ for both the ReLU and LReLU versions. We do see that for both cases the 200-width network does not perform as well as the wider networks as it consistently achieves a higher MAE validation loss after fine-tuning. However, there is no clear gain in performance to support increasing the width of the network beyond 512 neurons. Therefore we will use hidden layer widths of 512 to optimise the performance and training speed.

Finally, we test the depth of the network, that is, the number of hidden layers our network contains. Once again we train two identical networks, one with a LReLU activation function, and the other with the ReLU activation function, shown in Fig. 8.3 as solid and dashed lines

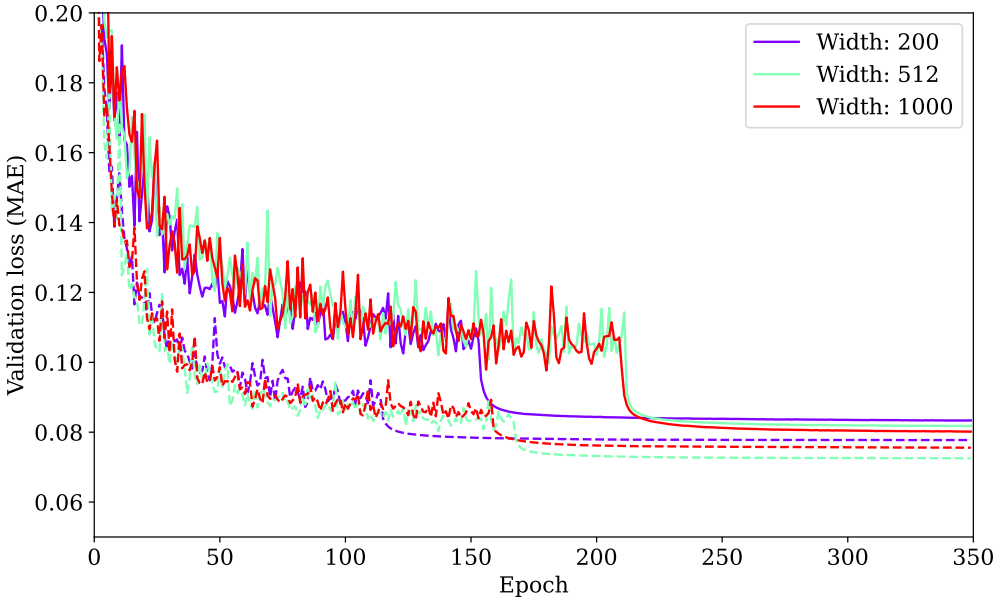


Figure 8.2: Measuring the MAE loss on the validation dataset during training, when altering the hidden layer widths of our network for two activation functions, ReLU (dashed) and LReLU (solid). Each network has two hidden layers and a linear output activation function. There are no significant benefits to increasing the width of our network beyond 512 neurons per layer.

respectively. An interesting observation is the improvements seen with the LReLU network when more layers are included. In Fig. 8.1 we saw the ReLU activation function performs best when two hidden layers were used, but as the number of hidden layers increases the performance increase with the LReLU network puts it ahead of all of the ReLU performances. Furthermore, we do see performance gains when increasing the number of hidden layers up to a certain number when they start to converge on a minimum MAE loss. We find, that for both activation functions, once there are five hidden layers, there are no further significant gains in network performance against the validation set when more layers are used. Computational speed again is a factor here, with the percentage increase in the training time needed between a network with one hidden layer and a network with eight hidden layers being $\sim 217\%$. Our final network architecture, based on the results presented here, is six hidden layers, each with 512 neurons and LReLU activation functions.

Having made this choice, we can explain in a bit more detail the difference between a ReLU and a Leaky ReLU (LeLU). A Leaky ReLU builds from the original ReLU by modifying the handling of negative input values. The original ReLU returns an output of zero for a negative input,

$$\mathcal{F}(s_{ij}) = \max(0, s_{ij}), \tag{8.3}$$

whereas a Leaky ReLU assigns a non-zero slope on the negative end,

$$\mathcal{F}(s_{ij}) = \max(\alpha s_{ij}, s_{ij}). \tag{8.4}$$

In Eqn. 8.4 α is a hyperparameter generally set to 0.01, and s_{ij} is the total input to neuron i in the jth layer. The Leaky ReLU solves the 'dying' ReLU problem (Lu et al., 2019), where a standard ReLU can become inactive and only output zero for any input value. In this case, it can

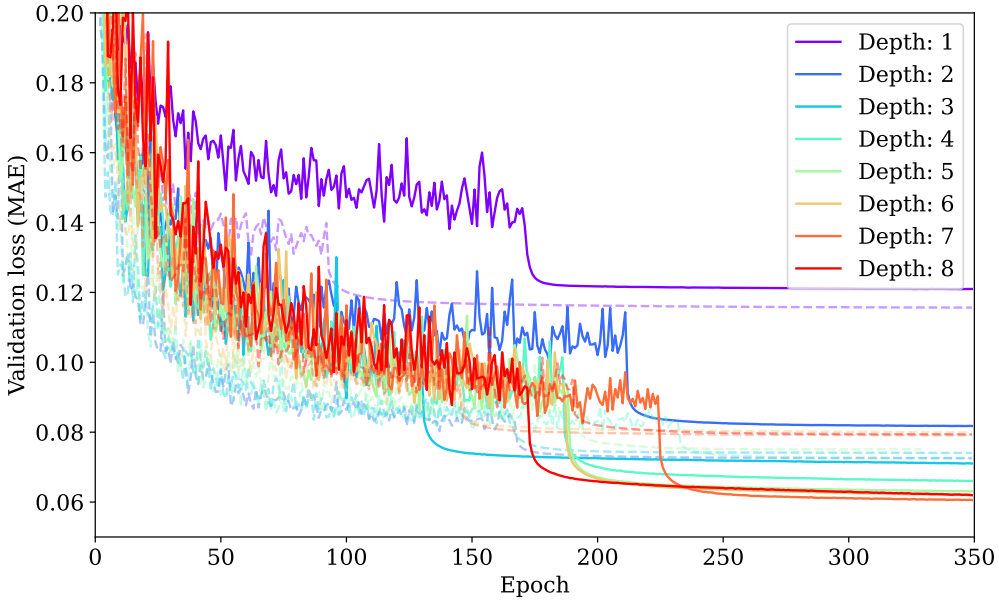


Figure 8.3: MAE validation loss during training when modifying the number of hidden layers in the network, with different colours indicating different numbers of layers, as shown by the legend. We keep the width of the network fixed at 512 and show results for two activation function regimes, LReLU (solid) and ReLU (dashed). We see the LReLU function has greater potential for improvement than the ReLU networks. An increase in depth does improve the performance of our network up to a depth of five or six layers. Beyond this number, there is only a modest improvement in the MAE at the expense of an increase in the computational cost.

never recover and can lead to network regions becoming 'inactive'. We find using a Leaky ReLU instead of ReLU improves the MAE performance during training on the validation set, reducing the average MAE loss by $\sim$ 29%.

We make use of the Adaptive Momentum Estimation (Adam) optimiser which is a popular momentum-based gradient descent optimization algorithm (Kingma & Ba, 2014; Reddi et al., 2019) and set the learning rate to 0.005. We add the AMSGRAD variation (Tran et al., 2019) which aims to improve the performance of Adam around the minima on the error surface using a stochastic method, which evaluates the weights after every mini-batch iteration (mini-batches are small subsets of the whole training set). At the end of each epoch, we save the model weights if the performance on the validation set has improved (as measured by Eqn. 8.2) and continue training until there is no improvement for 30 epochs at which point training is stopped. Then the learning rate is reduced to 10^{-5} for a fine-tuning stage with the RMSprop optimizer (Tieleman & Hinton, 2012), allowing us to take small gradient steps towards the minima of the error surface. RMSprop uses stochastic gradient descent and assumes the error surface is a quadratic bowl. This method boosts the performance of our emulator as we can descend into fine local minima, and we measure improvements to our network by tracking the MAE of the validation samples throughout training. We see evidence of this in Figs 8.1, 8.2 and 8.3 where the gradient rapidly drops when the network transitions into fine-tuning training mode.

8.2.3 Ensembling

Before training, the weights of a network are often initialized according to some distribution, often random. We use an initializer described in Glorot & Bengio (2010). Due to the stochastic nature of the training process training a single network is insufficient since the error surface is likely to contain many local minima and one network is unlikely to traverse enough of the weight space to find the best possible mapping. Over-fitting is also a potential problem due to the large number of parameters especially as more layers are added. One solution to these issues is ensembling multiple network predictions. This involves training several identical networks with different weight initializations and shuffling the validation and training sets between each model in the ensemble so the models are distributed from input to output. This should allow for a more robust final prediction. We average over the predictions of each model to negate any over- or under-fitting to different features of the data. Ensemble refers to the stack of identical neural network models that we average over to produce a final prediction.

Using this method, we train 5 separate networks as described above, each with the same model architecture. The final emulator prediction is the average of the predictions from the ensemble of models. Ensembling is a rich avenue for exploration with reviews for popular ensembling methods available (e.g. Opitz & Maclin, 1999; Sagi & Rokach, 2018; Ganaie et al., 2022). There is scope in the future to improve on this method via a method called stacking (Wolpert, 1992) where the ensemble networks themselves are the inputs to a single network with generalizes the outputs for improved results. However, this works best where the ensemble networks are varied and provide different information, such as different architectures or combining different types of machine learning algorithms.

8.3 Parameter fitting

We aim to use our emulator for inference on target datasets; that is, fitting our model to given datasets. We employ a Markov-Chain Monte Carlo (MCMC) sampler to compare our generated models against the observed datasets with the goal of sampling from a set of parameters that produces the models that best fit the observables. The Metropolis-Hastings algorithm (Robert et al., 2004) is a common and simple method of executing an MCMC, generating serially correlated draws from a sequence of probability distributions, eventually converging to a given target distribution. The means of convergence comes from the minimization of the absolute error between the emulator output and the observational constraints. We note that as we are minimizing the absolute error between multiple data sets, we do not take into account their associated errors. We do wish to weight certain datasets over others to allow us to investigate the effect of requiring better fits to some datasets and how this affects the reproduction of other datasets, as well as see how the optimal parameter choices change as a result. We therefore introduce a modified version of the MAE (introduced in Eqn 8.2) which includes a vector of heuristic weights, $\mathbf{W}$, to vary the contribution of the residuals from constraint i to the total error,

$$\mathrm{MAE}^{\mathrm{obs}}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{n^{\mathrm{obs}}} \sum_{i=1}^{n^{\mathrm{obs}}} \frac{\mathbf{W}_i}{n_i^{\mathrm{obs}}} \frac{|\mathbf{y}_i - \hat{\mathbf{y}}_i|}{\boldsymbol{\sigma}_i}, \tag{8.5}$$

where $\hat{\mathbf{y}}_i$ is the predicted value of the i-th observable constraint, and $\mathbf{y}_i$ is the corresponding observable value across n_i^{obs} datapoints. Due to $\hat{\mathbf{y}}_i$ and $\mathbf{y}_i$ being vector quantities, the modulus represents the L1 norm. $\boldsymbol{\sigma}_i$ is a vector of errors corresponding to $\mathbf{y}_i$. We sum over the n^{obs} observable constraints. The different observational datasets contain different numbers of data points, therefore we divide the i-th dataset's weighted absolute error by the number of data points, n_i^{obs}, for equal contribution to the mean error result. In later sections, we refer to Eqn 8.5 as the mean absolute error (MAE).

The Metropolis-Hastings procedure for updating a Markov Chain compares the likelihoods from the current parameter location or state to a proposed (new) state. Assuming uniform priors throughout, each chain is initialized on a random point in the parameter space which is assigned as the *current* state, $\mathbf{x}$. Then we sample a proposed state, $\mathbf{x}'$, from independent Laplacian proposal distributions about $\mathbf{x}$, $\mathcal{L}(\mathbf{x}'|\boldsymbol{\mu}, \mathbf{b}) = (1/2\mathbf{b}) \exp(-|\mathbf{x}' - \boldsymbol{\mu}|/\mathbf{b})$ where $\boldsymbol{\mu}$=$\mathbf{x}$ and the scale parameter vector $\mathbf{b}$ is set as 1/20th of the parameter ranges given in Table 8.2. The proposed state must satisfy the condition that the proposal lies within the defined parameter bounds given in Table 8.2. We decide whether the proposed state is accepted or not by measuring the likelihood improvement of emulator predictions to the observational data from the current to the proposed state using a Laplacian likelihood with scale parameter b_{obs} = 0.005. Taking the ratio of likelihoods at states $\mathbf{x}'$ and $\mathbf{x}$ is known as the *acceptance ratio*, α,

$$\alpha = \frac{\mathcal{L}(f_*(\mathbf{x}')|\boldsymbol{\mu}, b_{\text{obs}})}{\mathcal{L}(f_*(\mathbf{x})|\boldsymbol{\mu}, b_{\text{obs}})}, \tag{8.6}$$

where $\boldsymbol{\mu}$ represents the vector of observable values, $f_*(\cdot)$ is the emulator and the scale parameter b_{obs} = 0.005. In technicality, we could use a ratio of errors as an acceptance ratio in our MCMC, however, doing so may not align with the principles of Bayesian inference and so could have implications on the accuracy and efficiency of our algorithm. The likelihood is often used in Bayesian inference due to its probabilistic interpretation, as it provides a measure of how well the model explains the observed data given a set of parameters. The acceptance ratio is compared to an acceptance criterion, u, which is a random uniform number $u \in [0, 1]$; a proposed state is *accepted* if $\alpha \geq u$, in which case $\mathbf{x} = \mathbf{x}'$ and the next sample is drawn from a Laplacian centred on the new state, or a proposed state is *rejected* if $\alpha < u$ for which case we sample again from the original Laplacian centred on $\mathbf{x}$. Using this method, if the error between the emulator predictions and the observables is improved when moving from state $\mathbf{x}$ to $\mathbf{x}'$ the sample is always accepted, else we accept the proposed state $\mathbf{x}'$ with a probability α. We expect the density of accepted samples to trace the regions in the parameter space which give the best fits to the observational data. At the start of the chain, there will be a burn-in phase as the accepted samples tend towards local maxima in parameter space so we discard the first 50% of accepted samples removing this burn-in phase. Testing multiple chain lengths we find chains converge to local MAE minima (given by Eqn. 8.5) within the first 5,000 samples and so we choose the chain length as 7,500 (after accounting for the burn-in phase).

8.4 Datasets

Our decision as to which `GALFORM` input parameters to vary comes from a combination of physical motivation and informed choices from previous analyses, largely that of Elliott et al. (2021). We

differ from their parameter choices by focusing more on the contribution from quiescent galaxies and less on galaxies experiencing a starburst. For a Euclid-type instrument observing Hα galaxies, we find burst galaxies only affect the extremely bright end of the luminosity function and would have little impact on the overall number counts (see for example the predictions from Lacey et al. 2011 for the number counts of galaxies in the UV, which is also sensitive to recent star formation). Close to the Euclid flux limit quiescent galaxies are the dominant type. Burst galaxies do, however, dominate the high flux tail of the Hα emitter counts, but this is not important for the overall clustering measurement.

Below we outline the process of obtaining our training and testing data and list the calibration and comparison datasets.

8.4.1 Training and testing data

This work uses a supervised machine-learning method to emulate running a computationally expensive model, `GALFORM`. Training the emulator requires running the full model. Generally, the more samples used during training, the better the output predictions should be. However, to create the large number of parameter examples required for this work `GALFORM` is too expensive to run at every snapshot in the redshift range of interest (see chapter 7). Instead, we chose to interpolate between the outputs at a select number of redshifts; in Chapter 7 we demonstrated the number of redshifts and the fraction of the simulation volume that need to be modelled to produce accurate predictions for the number counts.

Despite the computational gains made with the interpolation method for computing number counts and redshift distributions that was explained and tested in Chapter 7, we are still limited by how many evaluations of `GALFORM` we can perform. Nevertheless, we aimed to produce 3000 model runs for different parameter combinations to satisfy the training and testing of the deep learning emulator. Model parameters were generated via Latin hypercube sampling for even and efficient spacing between the parameter values (as described in Loh, 1996; Bower et al., 2010). Latin hypercubes are particularly useful when exploring multiple input variables that may have some interaction. A Latin hypercube works by dividing the range of each input variable into equally probable intervals, and then randomly selecting from each interval. The result is a grid-like structure of a diverse set of samples. The parameter ranges sampled are given by Table 8.2; the choice of the subset of parameters to be varied was influenced by previous parameter explorations by Lacey et al. (2016) and Elliott et al. (2021).

The Latin hypercube sampler generated 3000 sets of the 11 parameters which iteratively replaced the corresponding input parameters in the base Lacey et al. (2016) model. This resulted in 3000 `GALFORM` outputs, each with appropriate Hα redshift distribution and $z = 0$ r and $K-$band LFs made using the interpolation method described. The `GALFORM` inputs and outputs created the input-output vector pairs, ($\mathbf{x}_i$, $\mathbf{y}_i$) for the deep learning emulator, where $\mathbf{x}_i$ is the ith set of Latin hypercube model parameters and $\mathbf{y}_i$ is the corresponding output vector of the redshift distribution and LFs. The python package `sci-kit learn` allows for the separation of the samples randomly into three sets: the training set, the validation set, and the holdout set. The training and validation set is used during the training of the emulator, and the hold-out set will be kept separate for evaluating performance on unseen data. The ratio of training samples to hold-out test samples

was 29:1 and for each network trained, 20% of the training samples were randomly chosen as the validation data.

8.4.2 Calibration and comparison datasets

Using several datasets, we will use our emulator to calibrate GALFORM for Euclid-like mocks. Traditionally, GALFORM has been calibrated mostly using local data, as these have been the measurements with the smallest errors (see Lacey et al. 2016). We continue this trend by using the r and $K-$band LFs measured from the GAMA survey (Driver et al., 2012); here, these data replace the older $b_{\rm J}$ and K-band LFs typically used to calibrate GALFORM. This choice has the advantage that the same team has done the data reduction and made the assumptions about the k-correction and any evolutionary corrections. In addition, we use the Hα redshift distribution measured by (Bagley et al., 2020). Using calibration datasets at different redshifts greatly reduces the volume of the viable parameter space.

The full list of calibration and comparison datasets and their respective selection criteria are as follows:

(i) For the Hα redshift distribution, we calibrate our emulator to the redshift distribution from Bagley et al. (2020). They used measurements from a combination of two slitless spectroscopic WFC3-IR datasets, 3D-HST+AGHAST and the WISP survey (Atek et al., 2010) to construct a Euclid-like sample. They detect Hα+[NII]-emitting galaxies in the redshift range $0.9 \leq z \leq 1.6$ with line fluxes $\geq 2 \times 10^{-16}$erg s^{-1}cm^{-2}.

(ii) For the $z = 0$ K-band and $r-$band luminosity functions, we make use of data from Driver et al. (2012) who used the Galaxy and Mass Assembly Survey (GAMA) dataset, combined with GALEX, SLOAN Digital Sky Survey (SDSS) and UKIRT) Infrared Deep Sky Survey (UKIDSS) imaging to construct the low-redshift ($z < 0.1$) galaxy luminosity functions in multiple bands.

We also compare our best-fitting models to the previous local LF calibration data (the $K-$band LF measured by Cole et al. 2001 and the $b_{\rm J}$ measured by Norberg et al. 2002). This is to see if the new local calibration datasets still give good fits to the old calibration data; this is an indirect way of seeing (through a model) if these observational LFs are consistent with one another.

8.5 Results

8.5.1 GALFORM runs for training and testing

We draw 3000 samples from the parameter ranges presented in Table 8.1 using a Latin hypercube sampler (Loh, 1996). We visualise the distribution of samples for three of the parameters in Fig. 8.4 as a corner plot, presenting three parameters that influence the SN feedback. We show histograms of the distribution for each parameter along the diagonal. We see that this sampling method produces a uniform distribution across the parameter ranges, meaning there is little bias

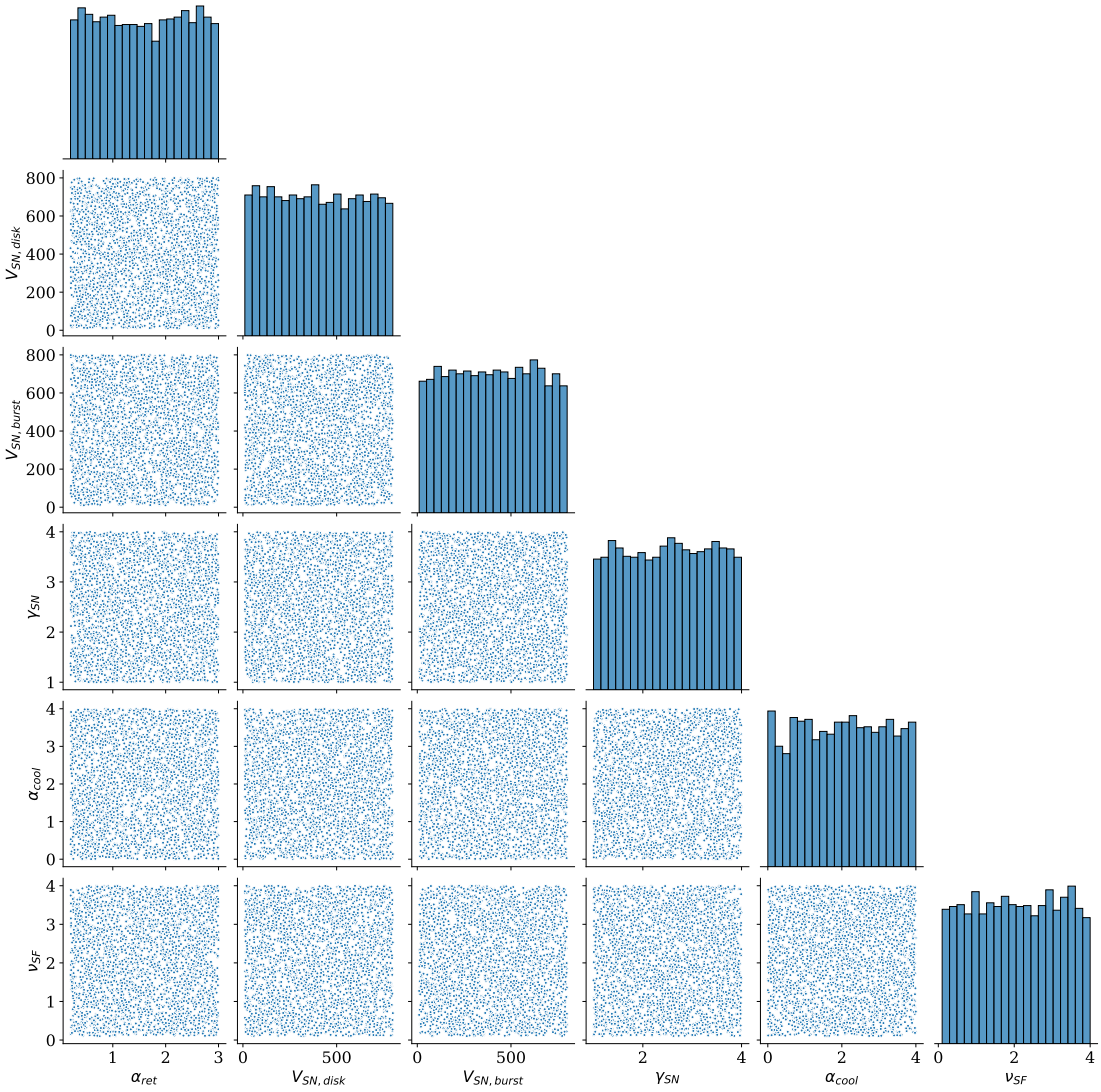


Figure 8.4: A visualisation of the distribution of the 3000 Latin hypercube generated samples across 11 parameters. We present six parameters out of the 11, four relating to SN feedback; α_{ret}, $V_{SN,disk}$, $V_{SN,burst}$, and γ_{SN}, plus ν_{SF} relating to the star formation of quiescent galaxies, and α_{cool} for AGN feedback. These plots show a uniform, unbiased sampling of the parameter ranges.

towards any particular range of values for any parameter. This is consistent across the 11 parameters sampled.

The 3000 parameter sets are combined into a data frame that GALFORM automatically reads to generate a model from each set. Due to the large computational expense of generating this many models, we choose the interpolation technique for Hα number count calculations as this means we can get away with running fewer redshift snapshots and subvolumes and still accurately replicate the number count outputs from the much more expensive lightcone method (see chapter 7). We remind the reader that the term subvolume refers to the random sampling of all merger histories (without replacement), each subvolume covered the full simulation volume but at a reduced number density (see §7.3.1). We start with the Lacey et al. (2016) model and replace the parameters highlighted in Table 8.1 before running GALFORM with five redshift snapshots (plus $z = 0$) and five subvolumes each. The number of snapshots and subvolumes needed was determined in §7.3.2. The redshifts chosen are $z = 0.69, 0.90, 1.14, 1.60, 2.00$. These were selected to cover the redshift range probed

by the Bagley et al. (2020) Hα redshift distribution, which matches well the expectations for the *Euclid* space mission (Laureijs et al., 2011; Racca et al., 2016). We also generate an extra redshift snapshot at $z = 0$ (with the same number of subvolumes) as we wish to fit the $K-$ and $r-$band luminosity function outputs to the Driver et al. (2012) luminosity functions. In total, 3000 GALFORM models were produced, each with six redshift snapshots, which themselves contain five subvolumes, equating to 90000 individual GALFORM runs.

Out of the 3000 GALFORM submissions, all but one were completed successfully. However, there are examples where one or more of a model's subvolumes for a particular redshift snapshot failed to complete. In these cases, the required outputs are calculated from the remaining subvolumes. We note that for the cases where one or more subvolumes failed to complete, a small number of those model parameters had values at the extremities of the prior ranges. We noticed that a lot of the models that failed had a γ_{SN} parameter towards the upper boundary of the prior range. For the model that had zero completed subvolumes for any redshift snapshot, the parameter γ_{SN} had a value of 3.99 which is at the peak of this parameter range. In view of the predictions from the nearby models which did complete, it is unlikely that the best-fitting model for our chosen calibration datasets will have parameter values close to those of the small number of models that failed. In the case where some subvolumes were successful, we simply took this into account when calculating the model predictions. Despite this, we were able to produce 2999 Hα number count outputs and the corresponding model $K-$ and $r-$band luminosity functions at $z = 0$.

The completed GALFORM models were examined to see if any models provided good fits to the redshift distribution and luminosity function calibration datasets. If any models happen to give a good fit to the calibration datasets then the objective of this project will have been met, without further optimisation being necessary. For each model, we use Eqn. 8.5 to calculate the error to compare to the observables of Bagley et al. (2020) and Driver et al. (2012) for the redshift distribution and $z = 0$ luminosity functions respectively. For each statistic, we compare to recent GALFORM parameter exploration efforts from Lacey et al. (2016) and Elliott et al. (2021). We present the best fitting models for the redshift distribution, $K-$ and $r-$band $z = 0$ luminosity functions in Figs. 8.5, 8.6 and 8.7 respectively, showing the model that fits best to the redshift distribution observables as a solid purple line, and the best-fit model to the luminosity functions as a solid red line. We also plot the Lacey et al. (2016) data as a dashed blue line and the Elliott et al. (2021) data as a dashed green line. We did individually find the best models for each statistic and found that the same model that best fits the $K-$band luminosity function observation, also best fits the $r-$band luminosity function observation, hence we label this the best fit ϕ model.

In Fig. 8.5 we present the predictions from the best fitting GALFORM model (purple) to the redshift distribution data of Bagley et al. (2020) out of the 2999 models available in the Latin hypercube sampling of the model parameter space. This model is a very good fit for the observed data, with an MAE of 0.083. There is some degree of overprediction at low redshift but the model prediction is still within the error bounds for most of the data points. This model represents a significant improvement over the redshift distributions produced by the Lacey et al. (2016) and Elliott et al. (2021) models as they achieve an MAE score of 0.263 and 0.425 respectively. They fail to recreate the trend of the observed distribution, underpredicting the number counts as a function of redshift at low to medium redshift values. We also plot the model which is the best fit to the $z = 0$ $K-$ and $r-$band Driver et al. (2012) luminosity functions. It is clear that this model, also,

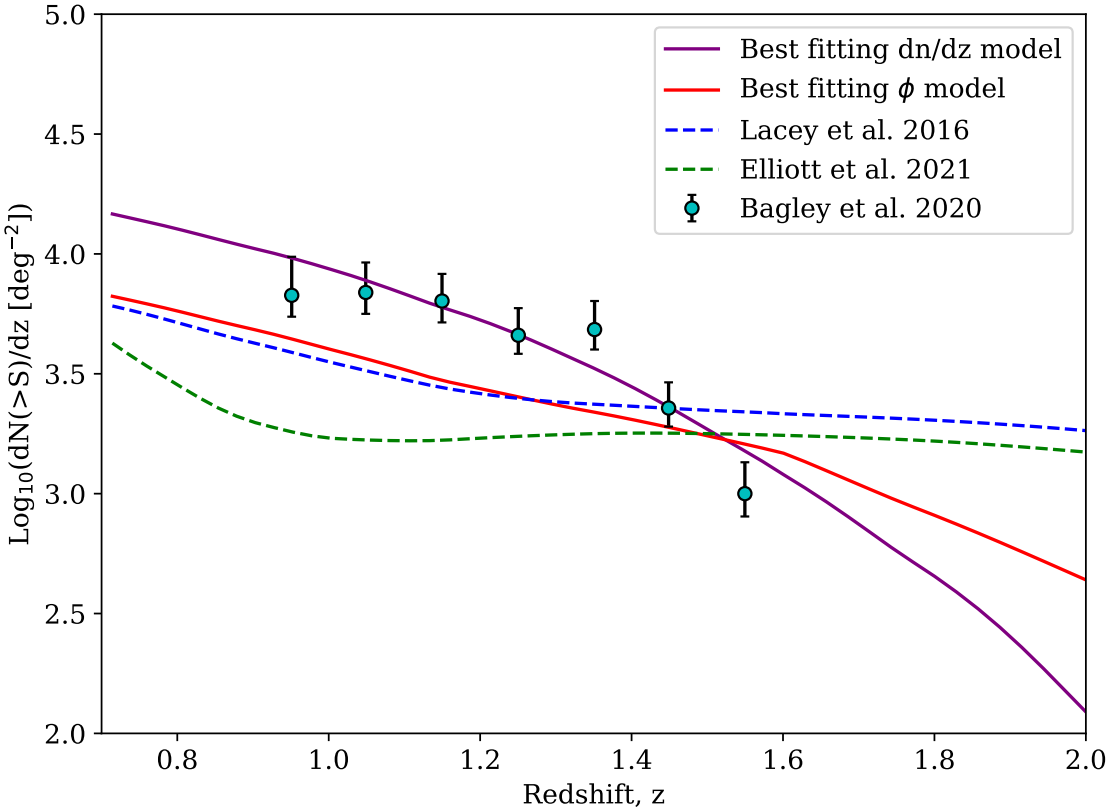


Figure 8.5: Best fitting GALFORM model (purple) from the 2999 Latin hypercube sampled models when compared to the Bagley et al. (2020) Hα redshift distribution data (symbols with errorbars). Note that no optimisation has been performed in this comparison; we are simply plotting the best-fitting model from the Latin hypercube sampling of the parameter space. We also plot the best fitting GALFORM model when compared to Driver et al. (2012) luminosity functions alone (red) and previous GALFORM models by Lacey et al. (2016) (blue dashed) and Elliott et al. (2021) (green dashed), which used mostly local data in their calibration.

does not suitably fit the observed redshift distribution data underpredicting the number of galaxies out to the higher redshifts.

When fitting individually to the observed $z = 0$ $K-$ and $r-$band Driver et al. (2012) luminosity functions, the best fitting GALFORM model (red) for each band was the same. The $K-$band luminosity function results are shown in Fig. 8.6 and $r-$band luminosity function results are shown in Fig. 8.7. We once again plot the luminosity functions for both the Lacey et al. (2016) and Elliott et al. (2021) models and the GALFORM model that best fits the redshift distribution observations (purple). For the $K-$band luminosity function, the model has an MAE score of 0.175, and for the $r-$band a score of 0.150. This is compared to the Lacey et al. (2016) model scoring 0.204 for the K-band, and 0.247 for the r-band, and the Elliott et al. (2021) model scoring 0.210 and 0.304 for the K- and r-band respectively. The model performs well in predicting the turnover magnitude of both luminosity functions, where the Lacey et al. (2016) model predicts a slightly brighter magnitude turnover, and Elliott et al. (2021) predicts a slightly fainter turnover magnitude; this is more prominent in the $K-$band luminosity function plot (Fig. 8.6). The previous predictions by Lacey et al. (2016) and particularly Elliott et al. (2021) do well at fitting to the data therefore any improvements we can find will be more subtle compared to the redshift distribution. We use this data to add further constraints to our final model when predicting the

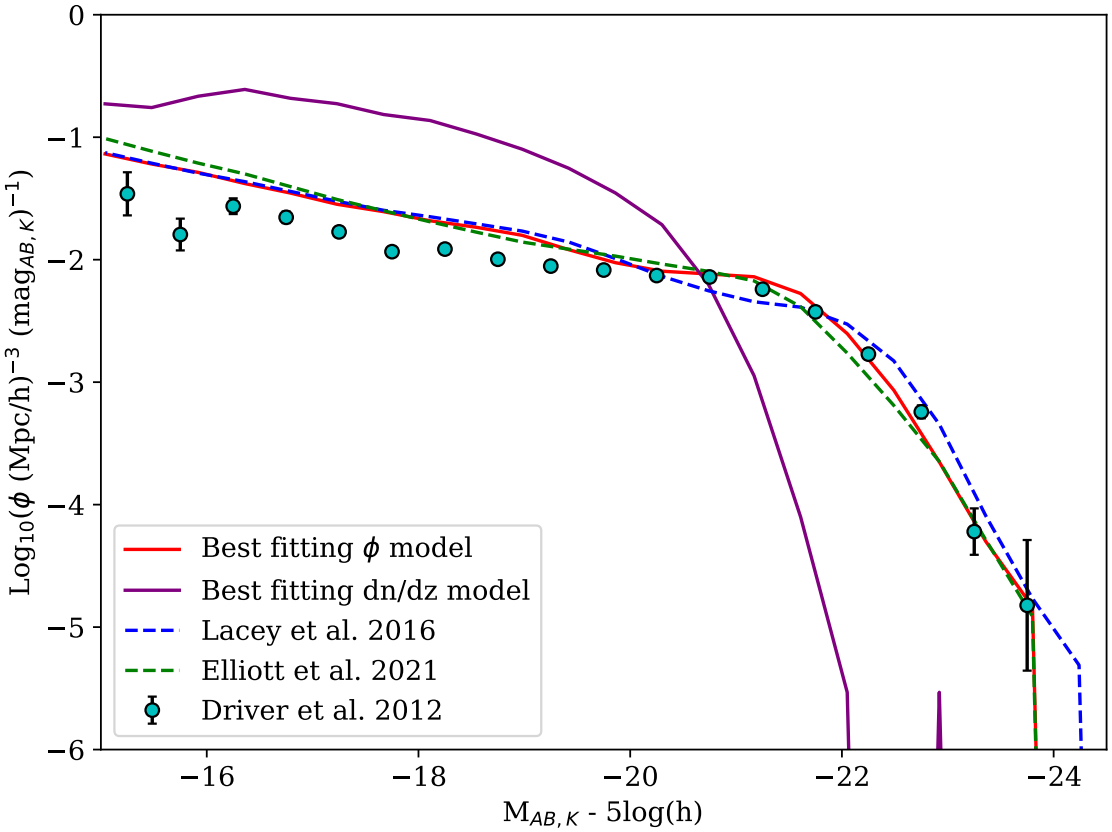


Figure 8.6: Best fitting GALFORM model (red) from 2999 Latin hypercube samples models when compared to the Driver et al. (2012) $K-$band luminosity function data. This is the same best-fitting model when fit to the Driver et al. (2012) $r-$band luminosity function. We also plot the best fitting GALFORM model when fit to the Bagley et al. (2020) redshift distribution (purple) (Fig. 8.5) and the Lacey et al. (2016) (blue dashed) and Elliott et al. (2021) (green dashed) models. Note that these latter two models were calibrated against mostly local data, but different LFs than the one shown here.

redshift distribution. The reason for this is explained when looking at the GALFORM model that is the best fit to the redshift distribution data of Bagley et al. (2020), seen in Fig. 8.5. Although this model fits very well with the redshift distribution data, it performs very poorly against the luminosity function, outputting an unrealistic luminosity function. This model over-predicts the number of galaxies at faint magnitudes and then massively under-predicts galaxies at the bright end. This validates our efforts to improve on existing GALFORM models by finding a set of parameters that maximises the compromise between fitting the redshift distribution and luminosity functions via MCMC methods; we need to explicitly include all three observational calibration datasets in the metric assessing model performance and to optimize other the model parameter space. The results presented above give us some hope that the GALFORM model can reproduce each of the calibration datasets; we need to perform an optimisation to further tune the parameters to attain agreement for all datasets from a single model.

The 2999 successful GALFORM model outputs will serve as the training and testing data for our emulator development.

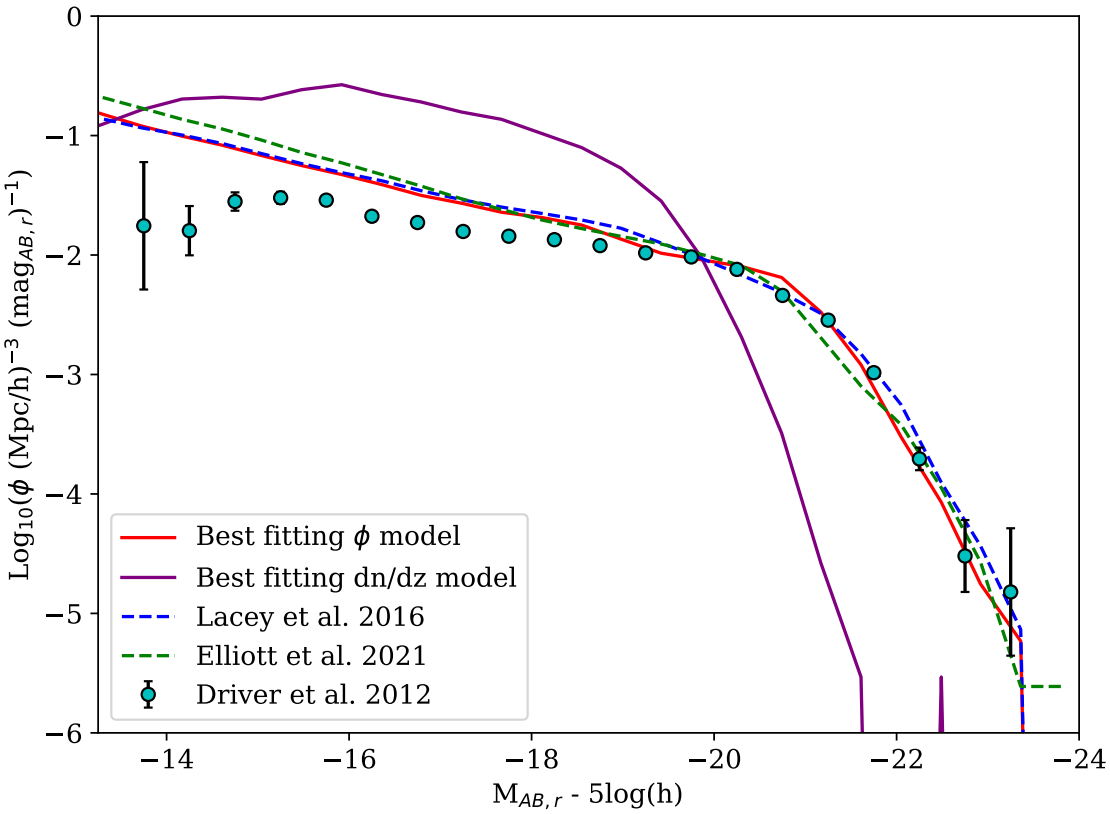


Figure 8.7: Best fitting GALFORM model (red) from 2999 Latin hypercube sampled models when compared to the Driver et al. (2012) $r-$band luminosity function data. This is the same model seen in Fig. 8.6. We also plot the best fitting GALFORM model when fit to the Bagley et al. (2020) redshift distribution alone (purple) (Fig. 8.5) and the Lacey et al. (2016) (blue dashed) and Elliott et al. (2021) (green dashed) models for reference.

8.5.2 Emulator performance

The development of the architecture for the individual networks of the emulator is described in §8.2. During the architecture training phase, we only ran our networks up to a training epoch of 350. For the final network, we chose not to limit the epochs but instead included an early stopping clause which stopped and saved the network at its lowest training MAE validation loss score. On average the networks found their lowest loss score between 500 and 700 epochs. We also took inspiration from Elliott et al. (2021) by ensembling multiple networks of equal architecture and averaging over the outputs to produce a final result which is our emulator result. Elliott et al. (2021) averaged over ten equal networks. We tested an ensemble of five and ten networks and found little to no improvement in the performance of the emulator against the hold-out set when ten networks were used over averaging across five networks. Note that going from one network to an ensemble of five gives roughly a ten percent reduction in the MAE. Furthermore, the more networks to be averaged over, the greater the computational time which becomes important as we run an MCMC across a substantial amount of walkers each with on the order of 10,000 steps. Therefore, our emulator consists of five equal architecture networks (described in § 8.2). We want to evaluate the emulator's ability to output GALFORM at new points in the parameter space. Our set of 2999 GALFORM outputs was split up with 96.67 per cent of the outputs used for training our emulator as described in § 8.2 (equating to 2899 parameter combinations) and the remaining 3.33 per cent

(100 parameter combinations) being used as unseen outputs for testing purposes (hold-out set). This split maximises the number of training samples and provides an appropriate range of unseen test samples to evaluate the network reliably. During the training of each network, we randomly split the 2899 parameter output combinations into a training set and a validation set with 20 per cent going towards validation (580 parameter combinations). For each network version trained in the emulator ensemble, the training and validation sets were shuffled.

In the top panel of Fig. 8.8 we show the emulator predictions against the hold-out set outputs from the corresponding full GALFORM runs. A perfect emulator would follow the $y = x$ line (dotted) with no scatter. In general, we see the emulator following a tight relation to the diagonal across the three statistics, indicating that the emulator is accurately predicting GALFORM output for the holdout set parameters, without any significant biases and a reasonably small scatter. Out of the three statistics, the redshift distribution predictions appear to have a greater uncertainty than the K and r-band luminosity functions. However, this is largely an artefact of these predictions spanning a smaller dynamic range than the other statistics, so this scatter plot is 'zoomed-in' compared to the others (covering just over 4.5 decades in scales as opposed to six decades in the other panels). In the lower panel of Fig. 8.8 we show the performance of the emulator across the three statistics on a sample of the holdout set parameters, plotting the emulator outputs as dashed lines and the true GALFORM outputs as solid lines. The parameter samples drawn from the holdout set were chosen to reflect the range of emulator performances, including parameters that the emulator most struggled with for each statistic. Each colour across the three panels is the same combination of parameters from the holdout set. The luminosity function plots display the ability of the emulator to predict beyond the resolution of GALFORM when the true model was generated with a finite sample of the subvolumes of the simulation, which can result in some luminosity bins being empty at the bright end. The lower panel of Fig. 8.8 reveals some sources of inaccuracies in the predictions, particularly the redshift distribution, which is more prone to exhibiting noisy behaviour for some choices of parameters, for example, the low redshift distribution (orange line) is poorly predicted. The error bars for the redshift distribution predictions are fairly even across the redshift bins and this is reflected in the lower panel plots where the majority of predictions follow the shape of the true GALFORM output but with a varying degree of offset. The main source of errors for the luminosity function predictions is seen at low values of ϕ. We do see that at the bright end of the luminosity function plots the predictions can become noisy but the overall shape is well captured.

The majority of emulator predictions for the redshift distribution are reasonably close to the GALFORM predictions, but we do come across cases with substantial discrepancies between the true and predicted outputs (as exhibited by the orange line in the bottom row of Fig. 8.8). We see far fewer cases like this within the holdout set of poor predictions of the true GALFORM outputs when it comes to both luminosity functions, with the largest discrepancy seen in the blue parameter set. These poor predictions are usually indications that the training data did not contain sufficient examples of this behaviour as these examples appear to be extreme cases of the output and so are less common. The emulator constructs a function $f_*(\cdot)$ by fitting it to the training examples, where $f_*(\cdot)$ can interpolate between the points in the parameter space. However, the interpolation is less reliable in the sparser regions of the space, such as at the extremities of our parameter bounds.

We can see that at the bright ends of the K- and r-band luminosity functions in Fig. 8.8, the emulator tends to slightly over-predict the GALFORM output, seen by the distribution of points

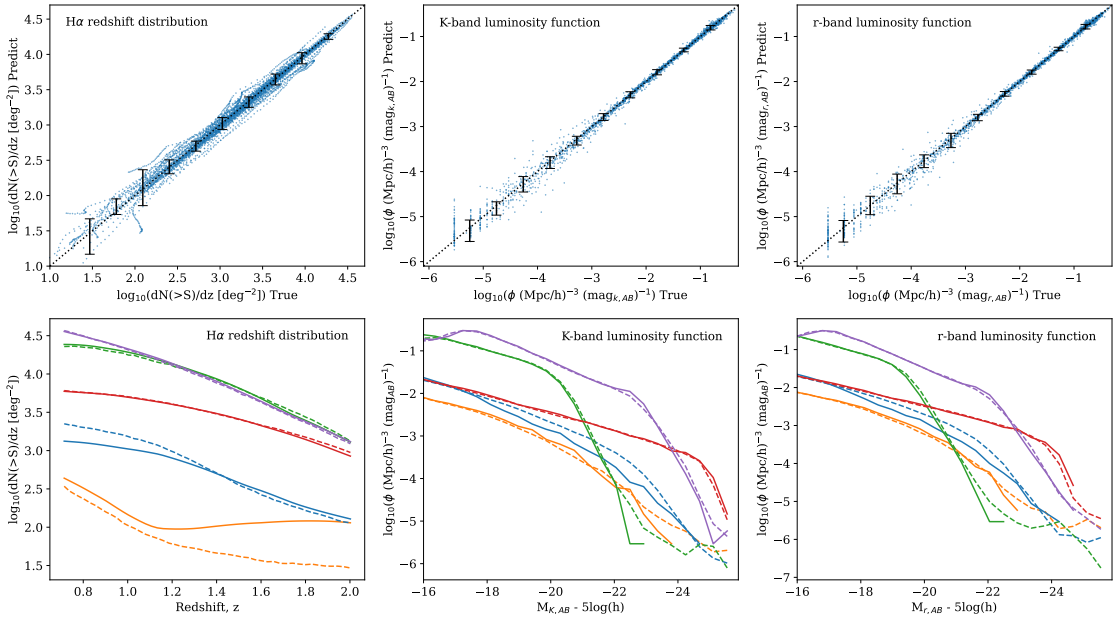


Figure 8.8: Emulator performance across the three calibration statistics computed with the holdout parameter sets. The top row shows the emulator output (y-axis) against the true GALFORM output (x-axis). Black error bars indicate the 10-90th percentile range of the residuals. The bottom row shows a draw of emulator outputs (dashed lines) and true GALFORM outputs (solid lines) for selected parameter sets. In these panels, different colours denote different parameter sets.

curving towards the prediction side. This is a consequence of using a sub-region of the full P-MILL simulation volume (0.6% of the full volume). This leads to noisy data at low galaxy number densities, and, as seen in the bottom row of Fig. 8.8, cut-offs at different luminosities for different choices of parameters, for example, the true green parameter set cuts off well before the true red parameter set for both the K- and r-band luminosity function plots. Our emulator has to output a fixed number of bins, therefore during training, we mask any luminosity bins which contain zero galaxies when computing the loss. This leads to the emulator having fewer brighter luminosity bins to fit which are biased towards having higher values of ϕ in these brighter bins. This causes more cases of over-prediction at these luminosities. This problem is minor since the Driver et al. (2012) luminosity function data does not sample ϕ to very low number densities. These issues could be resolved by evaluating GALFORM on a larger fraction of the P-MILL simulation volume, although this would be more computationally expensive with little gain.

We also evaluate the performances of the emulator against an existing GALFORM model by Lacey et al. (2016). The emulator's predictions across the three statistics with the original Lacey et al. (2016) GALFORM model are shown in Fig. 8.9, where the original GALFORM model run is shown as a solid line, and the emulator predictions are shown as dashed lines. We generate the redshift distribution of emission line galaxies for the Lacey et al. (2016) model using the interpolation method described in chapter 7, using nine redshift snapshots between $0.69 \leq z \leq 2.00$ and 11 subvolumes of the P-MILL simulation volume. We see an overall good fit to the true model, with the redshift distribution overpredicting the true GALFORM model by a small amount. This coincides with our findings of the emulator performance on the holdout set above. For the redshift distribution, our emulator can still accurately identify the shape of the true model. Both the K- and r-band luminosity functions do well at matching the true GALFORM model, with the only deviation seen around the turnover point at magnitudes $\sim$-22 for the K-band and $\sim$-21 for the

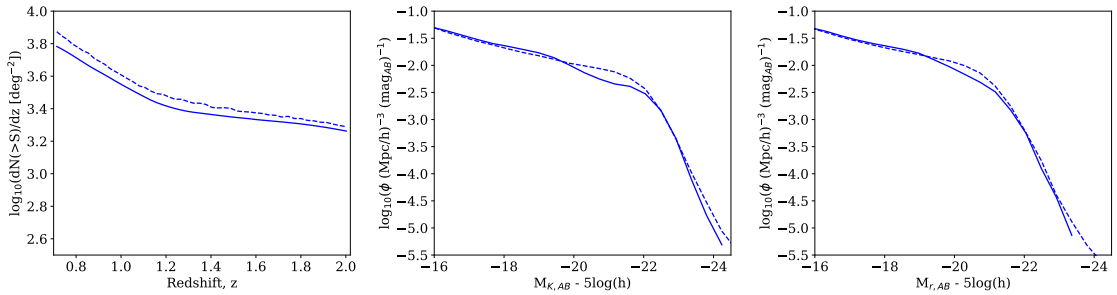


Figure 8.9: Emulator predictions (dashed lines) using the Lacey et al. (2016) GALFORM parameters compared with the true GALFORM outputs (solid lines). We predict the redshift distribution (left), and the $z = 0$ K- (middle) and r-band (right) luminosity functions.

r-band. Our emulator is unable to recreate the dipped features around these magnitudes which indicates a deficiency of these types of parameters within our training set. The possible changes we could make to the emulator's training set that we highlighted before would improve our predictions against the Lacey et al. (2016) parameter set.

8.5.2.1 Scaling with training set size

To illustrate the scaling of emulator performance with the number of full GALFORM calculations, we train three emulators with 900, 1900 and 2899 samples of training parameters respectively (in each case split with 20 per cent of the samples going towards validation). The emulators consist of an ensemble of five identical networks each trained on the same (shuffled) training and validation sets, and the performance is evaluated on the same 100 holdout parameter samples as has been used so far. We present our findings in Fig. 8.10 where the emulator performance in terms of the MAE computed for the holdout set is plotted against the number of full model runs used to train the emulator. The dashed line shows the average performance of the individual networks, and the solid line shows the performance of the ensemble. The top panel displays the average MAE score of the holdout set across the three statistics and the second, third and fourth panels show the average redshift distribution, K- and r-band luminosity function MAEs of the holdout parameters respectively. The emulator shows a clear reduction in the MAE with increasing numbers of training samples, and using an ensemble of models provides a near-consistent improvement in performance compared to using a single model. The performance of the emulator is improved by almost $\sim 12\%$ by averaging over an ensemble of neural networks, rather than just using one network. We find that when using more than 5 networks in the ensemble, the performance increase of adding more networks saturates. This plot also gives insight into the contributions to the overall performance of our emulator as we see the redshift distribution MAE score consistently higher than the luminosity functions, pushing the average across the three statistics higher. This provides useful insight when we aim to fit parameters to observations and how we manage the errors. The MAE results give us confidence that the model can learn a function which provides a good approximation to GALFORM across the full parameter space.

The results shown display the potential of the emulator given sufficient training resources. We have chosen to investigate the model over a significant number of parameters covering large ranges (Table 8.1). Therefore the emulator will benefit greatly from more full GALFORM examples. However, this comes with a computational cost of generating said GALFORM output samples.

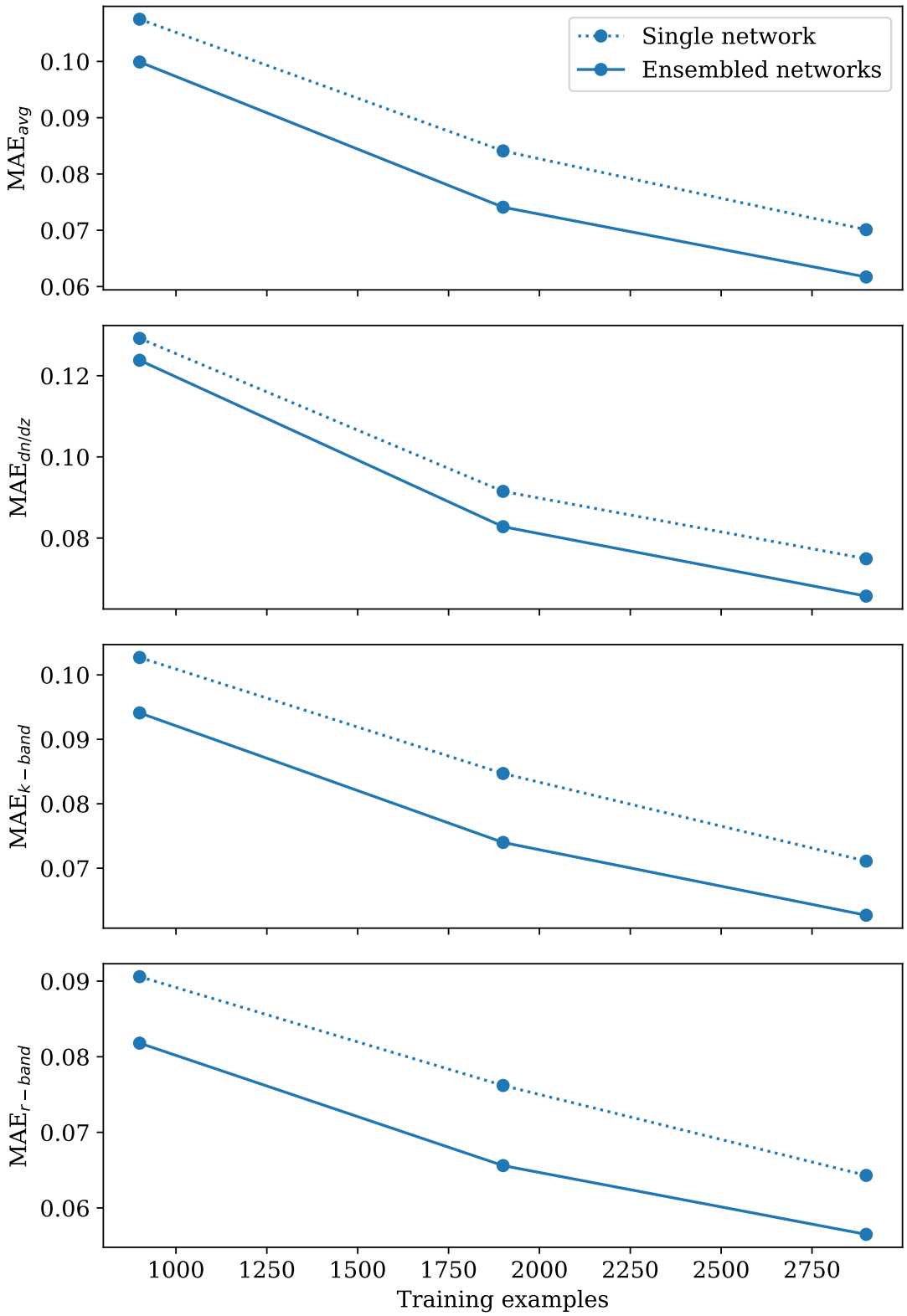


Figure 8.10: Mean absolute error (MAE) of a single (dotted lines) and ensembled (solid lines) emulator trained with increasing numbers, N, of full `GALFORM` runs. The networks were trained on 900, 1900, and 2899 parameter samples and performance was evaluated on the same holdout set containing 100 samples.

8.5.3 Parameter fitting on the calibration data - model optimisation

Here we apply the methods described in § 8.3 to calibrate the model to the datasets introduced in §8.4.2. We begin by investigating the tensions between the three statistics by adjusting the weightings applied to the residuals between our emulator prediction and each dataset (given by Eqn. 8.5) and then performing an MCMC parameter search to see how the best-fitting parameter choices respond. In Fig. 8.11 we show the emulator predictions for the best-fitting parameters found from five MCMC chains using a number of different weighting schemes. To achieve accurate predictions for Euclid we need to fit closely to the redshift distribution data of Bagley et al. (2020). However, to reduce the overall model parameter space, it is also important to constrain the model to reproduce the local universe luminosity functions. Hence we need to find an effective balance of fits between the two. When the weighting to the redshift distribution data is low, for example, a weighting of one or two (blue and orange lines in Fig. 8.11 respectively), we see poor accuracy reproduction of the redshift distribution data and strong performance with the luminosity functions, particularly around the LF break. As the redshift distribution weighting increases, we notice increasing deviation at the bright- and faint ends of the luminosity functions, but an improved fit to the redshift distribution data, with the predicted distributions being within the error bounds of the observations. When evaluating the fits to the luminosity functions, a weighting of four to the redshift distribution (green line) constraint finds the exponential L^* break magnitude well and stays just as tight to the high redshift data points in the redshift distribution plot as the weighting of six (purple line). The spread across the luminosity functions for the different weightings is surprisingly low given that the spread in the redshift distribution fits is large in comparison. This could be an early indicator that there are multiple regions in the parameter space that can fit these models, according to the emulator. This likely arises from the error of the emulator outputs, particularly for the redshift distribution predictions. It is worth noting that these parameter fits come from a low number of MCMC chains and, therefore we expect to see improvements in the best-fitting parameters when we evaluate 100 MCMC chains.

We are unable to find a set of parameters that can reproduce all of the constraints effectively with equal weighting. Therefore we need to give more weight to the constraint with the highest importance, which is the redshift distribution function. We have seen that there are trade-offs or tensions to consider when aiming to find the best-fitting model. Fitting with a greater emphasis on the redshift distribution degrades the fit to the K- and r-band luminosity functions, as expected. With these considerations in mind, we choose heuristic weights such that the redshift distribution is strongly weighted while retaining appropriate fitting to the luminosity functions, particularly minimizing the over-prediction of the bright ends. Ideally, the precise choice of weight should not matter, so long as it is within some range. Therefore, we weight the redshfit distribution constraint four times higher than the luminosity functions when calculating the MAE given by Eqn. 8.5. That is, we set $W_i = 4$ for the redshift distribution constraint, and apply a single weighting to both K- and r-band luminosity function constraints.

With the weighting scheme between the three statistics fixed, we re-calibrate the `GALFORM` model across the three constraints to produce an estimate of the best-fitting parameters. We run 100 MCMC chains with our emulator, each with 7,500 steps in length after the burn-in phase (which is 7,500 steps). The residual of each sample is computed using the emulator and our modified weighted MAE function and we find that the minimum MAE obtained with each chain

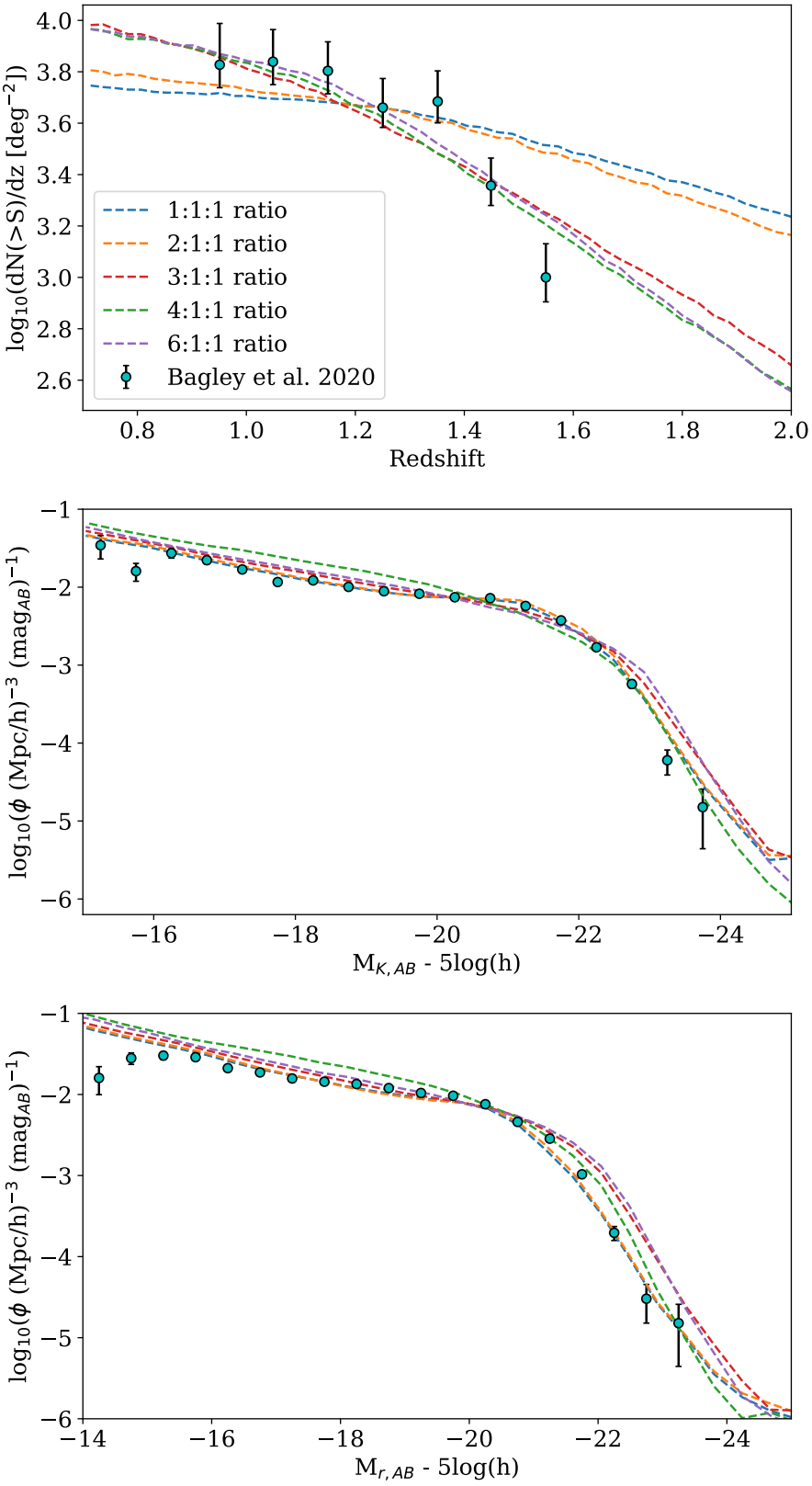


Figure 8.11: Best MCMC fits across five different weighting schemes, increasing the weighting towards the redshift distribution to display the tensions between the constraints. We show a redshift distribution weight value W of one (blue), two (orange), three (red), four (green) and six (purple), plotted with the Bagley et al. (2020) Hα redshift distribution and Driver et al. (2012) $z = 0$ K- and r-band luminosity functions.

lies in the range $\sim 0.25 - 0.28$. As we have seen in §8.5.2, our emulator outputs have an associated error, so we can not confidently discern which parameter sets give the best fit to the observational data with the emulator alone. Hence, we evaluate the parameters that gave the lowest MAE value from each of the 100 MCMC chains with the `GALFORM` code.

In Fig. 8.12 we illustrate the regions in the parameter space that are most sampled with our MCMC using 20 MCMC chains. The shaded regions represent the accepted samples from our chains, each 7500 steps long after discarding the burn-in. The shading indicates the density of the accepted samples, with the darker regions corresponding to the more favoured regions of the parameter space in this weighting scheme. We contour the density such that the darkest region corresponds to the 25th percentile, then the 50th percentile and the lightest region is the 75th percentile. Also shown in Fig. 8.12 are 1D histograms of the density of accepted samples. From our findings, for some parameters, a reasonably large range of parameter values results in acceptable fits across our constraints. However, though as plotted in one or two dimensions the space appears widely sampled, when moving to a higher dimension the acceptable regions are reduced significantly. This is the effect of high dimensionality of the parameter space, as described in Bower et al. (2010). We see that to fit the three statistics using the weighting scheme described, the fits prefer high values of $\gamma_{\rm SN} \sim 4.0$ possibly beyond the sampling parameter boundary. We have the option to extend our parameter space, but doing so will explore parameters beyond the known space used to train our emulator. This could result in more uncertain predictions. Furthermore, we do not want to extend our parameter ranges to extremes so they represent unphysical choices for the processes being modelled. We also observe a bimodal distribution for the $V_{\rm SN,\,burst}$ parameter which tends towards the lower and upper boundary of our parameter range at $\sim$ 10kms^{-1} and $\sim$ 800kms^{-1} respectively. In contrast, the parameters $f_{\rm ellip}$, $f_{\rm burst}$, and $\tau_{\rm *burst,min}$ display little importance in the one dimension shown to the fitting as they show almost uniform sampling, whereas the parameters that contribute to the SN and AGN feedback are more tightly bound.

Out of the lowest MAE parameters from the 100 MCMC chains, we plot the best 50 sets of parameters as evaluated with the `GALFORM` code in Fig. 8.13. These runs cover a range of weighted MAE, between $0.25 - 0.31$, with the remaining runs extending to a weighted MAE of 0.64. We indicate the run with the lowest weighted MAE as a solid red line, and the remaining top 49 runs as solid blue lines. These 50 best-fitting runs all generalise the constraining datasets well and confirm the effectiveness of our MCMC and emulator while also indicating the scale of uncertainty present with our method. We isolate the run with the lowest MAE in Fig. 8.14, along with the emulator prediction for the same set of parameters (red dashed line) to observe the difference, along with the statistical galaxy properties of the model presented in Lacey et al. (2016) as the solid grey line. We see that there is a spread of possible parameters. Therefore the best-fitting parameters presented are just one realization of many possible choices due to the effects of calibrating to multiple data sets with tensions between them and the degeneracies between the parameters.

The spread across the 50 best MCMC chains as evaluated by `GALFORM` is tight across the K- and r-band LFs and there is somewhat more variance about the redshift distribution outputs particularly as we get to higher redshifts. Our best set of parameters is within the error bound of most of the Bagley et al. (2020) data points. Due to the tensions between the redshift distribution of ELGs and the local LFs, the general trend of fits the LFs is to over-predict the luminosity function at the bright end. This is particularly evident for the $z = 0$ r-band luminosity function, although

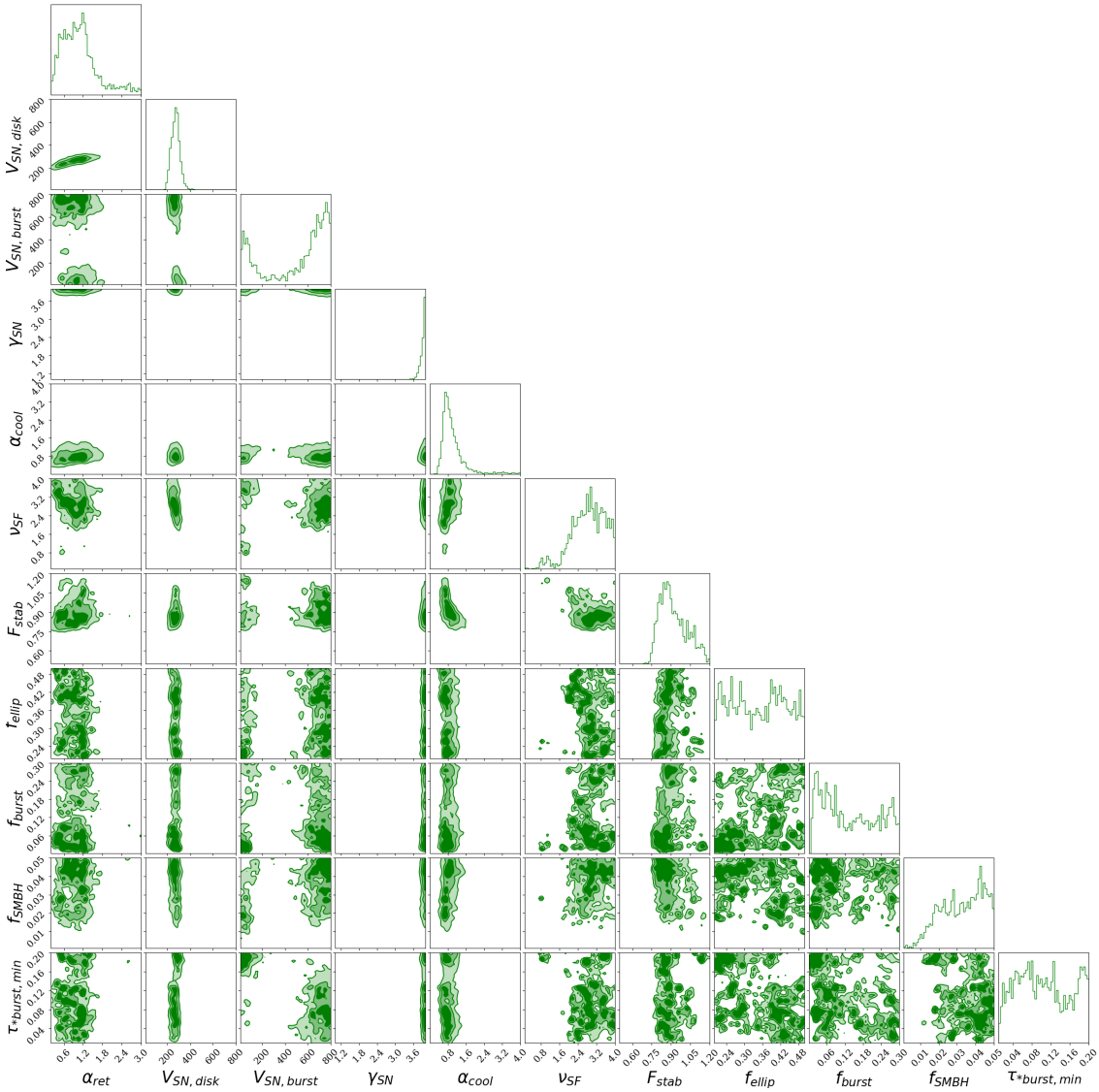


Figure 8.12: Accepted samples from 20 MCMC chains for fits to the redshift distribution, K- and r-band LFs. The first 50% of samples were discarded to allow for burn-in. The histograms show the marginalised distribution of the parameters. The ranges on each axis are the same as those quoted in Table 8.1. The shading corresponds to the density of chain steps, with darker colours corresponding to more densely sampled regions. The darkest regions correspond to the 25th percentile and the lighter regions to the 50th and 75th percentiles.

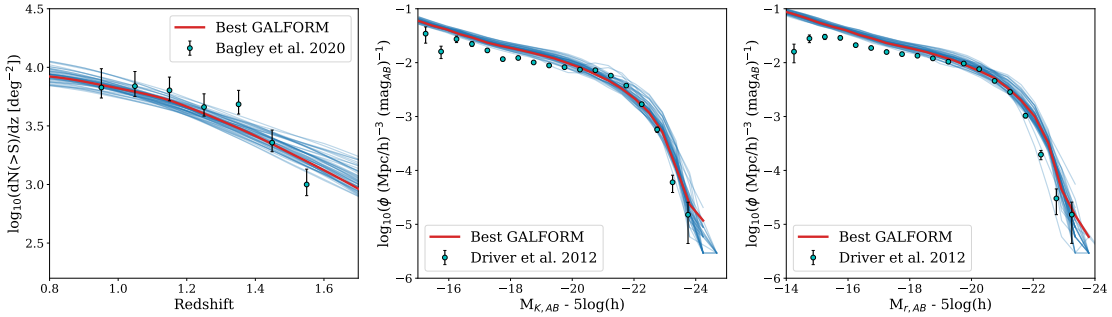


Figure 8.13: The GALFORM evaluations of the best-fitting parameters found with 100 MCMC chains, each 7,500 samples in length, using the constraint weightings described in the text. Here we plot a sample of the best 50 runs, as measured by weighted MAE (Eqn. 8.5). The red line indicates the parameter set with the lowest weighted MAE. The remaining 49 runs are plotted in blue. The data described in §8.4.2 is shown in cyan.

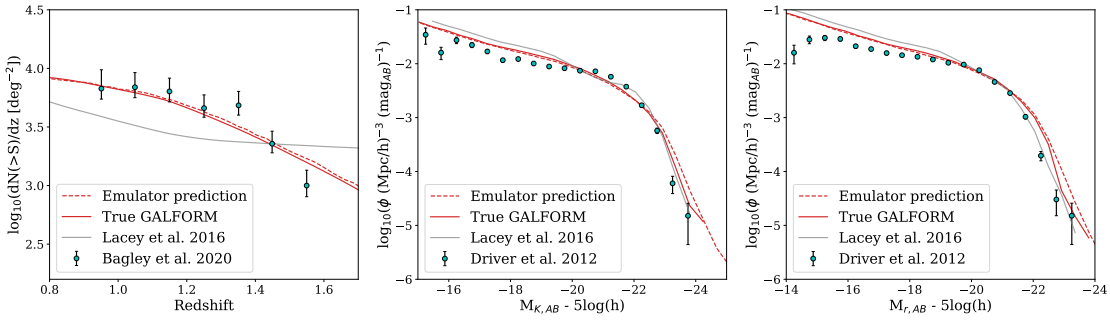


Figure 8.14: The predictions for the calibration data from the lowest MAE parameter set as evaluated by GALFORM (solid red) compared with the equivalent parameters evaluated by our emulator (red dashed) with the calibration data described in §8.4.2. The grey line shows the corresponding predictions made with the Lacey et al. (2016) model.

our parameters do well to replicate the turnover feature. There is greater uncertainty with our fits about the $z = 0$ K-band LF and our lowest weighted MAE parameter set predicts a far smoother turnover compared to the observed data from Driver et al. (2012). In Table 8.2 we show the set of parameters with the lowest weighted MAE to the observational data (corresponding to the red line in Figs. 8.13 and 8.14), the parameter range of the best 50 parameter sets, and compare with the parameters presented in Lacey et al. (2016) (hereafter named Lacey16). We note that the ranges shown in Table 8.2 do not lead to an equivalent fit for any choice of parameters within these ranges as the value of one parameter will constrain the choices of the other parameters, therefore we also present the best-fitting set of parameters as an example. We find that certain parameters, such as $V_{\rm SN,disk}$ and $\gamma_{\rm SN}$ are constrained to a tight range of values, whereas parameters such as $V_{\rm SN,burst}$, $f_{\rm burst}$, and $\tau_{*\rm burst,min}$ can be drawn from a large proportion of the explored range. Although we do see some models that have a large proportion of their parameter spaces sampled, it is not always a uniform distribution. The $\nu_{\rm SF}$ parameter appears to cover a very large range, however, when looking at the corner plot of Fig. 8.12 we see that the majority of the sampling occurs at the high values of $\nu_{\rm SF}$ but there is a small sub-region sampled at ~ 1.0. These bimodal distributions explain the parameter range of $V_{\rm SN,burst}$ presented in Table 8.2. The parameter fSMBH sampling distribution is skewed left which extends the accepted parameter range.

We compare the weighted mean absolute error of our best-fitting model with the Lacey16 model, using the same procedure described in §8.5.3, which is using the same weighting scheme we have been using up to this point. Using this metric, the new model is a better fit for the data. Over the three datasets, the new best fit found in this work achieves a weighted MAE of 0.25 and the weighted MAE of the Lacey16 model is 0.50. The MAE for the Lacey16 model is outside the range of the minimum MAE reached by our 50 best MCMC chains but within the range of the 100 MCMC chains. The improved MAE of the new best-fitting model (and indeed the majority of our MCMC-found models) is mainly due to the large improvements in the fits to the redshift distribution, while the fits of the new model to the K- and r-band LFs are at a similar to that of the Lacey16 model. The new model fits closer to the faint end of the observed LFs, whereas the Lacey16 model is closer to the Driver et al. (2012) datapoints at the bright end. This is particularly true for the r-band. The main source of error with the Lacey16 model is due to the poor fit to the redshift distribution predictions, whereas our model more accurately describes the drop off in number counts beyond $z \sim 1.4$. We can prove this by breaking down the errors per statistic, using

Table 8.2: Results from the 50 best-fitting MCMC parameters (as measured by the weighted MAE in Eqn. 8.5) found using our emulator. In the second column, we present the parameters for the best fit seen in Fig. 8.14, and in the third column, we indicate the parameter ranges of the 50 best runs of the 100 MCMC chains. The Lacey et al. (2016) model parameters are referenced in the final column.

Parameter	This work	Range	Lacey16
$\nu_{\rm SF}$	3.97	$1.00 - 3.97$	0.74
$V_{\rm SN,\,disk}$ [kms^{-1}]	201.30	$195.31 - 313.43$	320
$V_{\rm SN,\,burst}$ [kms^{-1}]	785.64	$34.77 - 798.83$	320
$\gamma_{\rm SN}$	3.98	$3.83 - 4.00$	3.40
$\alpha_{\rm ret}$	0.27	$0.21 - 1.70$	1.00
$\rm F_{stab}$	0.85	$0.77 - 1.18$	0.90
$f_{\rm ellip}$	0.22	$0.21 - 0.50$	0.30
$f_{\rm burst}$	0.083	$0.012 - 0.296$	0.05
$\tau^*_{\rm\,burst,min}$ [Gyr]	0.032	$0.012 - 0.192$	0.10
$f_{\rm SMBH}$	0.039	$0.013 - 0.048$	0.005
$\alpha_{\rm cool}$	0.79	$0.41 - 2.61$	0.80

a standard MAE error calculation (Eqn. 8.2). Our fit to the redshift distribution data of Bagley et al. (2020) is statistically more improved when compared to the fit by Lacey16; we can fit with an MAE of 0.09 whereas the previous model is far worse with a score of 0.26. This is not a surprise when observing the fits by eye as the Lacey16 model fails to replicate the trend of the data set very clearly. This is likely due to the Lacey16 matching to the redshift distribution of submillimeter galaxies instead of Hα emission line galaxies, both are star-forming galaxies, however, the submillimeter sources tend to reside at higher redshifts than emission line galaxies seen with Euclid. Although by eye the fits to the K-band luminosity functions are similar between our new model and the Lacey16 model, the MAE calculations indicate that our new model fits better to the Driver et al. (2012) data than the Lacey16 model. Our model achieves an MAE score of 0.17, whereas the Lacey16 model achieves 0.20. This is likely due to closer fits at the faint end contributing to a higher proportion of the MAE score. The fitting of the bright end is very similar but the Lacey16 model has a much sharper turnover of its luminosity function. We can break down the K-band LF MAE calculation further by only focusing on the bright half of the observable data points. As we have mentioned the Lacey16 model has a closer fit to the observable data points at the bright end as measured by eye, this is confirmed as the MAE of the bright half for the Lacey16 model is 0.11 and for our new model is 0.14. Finally, our model performs slightly better statistically than the Lacey16 model when predicting the r-band LF, scoring an MAE of 0.23 vs 0.25 for the Lacey16 model. This is likely for similar reasons as the K-band, where our model predicts closer to the Driver et al. (2012) data at the faint end. It is clear that the Lacey16 model is slightly better at predicting the luminosity function from the exponential break to the bright end as our model over-predicts, we achieve a lower MAE simply because there are fewer data points at the bright end contributing to the MAE calculation. If we focus only on the bright half of the luminosity function the Lacey16 model is a closer fit to the observed data compared to our model, with an MAE of 0.10 vs 0.16 for our model. Even so, our model still provides an adequate estimation of the luminosity functions as we are able to reasonably estimate the L* break in the data.

Due to the tensions between the statistics, better fits to the redshift distribution data would come

at the expense of more severe over-predictions for the bright-end luminosity functions as previously discussed, and as shown by the lines when increasing the weighting in Fig. 8.11. Similarly, if we try to improve the fits to the LFs, we cause a significant overestimation of the high redshift number counts.

8.5.3.1 Number counts predictions for *Euclid*

Galaxies emitting Hα(+N[II]) are the main target for the dark energy science conducted by *Euclid*, tracing large-scale structures at $z \sim 1-2$. Therefore we consider the number of emission line galaxies that meet the selection criteria for the *Euclid* wide survey. Satisfied with the best fitting parameters from the MCMC using our emulator as evaluated on GALFORM, we can use these models to predict the number of galaxies seen by a *Euclid*-like survey. The cumulative number counts are shown in Fig. 8.15, along with the recent WISP+3D-HST data from Bagley et al. (2020) in the redshift range $0.9 \leq z \leq 1.6$. The corrected number counts from Bagley et al. (2020) for the WISP+3D-HST data is $3266^{+157.7}_{-174.8}$ Hα+N[II] emitters deg^{-2}. Our models predict the galaxy density in the redshift range $0.9 < z < 1.8$, matching that of the *Euclid* wide survey. From our 50 models, the spread in emission-line number counts estimates for galaxies with a flux greater than the *Euclid* limit ($f \geq 2 \times 10^{-16}$ erg^{-1} s^{-1} cm^{-2}) is 2962-4331 deg^{-2}, with our best model to the calibration data outputting a number count of 3462.5 deg^{-2}, corresponding to ~ 52 million sources mapped by *Euclid's* 15,000 deg^2 wide-field survey. Our best-fitting model comfortably lies within the accepted limits of the Bagley et al. (2020) Hα+N[II] number counts. The number counts range between the 10th and 90th percentiles of the distribution of predicted counts is 3158-3952 deg^{-2} which mostly fits within the accepted limits of the observations, with our models tending towards a small over predictions. Looking at the cumulative number counts curve in Fig. 8.15 we see the majority of our models fitting well with the observed data, with our best model following the trend closely. We compare our number count predictions with those of Pozzetti et al. (2016) who empirically fit luminosity functions to several surveys, HiZELS, WISP and HST+NICMOS. We briefly outline the work by Pozzetti et al. (2016) in §6.4. Covering the redshift range $0.9 < z < 1.8$ to a flux limit of 2×10^{-16} erg^{-1} s^{-1} cm^{-2}, they expect about 2000-4800 Hα emitters deg^{-2}, therefore estimating in total 30-72 million sources will be mapped by *Euclid*. It is worth noting that the Pozzetti et al. (2016) predictions are in terms of observed Hα flux, i.e. they include intrinsic dust extinction in the Hα line luminosity and are corrected for [NII] contamination whereas our results blend Hα+N[II] to match the results of Bagley et al. (2020). At the resolution of *Euclid*, these two lines will be partially blended, meaning the Pozzetti et al. (2016) results are somewhat conservative.

In fig 7 of Bagley et al. (2020) (part of which is reproduced in Fig. 6.1) we see the observed cumulative number counts along with fits from various models including the three models from Pozzetti et al. (2016). For the purposes of this comparison, Bagley et al. (2020) has converted the Hα counts from the three Pozzetti et al. (2016) empirical models to Hα+[NII] counts using a fixed [NII]/Hα line ratio: Hα=0.71 (Hα+[NII]), the same conversion as used in Section5 of Pozzetti et al. (2016) while comparing the model counts to observations. Out of their three models, the only model that fits the $0.9 \leq z \leq 1.6$ observations well is Model 3 which shows a similar fit to our best fitting model shown in Fig. 8.15. fig.7 from Bagley et al. (2020) also shows the redshift distribution predictions from Pozzetti et al. (2016), where once again Model 3 is the best fit to the

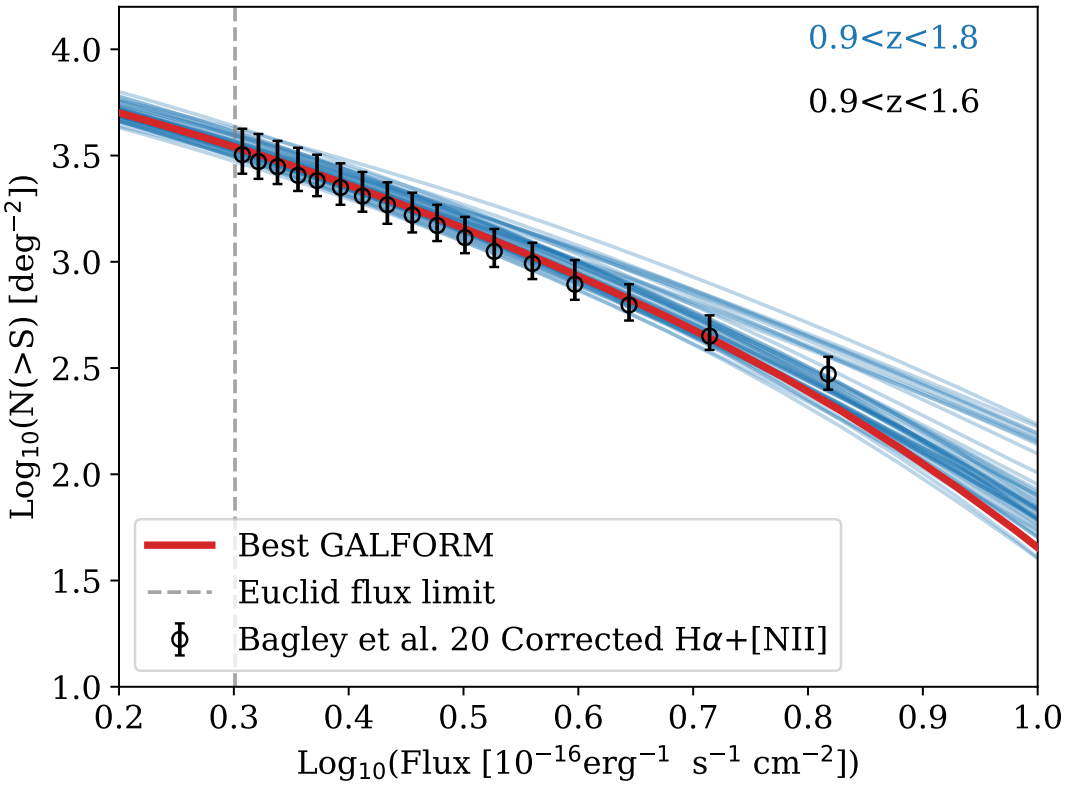


Figure 8.15: Number counts predictions from our 50 best MCMC parameters for galaxies between $0.9 < z < 1.8$ (blue lines), where we have highlighted our best set of parameters as evaluated by GALFORM in red. We plot this against the Bagley et al. (2020) $0.9 < z < 1.6$ number counts (black points). The Euclid flux limit at $f \geq 2 \times 10^{-16}$ erg^{-1} s^{-1} cm^{-2} is indicated as a vertical dashed grey line.

observed redshift distribution. However, the fits are only good for the first five data points before the drop off in counts observed $z \sim 1.4$. As seen in Fig. 8.14, our best model is a better fit to the observed redshift distribution data points as we represent more closely the trends beyond redshift 1.4.

8.5.3.2 Galaxy bias evolution predictions for *Euclid*

Complementing the number counts predictions for a *Euclid*-like survey, we can also use GALFORM to predict the average or effective clustering bias as a function of redshift. The clustering bias is a direct input into the calculation of the effective volume of a galaxy survey, which quantifies the statistical utility of the sample. In hierarchical clustering models with Gaussian distributed primordial fluctuations, the clustering of high peaks in the density field (which corresponds to rare and high-mass dark matter halos) can be described as an amplified version of the mass clustering (Kaiser, 1984). The galaxy bias is computed by taking the ratio between the correlation functions of galaxies and dark matter.

In order to measure the linear bias, we do not need to compute the non-linear two-point correlation function ($\xi(r)$) of dark matter directly. Instead, we will use the bias-halo mass relation, which has been predicted using theoretical models (as in Kaiser 1984) and is well measured from simulations, which allows for empirical extensions to the simple theoretical models (e.g. Tinker

et al. 2010). We choose to omit measurements of the correlation function due to time-saving reasons. To get an accurate measurement of the correlation function we would need to run all 1024 subvolumes to get the full galaxy catalogue. Although it is possible to randomly sub-sample the full galaxy catalogue and still get similar results for the bias (which is equivalent to running GALFORM on a subset of the simulation subvolume files), this would introduce more noise on scales affected by the shot noise. Hence, we would require more than the number of subvolumes we have been using so far, we would require a significant fraction of the full sample. On top of this, the calculation of $\xi(r)$ is a lengthy process. We instead choose to calculate the asymptotic effective bias in real space using the halo mass directly, calculating the effective bias using the COLOSSUS package (Diemer, 2018) using the numerically calibrated bias model from Tinker et al. (2010) for the bias-mass fit, and a virial mass definition. The Tinker et al. (2010) model was calibrated for a range of overdensities concerning the mean density of the universe.

In Fig. 8.16 we plot the results for the average effective linear bias ($b_{\rm eff}$) for individual redshift bins as a function of redshift. We average across every halo for each redshift bin simulated by the GALFORM code, filtering to only include galaxies which have luminosities above the *Euclid* flux limit. We present the results from our 50 best MCMC models as blue lines, and the effective bias of our best model described previously as a red line. We see a tight relation between our 50 best models up to redshift ~ 1.4 where we see some noise between our models. This is because, at higher redshifts, the number of galaxies seen by *Euclid* reduces, causing greater noise in the average bias calculations. The dashed and dotted grey lines show the Merson et al. (2019) models of the linear bias evolution with a WISP-calibrated lightcone and HiZELS-calibrated lightcone respectively. Clearly, their models show the evolution of $b_{\rm eff}$ as a linear relation $b_{\rm eff} = mz + c$, where z is the redshift of the galaxy sample, m is the gradient, and c is the intercept. Due to the mass-bias model chosen, our effective bias evolves with redshift as we see the ratio increase for larger redshifts. It could be argued that at low redshifts, the function is linear. We also see our model predicts a lower bias up to a redshift of ~ 1.8 than Merson et al., at which point the gradient of most of our models overtakes the fit of Merson et al. (2019). Some of the disparity in the bias predictions between the Merson et al models and our models can be traced to differences in the number of galaxies predicted. A difference in the number of predicted Hα ELGs per deg^2 and its dependence on redshift alters the number of galaxies and haloes within the model to calculate the bias. Unfortunately, Merson et al. (2019) does not explicitly reveal their number counts predictions for a *Euclid*-like survey. Merson et al also calibrates their dust extinction specifically to Hα luminosity functions compared to the integrated models involved with the GALFORM code. Modifying the dust extinction changes the distribution of galaxies between halos, hence changing the number of galaxies that are observed within the sample. Finally, differences in the choice of cosmologies (Merson et al uses a cosmology identical to the *Millenium Simulation* of Springel (2005)) and bias-halo mass relations will cause slight discrepancies. Our predicts are instead more in line with those predicted by Orsi et al. (2010) (Fig. 11)with their Hα sample defined by a limiting flux of log($F_{\rm H\alpha}$ erg s^{-1} cm^{-2}])>-16 and $EW_{\rm obs}$>100Å. A lower bias equates to a lower galaxy power spectrum, leading to larger errors in the clustering measurements (see §6.1).

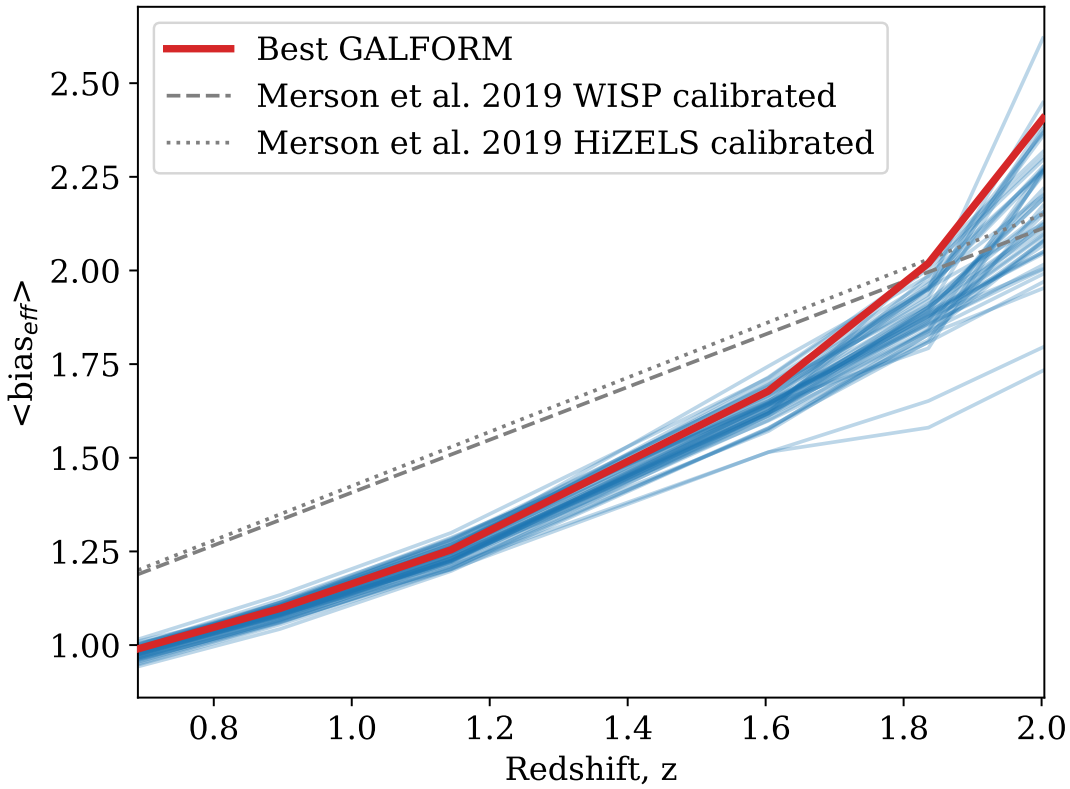


Figure 8.16: The effective clustering bias of our 50 best models as a function of redshift, evaluated using GALFORM and the Colossus routines for computing bias and a function of host halo mass. We have highlighted our best-fitting model to the calibration dataset as a red line. We also plot the bias from Merson et al. (2019) when adopting a WISP-calibrated lightcone (grey dashed line) and a HiZELS-calibrated lightcone (grey dotted line). These authors fitted straight lines to their measured bias-redshift curves.

8.5.3.3 Comparison to older calibration datasets

We have shown that our best-fitting model gives a good reproduction of the Driver et al. (2012) K- and r-band luminosity functions. Therefore we have expanded the comparison datasets to include an older K-band dataset from Cole et al. (2001) which was used in the calibration of many of the older GALFORM variants. In Fig. 8.17 we plot our best GALFORM model $z = 0$ K-band LF, found using our emulator-based MCMC calibrated on the Driver et al. (2012), and compared this with the Cole et al. (2001) K-band LF data. We also plot the Driver et al. (2012) K-band LF data for comparison. We see that the Cole et al. (2001) and Driver et al. (2012) data reasonably well, particularly for bright galaxies. The consistency between the new local calibration data and the old calibration data indicates that the two observational LFs are consistent with one another. Therefore, our GALFORM prediction agrees as well with the Cole et al. (2001) data as it does for the Driver et al. (2012) data, up to faint galaxies where the Cole et al. (2001) data has lower number counts and is noisier. The Cole et al. (2001) LF estimate overlaps mainly with the brighter end of the Driver et al. (2012) data (as expected given the greater depth of the GAMA survey compared with the 2dFGRS and 2MASS data used in the analysis by Cole et al.), and as we have seen in our previous analyses, the weighting scheme compromises our new model at the bright end. Therefore, the new model achieves poorer fits at the bright end when compared to the Lacey16 model for the

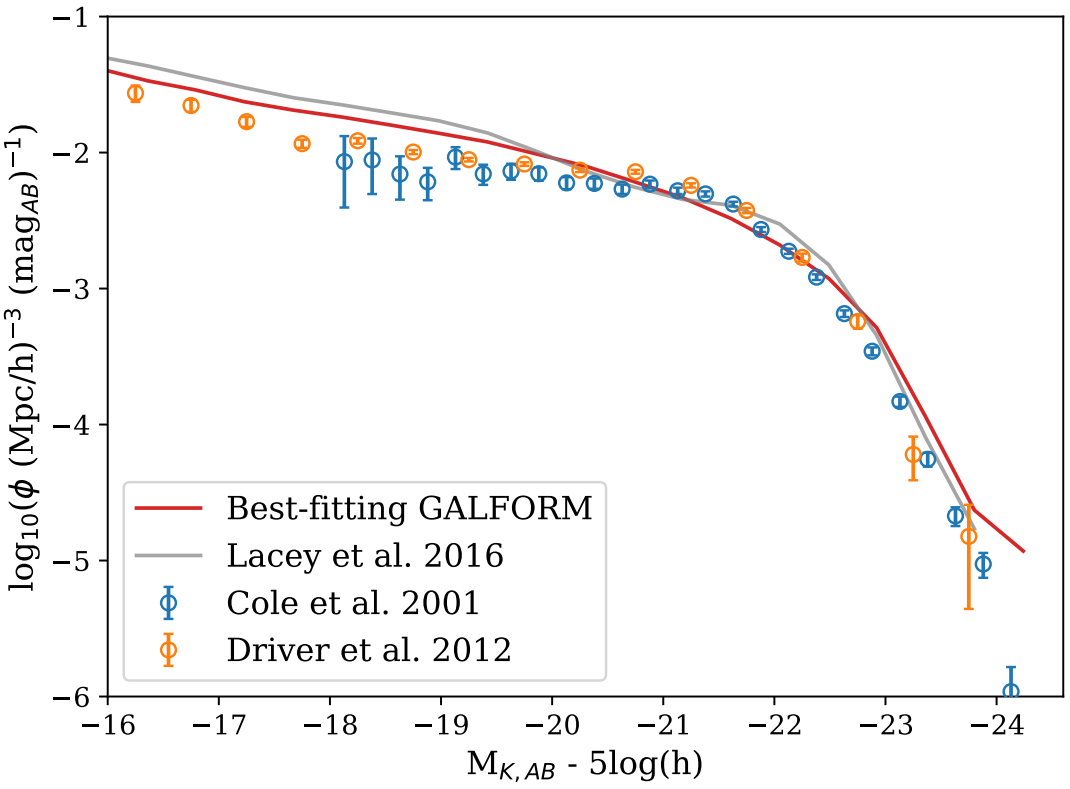


Figure 8.17: The prediction of the new model GALFORM variant introduced here (red line) for the $z = 0$ K-band LF compared with the Driver et al. (2012) (orange) and Cole et al. (2001) (blue) data sets. We also plot a previous GALFORM model by Lacey et al. (2016) (grey line), which was calibrated by hand to a collection of datasets including the Cole et al. LF. We calibrate our model to the Driver et al. (2012) data.

Cole et al. (2001) calibration data also. Our model across all data points scored an MAE of 0.27, this is worse than the Lacey16 model which achieves an MAE of 0.23. Our model scores worse for similar reasons as previous analysis on the Driver et al. (2012) data; over-prediction at the very bright end, and a much shallower turn-over. Nevertheless, our model is still a good approximation to the Cole et al. (2001) data.

8.6 Discussion and Conclusions

In this chapter, we have presented a method for efficiently exploring and calibrating the GALFORM semi-analytical model across a wide range of outputs. We have used data at intermediate redshifts, as well as local data in the model calibration. As a consequence, we have discovered underlying tensions between the three statistics as we are unable to find equally good fits to the Hα redshift distribution, and $z = 0$ K- and r-band luminosity function constraints when all data sets are equally considered in our MCMC (as seen in Fig. 8.11). The weight given to the redshift distribution constraint was increased, moving to a different region of parameter space which modified the fits to the K- and r-band LFs, leading to discrepancies between our model and the observational datasets. For example, again in Fig. 8.11, we see that increasing the weighting towards the redshift distribution degrades the LF fits somewhat, leading to over-prediction at the bright ends. This

could suggest the drop off in number counts at high redshifts for the Bagley et al. (2020) data, in reality, is less sharp compared to what is observed as we find better fits to local LFs when the slope of the redshift distribution is shallow. Another suggestion is the treatment of higher redshift number counts in GALFORM needs to be revised. However, the struggle to find a set of parameters that reproduce all of the constraints effectively (with equal weighting) speaks more to the data from Bagley et al. (2020) and Driver et al. (2012) than the GALFORM model itself.

The emulation method presented here takes inspiration from previous explorations by Elliott et al. (2021). we have focused on maximizing the accuracy of our GALFORM emulator across the whole parameter space defined. We aim to build an emulator which allows us to explore calibration datasets relevant to current and upcoming galaxy surveys. As shown in Fig. 8.8, our emulator performs reasonably well for the constraint from the redshift distribution of ELGs. From Fig. 8.10 we have shown that the performance of the emulator scales with the number of training samples it is exposed to; therefore we can improve our emulator to better predict the redshift distribution constraint by increasing the number of training samples. This increases the computational expense of the process, so another solution could be to reduce the parameter space using the knowledge gained from this study or to use a system of weights to alter the contributions from the calibration datasets. We can produce high-accuracy parameter estimates for fits to datasets by increasing the training samples or using 'zoom-in' training samples as in previous work (e.g. Bower et al. 2010) to focus on a more concentrated region of parameter space which is thought to give acceptable fits to the containing datasets. This method would come at the cost of human intervention where as increasing the number of training samples to create a more accurate emulator would not. Our method described is more reproducible allowing for more uses, whereas Bower et al. (2010) is more subjective to the underlying physics.

In §8.5.3 we calibrated our model to the three sets of observational data under consideration (explained in §8.4.2). However, we did not consider the observational error bars during the model exploration. Instead, we used an absolute error metric to compare the distance between the emulator and the full model calculations. Hence, it is difficult to provide meaningful uncertainty around our estimations of the best-fitting parameters quoted in Table 8.2. Previous calibration efforts for semi-analytical models make trade-offs between certain observational constraints resulting in a poorly defined best-fitting model. We have attempted to reproduce and clarify this process automatically using heuristic weights on the constraints. There is scope in the future for a more robust calibration analysis with an improvement to the treatment of observational errors.

Similarly, we have not considered the uncertainties associated with our emulator. Therefore our approach could be extended to include a more robust measure of our emulator's uncertainty in reproducing GALFORM outputs. There are two types of uncertainty to account for when emulating a set of model outputs, the uncertainty with the emulator's parameters (that is the weights of the neural network), and the uncertainty inherent in the data generation process (for example, the sampling noise on the GALFORM outputs). The network hyperparameter error space was explored using a trial-and-error process to justify the choice of our network architecture. We further reduce uncertainties relating to our emulator's weights by averaging several individual network estimates within our ensemble. When averaging over the estimations from the individual networks, we may be rejecting regions of the GALFORM parameter space which may contain reasonable fits to the data as the average may be away from the best fit. However, regions in the parameter space that

are most difficult for our emulator to model are regions that GALFORM outputs that are 'unusual' or 'undesirable' (for example, LFs without a clear exponential break, seen in the bottom panel of Fig. 8.8) which are unlikely to be good matches to the observations. It would be ideal for our emulator to return an estimate of its prediction uncertainty, calculated from the emulator's weights and uncertainty inherent in the data-generation process. Despite GALFORM being a deterministic code, we are still limited by the noise associated with sampling from a fraction of the total volume of the PMILL simulation. Approaches exist that incorporate the uncertainties discussed into a deep learning framework, for example, Bayesian neural networks (Bishop, 1997) which incorporate epistemic and aleatoric uncertainty into a deep learning framework, and deep kernel learning approaches where a deep neural network transforms the input to the kernel of a Gaussian process regression (Wilson et al., 2016; Patacchiola et al., 2020). These may be a promising line of inquiry to combine the strengths of various processes to outperform current basic methods such as plain Gaussian process models and deep neural networks, as well as provide more robust uncertainty estimates and analysis.

In Fig. 8.10 we showed that we can improve the performance of our emulator by as much as 12% by averaging over five neural networks, instead of using just one. There is a rich avenue of ensembling techniques that could be beneficial in these tasks. Our method used a simple average of the five network predictions, but if a collection of machine learning algorithms can give better fits to certain examples than others (i.e. errors that are not strongly correlated), it may be possible to use a more sophisticated approach to combine the respective advantage of a variety of different algorithms (see e.g. Wolpert, 1992; Opitz & Maclin, 1999; Sagi & Rokach, 2018; Ganaie et al., 2022).

We trained an ensemble of deep learning algorithms to approximate the full GALFORM model using 2999 evaluations of GALFORM (spread across training, validation and testing sets). We use our emulator to explore the GALFORM parameter space and to calibrate the model parameters to a set of three observations. Typical, the exploration of a model parameter space and the determination of a best-fitting set of parameters traditionally requires more than 3000 explicit full calculations of the semi-analytical model. Given the number of parameters and their ranges, our emulator can be considered accurate, particularly in the regions of GALFORM parameter space where the full model outputs closely match the observed Universe.

We have discovered that the majority of variance in the output of our model is due to a few key parameters, which leads to tensions when trying to calibrate to multiple observational datasets. The tensions between the observable datasets were explored, using our MCMC algorithm to fit the emulator output to the constraints, eventually finding the weighting scheme for a global fit to the observations. With this, our techniques find a set of parameters which provides an improved fit to the redshift distribution data as compared with an earlier version of a GALFORM model presented in Lacey et al. (2016). We go further by producing number count predictions for a Euclid-like survey using our best model, improving on previous empirical models by Pozzetti et al. (2016) by using more recent and complete datasets from Bagley et al. (2020). For a flux limit of 2×10^{-16} erg^{-1} s^{-1} cm^{-2} between the redshift range $0.9 < z < 1.8$, our 50 best models predict 2962-4331 Hα emission-line sources deg^{-2}, with 3158-3952 sources deg^{-2} between the 10th and 90th percentile. Our best-fitting model estimates 3462.5 sources deg^{-2}, which is comparable to the Bagley et al. (2020) observation. The predictions we produce for the number of galaxies estimated

to be seen from the Euclid wide field are more constrained than previous models and are better in line with the observed number counts of WISP+3D-HST.

Our bias predictions follow a similar trend to that of Merson et al. (2019), but with some differences in detail, with Merson et al. fitting the bias to a linear function, whereas we find a more non-linear relation at high redshifts. This could be down to the choices of dust extinction models, bias-halo mass relation models and cosmologies, as well as differences in the number counts predictions for a *Euclid*-like survey. Merson et al. (2019) does not identify some of these choices or values, meaning we can only speculate. A lower bias measurement equates to larger uncertainties with a clustering measurement. At low redshifts, we see a reduced bias estimate, compared to previous work. However, at the redshift where *Euclid* can see Hα ELGs we see comparable bias calculations to previous works (e.g. Orsi et al., 2010; Merson et al., 2019; Zhai et al., 2021).

The emulation approach is intended to calibrate `GALFORM` using the observed galaxy redshift distribution (plus local luminosity function data) to generate a mock catalogue for the *Euclid* survey (Laureijs et al., 2011; Racca et al., 2016) to predict the abundance and clustering of galaxies. This requires model outputs over a large number of redshifts (see the lightcone method described in chapter 7), which makes running `GALFORM` computationally expensive. Calibrating the model across a large redshift range would be immensely more expensive for direct MCMC methods, and is a very difficult task to achieve by eye. Therefore, this method reduces the required number of model evaluations as much as possible. We have demonstrated that the number of runs needed to achieve good emulator accuracy is a fraction required for manual direct calibration methods, and we can emulate over a wide range of outputs.

We believe the methods presented can optimise the process of accurately exploring and calibrating semi-analytical models intuitively and inexpensively, acting as an alternative to other emulation techniques in the literature. This method is an invaluable tool for the rapid assessment of the implications of changes to the underlying model.

Chapter 9

Final thoughts on the academic project

Throughout this academic part of the thesis, we have introduced a variety of novel techniques as we aimed to predict up-to-date Hα number counts and clustering bias calculations for the *Euclid* survey. The research and methods presented are not constrained to the tasks outlined here, rather they provide opportunities for gains in the wider field of astronomy.

We have outlined a more rapid method for generating a large number of GALFORM models compared to the literature standard of generating a lightcone. Instead of running every snapshot redshift between $z = 0$ and the target redshift, it is possible to run only a reduced sample of snapshots and interpolate between the luminosity functions to produce accurate counts. Similarly, we do not need to run the full N-body simulation volume as sampling only 1% of the volume still outputs effective outputs with acceptable noise. We have shown how this method can be used to output cumulative number counts and redshift distribution results. However, this method is not limited to these outputs. It is possible to tailor the interpolative method to generate any output from the GALFORM model, and could even be introduced into alternative SAMs. GALFORM already has the capability to interpolate between redshift snapshots to output a range of statistics, including galaxy morphology, mass, gas fraction and star formation rates, to name a few. The benefits of this method are clear, as long as an appropriate number of redshift snapshots and simulation subvolumes is used, the computational time and power required to run a SAM can be greatly reduced.

Along the same lines, the benefits from the advancements we have made in GALFORM model emulation and optimisation are not limited to the contexts presented in this thesis. We have proved the effectiveness of emulating specific outputs of a SAM using a relatively simple ensemble of deep networks, predicting the redshift distribution and local Universe luminosity functions to a high degree of accuracy. The pipeline presented here can be modified with ease to fit any emulation task of a SAM such as GALFORM. The requirements for an emulator to predict accurate outputs come down to using an appropriate number of training samples which is correlated to the number of input parameters and their ranges, a large number of parameters with large ranges leads to more training samples required to cover the parameter space. Evidence from our work and previous works (such as Elliott et al. 2021) shows the number of statistics the emulator predicts has little effect on performance. Therefore this technique could be used to predict a greater variety of SAM outputs than the ones presented here. As mentioned in chapter 8, there is promising scope in the future to

use more advanced methods of ensembling. The benefits of using an emulation method such as this one are clear, tremendously reducing the time taken to generate SAM samples compared to the full traditional methods. Combining emulation with MCMC leads to rapid sampling of a parameter space for the calibration of a model. If we wish to improve our work, we would introduce better quantification of uncertainty with our emulator predictions and MCMC samples.

We believe the work presented in this thesis will lead to rapid advancements in the field of computational cosmology, as the utilisation of machine learning becomes more prominent in the academic realm, there is less dependence on running full cosmology simulations to obtain predictions of our Universe. Instead, the same outcome could be achieved in a matter of seconds. The downfall of this method is machine learning is simply a mapping function from an input to an output with no understanding of the physical properties that achieve the output. Hence, using machine learning in conjunction with full SAMs is key to uncovering improved fits to observed datasets, as well as uncovering hidden tensions between constraints.

Part III

Bibliography and Appendix

Bibliography

Abadi M., et al., 2016, pp 265–283

Alzubaidi L., et al., 2021, Journal of Big Data, 8, 53

Angulo R. E., Baugh C. M., Frenk C. S., Lacey C. G., 2008, MNRAS, 383, 755

Angulo R. E., White S. D. M., Springel V., Henriques B., 2014, MNRAS, 442, 2131

Arulkumaran K., Deisenroth M. P., Brundage M., Bharath A. A., 2017, arXiv preprint arXiv:1708.05866

Atek H., et al., 2010, The Astrophysical Journal, 723, 104

Atek H., et al., 2011, The Astrophysical Journal, 743, 121

Bagley M. B., et al., 2020, The Astrophysical Journal, 897, 98

Bailey V. P., et al., 2023, Techniques and Instrumentation for Detection of Exoplanets XI, 12680, 283

Baron D., 2019, arXiv preprint arXiv:1904.07248

Bassett B., Hlozek R., 2010, Dark energy: observational and theoretical approaches, p. 246

Baugh C. M., 2006, Reports on Progress in Physics, 69, 3101

Baugh C., Lacey C. G., Frenk C., Granato G., Silva L., Bressan A., Benson A., Cole S., 2005, Monthly Notices of the Royal Astronomical Society, 356, 1191

Baugh C., et al., 2019, Monthly Notices of the Royal Astronomical Society, 483, 4922

Benson A. J., 2010, Phys. Rep., 495, 33

Benson A. J., 2012, New Astronomy, 17, 175

Benson A. J., Bower R., 2010, Monthly Notices of the Royal Astronomical Society, 405, 1573

Benson A., Baugh C., Cole S., Frenk C., Lacey C., 2000, Monthly Notices of the Royal Astronomical Society, 316, 107

Bernstein G. M., Cai Y.-C., 2011, Monthly Notices of the Royal Astronomical Society, 416, 3009

Bigiel F., et al., 2011, The Astrophysical Journal Letters, 730, L13

Bishop C. M., 1997, Journal of the Brazilian Computer Society, 4, 61

Blitz L., Rosolowsky E., 2006, The Astrophysical Journal, 650, 933

Bower R. G., Benson A., Malbon R., Helly J., Frenk C., Baugh C., Cole S., Lacey C. G., 2006, Monthly Notices of the Royal Astronomical Society, 370, 645

Bower R. G., Vernon I., Goldstein M., Benson A., Lacey C. G., Baugh C. M., Cole S., Frenk C., 2010, Monthly Notices of the Royal Astronomical Society, 407, 2017

Boylan-Kolchin M., Springel V., White S. D., Jenkins A., Lemson G., 2009, Monthly Notices of the Royal Astronomical Society, 398, 1150

Brammer G. B., et al., 2012, The Astrophysical Journal Supplement Series, 200, 13

Campbell D. J., et al., 2015, Monthly Notices of the Royal Astronomical Society, 452, 852

Casali M., et al., 2007, Astronomy & Astrophysics, 467, 777

Christodoulou D. M., Shlosman I., Tohline J. E., 1994, arXiv preprint astro-ph/9411031

Chuang C.-H., et al., 2019, Monthly Notices of the Royal Astronomical Society, 487, 48

Clevert D.-A., Unterthiner T., Hochreiter S., 2015, arXiv preprint arXiv:1511.07289

Colbert J. W., et al., 2013, The Astrophysical Journal, 779, 34

Cole S., Lacey C. G., Baugh C. M., Frenk C. S., 2000, Monthly Notices of the Royal Astronomical Society, 319, 168

Cole S., et al., 2001, Monthly Notices of the Royal Astronomical Society, 326, 255

Cole S., et al., 2005, Monthly Notices of the Royal Astronomical Society, 362, 505

Combes F., Debbasch F., Friedli D., Pfenniger D., 1990, Astronomy and Astrophysics (ISSN 0004-6361), vol. 233, no. 1, July 1990, p. 82-95. Research supported by the Universite de Geneve and SNSF., 233, 82

Conroy C., 2013, ARA&A, 51, 393

Cora S. A., 2006, Monthly Notices of the Royal Astronomical Society, 368, 1540

Cranmer M. D., Xu R., Battaglia P., Ho S., 2019, arXiv preprint arXiv:1909.05862

Debattista V. P., Mayer L., Carollo C. M., Moore B., Wadsley J., Quinn T., 2006, The Astrophysical Journal, 645, 209

DeepMind 2019, https://deepmind.google/discover/blog/machine-learning-can-boost-the-value-of-wind-energy/

Diemer B., 2018, ApJS, 239, 35

Driver S. P., et al., 2012, Monthly Notices of the Royal Astronomical Society, 427, 3244

Dubey S. R., Singh S. K., Chaudhuri B. B., 2022, Neurocomputing

Efstathiou G., Lake G., Negroponte J., 1982, Monthly Notices of the Royal Astronomical Society, 199, 1069

Eisenstein D. J., et al., 2005, The Astrophysical Journal, 633, 560

Elliott E. J., Baugh C. M., Lacey C. G., 2021, MNRAS, 506, 4011

Euclid Collaboration et al., 2022, A&A, 662, A112

Faisst A. L., Capak P. L., Emami N., Tacchella S., Larson K. L., 2019, The Astrophysical Journal, 884, 133

Fanidakis N., Baugh C., Benson A., Bower R., Cole S., Done C., Frenk C., 2011, Monthly Notices of the Royal Astronomical Society, 410, 53

Feldman H. A., Kaiser N., Peacock J. A., 1993, arXiv preprint astro-ph/9304022

Font A. S., et al., 2008, Monthly Notices of the Royal Astronomical Society, 389, 1619

Fu J., Guo Q., Kauffmann G., Krumholz M. R., 2010, Monthly Notices of the Royal Astronomical Society, 409, 515

Gallego J., Zamorano J., Aragón-Salamanca A., Rego M., 1995, The Astrophysical Journal, 455, L1

Ganaie M. A., Hu M., Malik A., Tanveer M., Suganthan P., 2022, Engineering Applications of Artificial Intelligence, 115, 105151

Gao D., Zhang Y.-X., Zhao Y.-H., 2009, Research in Astronomy and Astrophysics, 9, 220

Gargiulo I. D., et al., 2015, Monthly Notices of the Royal Astronomical Society, 446, 3820

Geach J. E., Smail I., Best P., Kurk J., Casali M., Ivison R., Coppin K., 2008, Monthly Notices of the Royal Astronomical Society, 388, 1473

Geach J. E., et al., 2010, Monthly Notices of the Royal Astronomical Society, 402, 1330

Glorot X., Bengio Y., 2010, in Proceedings of the thirteenth international conference on artificial intelligence and statistics. pp 249–256

Goldstein M., Wooff D., 2007, Bayes linear statistics: Theory and methods. John Wiley & Sons

Goldstine H., Goldstine A., 1996, IEEE Annals of the History of Computing, 18, 10

Gonzalez-Perez V., Lacey C. G., Baugh C. M., Lagos C., Helly J., Campbell D., Mitchell P. D., 2014, Monthly Notices of the Royal Astronomical Society, 439, 264

Granato G. L., Lacey C., Silva L., Bressan A., Baugh C., Cole S., Frenk C., 2000, The Astrophysical Journal, 542, 710

Green J., et al., 2012, arXiv preprint arXiv:1208.4012

Griffin A. J., Lacey C. G., Gonzalez-Perez V., Lagos C. d. P., Baugh C. M., Fanidakis N., 2019, Monthly Notices of the Royal Astronomical Society, 487, 198

He S., Li Y., Feng Y., Ho S., Ravanbakhsh S., Chen W., Póczos B., 2019, Proceedings of the National Academy of Sciences, 116, 13825

Henriques B. M., Thomas P. A., Oliver S., Roseboom I., 2009, Monthly Notices of the Royal Astronomical Society, 396, 535

Hopkins A. M., Connolly A., Szalay A., 2000, The Astronomical Journal, 120, 2843

Jiang L., Helly J. C., Cole S., Frenk C. S., 2014, Monthly Notices of the Royal Astronomical Society, 440, 2115

Kaiser N., 1984, On the spatial correlations of Abell clusters. Vol. 284

Kampakoglou M., Trotta R., Silk J., 2008, Monthly Notices of the Royal Astronomical Society, 384, 1414

Kennedy J., Eberhart R., 1995, in Proceedings of ICNN'95-international conference on neural networks. pp 1942–1948

Kennicutt Jr R. C., 1998, Annual Review of Astronomy and Astrophysics, 36, 189

Kewley L. J., Nicholls D. C., Sutherland R. S., 2019, Annual Review of Astronomy and Astrophysics, 57, 511

Kingma D. P., Ba J., 2014, arXiv preprint arXiv:1412.6980

Kitzbichler M. G., White S. D., 2007, Monthly Notices of the Royal Astronomical Society, 376, 2

Komatsu E., et al., 2011, ApJS, 192, 18

Lacey C. G., Baugh C. M., Frenk C. S., Benson A. J., 2011, MNRAS, 412, 1828

Lacey C. G., et al., 2016, MNRAS, 462, 3854

Lagos C. d. P., Cora S. A., Padilla N. D., 2008, Monthly Notices of the Royal Astronomical Society, 388, 587

Lagos C. d. P., Lacey C. G., Baugh C. M., Bower R. G., Benson A. J., 2011, Monthly Notices of the Royal Astronomical Society, 416, 1566

Laureijs R., et al., 2011, arXiv preprint arXiv:1110.3193

Laureijs R., et al., 2012, in Space Telescopes and Instrumentation 2012: Optical, Infrared, and Millimeter Wave. pp 329–336

Li Y., Huang C., Ding L., Li Z., Pan Y., Gao X., 2019, Methods, 166, 4

Liu X., et al., 2019, The lancet digital health, 1, e271

Loh W.-L., 1996, The annals of statistics, 24, 2058

Lu Y., Mo H., Weinberg M. D., Katz N., 2011, Monthly Notices of the Royal Astronomical Society, 416, 1949

Lu Y., Mo H., Katz N., Weinberg M. D., 2012, Monthly Notices of the Royal Astronomical Society, 421, 1779

Lu Y., Mo H., Lu Z., Katz N., Weinberg M. D., 2014, Monthly Notices of the Royal Astronomical Society, 443, 1252

Lu L., Shin Y., Su Y., Karniadakis G. E., 2019, arXiv preprint arXiv:1903.06733

Maas A. L., Hannun A. Y., Ng A. Y., et al., 2013, in Proc. icml. p. 3

Manzoni G., et al., 2023, arXiv e-prints, p. arXiv:2311.10469

McCarthy P. J., et al., 1999, The Astrophysical Journal, 520, 548

Mehta V., et al., 2015, The Astrophysical Journal, 811, 141

Merson A. I., et al., 2013, Monthly Notices of the Royal Astronomical Society, 429, 556

Merson A., Wang Y., Benson A., Faisst A., Masters D., Kiessling A., Rhodes J., 2018, Monthly Notices of the Royal Astronomical Society, 474, 177

Merson A., Smith A., Benson A., Wang Y., Baugh C., 2019, MNRAS, 486, 5737

Momcheva I. G., et al., 2016, The Astrophysical Journal Supplement Series, 225, 27

Moore G. E., 1965, Electronics Magazine, 38, 114

Nair V., Hinton G. E., 2010, in Proceedings of the 27th international conference on machine learning (ICML-10). pp 807–814

Norberg P., et al., 2002, Monthly Notices of the Royal Astronomical Society, 336, 907

Ntampaka M., et al., 2019, arXiv preprint arXiv:1902.10159

Opitz D., Maclin R., 1999, Journal of artificial intelligence research, 11, 169

Orsi A., Baugh C., Lacey C., Cimatti A., Wang Y., Zamorani G., 2010, Monthly Notices of the Royal Astronomical Society, 405, 1006

Padilla N. D., Salazar-Albornoz S., Contreras S., Cora S. A., Ruiz A. N., 2014, Monthly Notices of the Royal Astronomical Society, 443, 2801

Patacchiola M., Turner J., Crowley E. J., O'Boyle M., Storkey A. J., 2020, Advances in Neural Information Processing Systems, 33, 16108

Perraudin N., Srivastava A., Lucchi A., Kacprzak T., Hofmann T., Réfrégier A., 2019, Computational Astrophysics and Cosmology, 6, 1

Planck Collaboration et al., 2014, A&A, 571, A16

Planck Collaboration et al., 2020, A&A, 641, A6

Popping G., Somerville R. S., Trager S. C., 2014, Monthly Notices of the Royal Astronomical Society, 442, 2398

Pozzetti L., et al., 2016, arXiv preprint arXiv:1603.01453

Racca G., Laureijs R., Stagnaro L., et al., 2016, Infrared, and Millimeter Wave (SPIE), 9904, 99040O

Ravanbakhsh S., Oliva J., Fromenteau S., Price L., Ho S., Schneider J., Póczos B., 2016, in International Conference on Machine Learning. pp 2407–2416

Reddi S. J., Kale S., Kumar S., 2019, arXiv preprint arXiv:1904.09237

Reyes R., Mandelbaum R., Seljak U., Baldauf T., Gunn J. E., Lombriser L., Smith R. E., 2010, Nature, 464, 256

Robert C. P., Casella G., Robert C. P., Casella G., 2004, Monte Carlo statistical methods, pp 267–320

Rodrigues L. F. S., Vernon I., Bower R. G., 2017, Monthly Notices of the Royal Astronomical Society, 466, 2418

Rosenblatt F., 1958, Psychological Review, 65, 386

Ross P., Maynard K., 2021, Intelligent Buildings International, 13, 159

Ruiz A. N., et al., 2015, The Astrophysical Journal, 801, 139

Rumelhart D. E., Hinton G. E., Williams R. J., 1986, nature, 323, 533

Russell S. J., Norvig P., 2010, Artificial intelligence a modern approach. London

Sagi O., Rokach L., 2018, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8, e1249

Samuel A. L., 1959, IBM Journal of Research and Development, 3, 210

Schaye J., et al., 2015, Monthly Notices of the Royal Astronomical Society, 446, 521

Schechter P., 1976, Astrophysical Journal, Vol. 203, p. 297-306, 203, 297

Schmit C. J., Pritchard J. R., 2018, Monthly Notices of the Royal Astronomical Society, 475, 1213

Scoville N., et al., 2007, The Astrophysical Journal Supplement Series, 172, 1

Seo H.-J., Eisenstein D. J., 2007, The Astrophysical Journal, 665, 14

Sharma K., Alsadoon A., Prasad P., Al-Dala'in T., Nguyen T. Q. V., Pham D. T. H., 2020, Computer methods and programs in biomedicine, 197, 105751

Sheikh H., Prins C., Schrijvers E., 2023, Mission AI: The New System Technology. Springer International Publishing, pp 15–41

Shim H., Colbert J., Teplitz H., Henry A., Malkan M., McCarthy P., Yan L., 2009, The Astrophysical Journal, 696, 785

Simha V., Cole S., 2017, MNRAS, 472, 1392

Skelton R. E., et al., 2014, The Astrophysical Journal Supplement Series, 214, 24

Sobral D., et al., 2009, Monthly Notices of the Royal Astronomical Society, 398, 75

Sobral D., Best P. N., Matsuda Y., Smail I., Geach J. E., Cirasuolo M., 2012, Monthly Notices of the Royal Astronomical Society, 420, 1926

Sobral D., Smail I., Best P. N., Geach J. E., Matsuda Y., Stott J. P., Cirasuolo M., Kurk J., 2013, Monthly Notices of the Royal Astronomical Society, 428, 1128

Spergel D., et al., 2015, arXiv preprint arXiv:1503.03757

Springel V., 2005, Monthly notices of the royal astronomical society, 364, 1105

Springel V., White S. D., Tormen G., Kauffmann G., 2001, Monthly Notices of the Royal Astronomical Society, 328, 726

Statista 2023, https://www.statista.com/statistics/871513/worldwide-data-created/

Sun Y., Wang X., Tang X., 2015, in Proceedings of the IEEE conference on computer vision and pattern recognition. pp 2892–2900

Thompson D., Mannucci F., Beckwith S., 1996, arXiv preprint astro-ph/9610135

Tieleman T., Hinton G., 2012, COURSERA: Neural Networks for Machine Learning, 4, 26

Tinker J. L., Robertson B. E., Kravtsov A. V., Klypin A., Warren M. S., Yepes G., Gottlöber S., 2010, ApJ, 724, 878

Tran P. T., et al., 2019, IEEE Access, 7, 61706

Tresse L., Maddox S., Le Fevre O., Cuby J.-G., 2002, Monthly Notices of the Royal Astronomical Society, 337, 369

Valentino F., et al., 2017, Monthly Notices of the Royal Astronomical Society, 472, 4878

Vernon I., Goldstein M., Bower R. G., 2010, Bayesian analysis., 5, 697

Wang Y., et al., 2022, ApJ, 928, 1

Weiner B., 2009, HST Proposal, p. 11600

White S. D., Rees M. J., 1978, Monthly Notices of the Royal Astronomical Society, 183, 341

Wilson A. G., Hu Z., Salakhutdinov R., Xing E. P., 2016, in Artificial intelligence and statistics. pp 370–378

Wolpert D. H., 1992, Neural networks, 5, 241

Xu B., Wang N., Chen T., Li M., 2015, arXiv preprint arXiv:1505.00853

Xu Q., Zhang M., Gu Z., Pan G., 2019, Neurocomputing, 328, 69

Xu X., et al., 2020, Engineering, 6, 1122

Yu K.-H., Beam A. L., Kohane I. S., 2018, Nature biomedical engineering, 2, 719

Zhai Z., Benson A., Wang Y., Yepes G., Chuang C.-H., 2019, Monthly Notices of the Royal Astronomical Society, 490, 3667

Zhai Z., Wang Y., Benson A., Colbert J., Bagley M., Henry A., Baronchelli I., 2021, arXiv preprint arXiv:2109.12216

Zhang Z., Sabuncu M., 2018, Advances in neural information processing systems, 31

Zhang X., Wang Y., Zhang W., Sun Y., He S., Contardo G., Villaescusa-Navarro F., Ho S., 2019, arXiv preprint arXiv:1902.05965

de Oliveira R. A., Li Y., Villaescusa-Navarro F., Ho S., Spergel D. N., 2020, arXiv preprint arXiv:2012.00240

Colophon

This thesis is based on a template developed by Matthew Townson and Andrew Reeves. It was typeset with LaTeX 2_ε. It was created using the *memoir* package, maintained by Lars Madsen, with the *madsen* chapter style. The font used is Latin Modern, derived from fonts designed by Donald E. Kunith.

Appendix A

Kromek Appendix - redacted