Machine Learning and Galaxy Formation

EDWARD JOHN ELLIOTT

Machine Learning and Galaxy Formation

Edward Elliott

A thesis presented for the degree of
Doctor of Philosophy

Institute for Computational Cosmology
The University of Durham
United Kingdom

July 2023

Machine Learning and Galaxy Formation

Edward Elliott

Abstract

Galaxy formation and evolution involves the interplay of a large number of complex, non-linear processes, many of which act at scales beneath those accessible to even the most modern galaxy formation simulations. Galaxy formation models therefore include parameterised sub-grid processes, which must be calibrated against selected observational constraints. In this thesis, I explore the application of machine learning and optimisation methods to characterize and calibrate a semi-analytic model of galaxy formation, GALFORM. I investigate the application of deep learning to this problem, building an accurate emulator of the full model over a ten dimensional parameter space from just 1000 GALFORM evaluations. I investigate the calibration of GALFORM to a large number of datasets, and investigate tensions between different choices of calibration datasets and the parameters themselves. Next, I present an investigation into the controversial requirement for a top-heavy stellar initial mass function in starbursts in the GALFORM model, which it was argued was necessary for the model to match the constraints from the number counts of sub-millimeter galaxies, their redshift distribution, *and* the local K-band luminosity function. Here, I apply Bayesian Optimisation to search the model parameter space for optimal fits to these datasets, and demonstrate that GALFORM is not capable of reproducing these data simultaneously with a solar neighbourhood IMF, and that the top-heavy IMF alleviates this problem.

Supervisors: Carlton Baugh and Cedric Lacey

Acknowledgements

First, I would like to thank my supervisors Carlton Baugh and Cedric Lacey for their guidance, support and patience over the last few years. Thanks go also to John Helly and Alastair Basden for their help with technical matters. I would also like to thank my parents for their immense patience and unwavering support, and to my brothers for keeping me sane—particularly James during coronavirus lockdown.

I would also like to thank my supervisors during my time at Optiver, Ingi and Kevin, for their support, hospitality and willingness to share their knowledge.

Thank you also to my roster of housemates; Henry, Cameron, Jack, Vicky, Kevin, and to everyone in department who shared stimulating coffee-time chats.

Thanks also to my rock-like school friends, Ted, Rob, Ric, Alex, and Stan, who have been a never-ending source of life, laughter, and love.

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.

Some of the work presented in this thesis has been published in journals and conference proceedings - the relevant publications are listed below.

Publications

Chapter 3 contains material published as *Efficient exploration and calibration of a semi-analytical model of galaxy formation with deep learning.* Elliott E J, Baugh C M, Lacey C G. 2021. MNRAS, 506, 4011-4030.

Chapter 4 is the basis of a paper that is about to be submitted for publication.

I led the writing of the above papers, and was responsible for all of the calculations and made all of the plots. I also led the design of the machine learning solutions.

Parts of the descriptions of the `GALFORM` model from Chapters 3 and 4 have been removed from these chapters and consolidated into one piece of text in Chapter 2, to avoid repetition.

Chapter 5 is based on work carried out during a placement with Optiver, a proprietary trading and market making company based in the Netherlands; this was part of my research training as a student in the Centre for Doctoral Training in Data Intensive Science at Durham University. This project was undertaken in collaboration with another PhD student, Ka W. Kwok. Chapter 5 is my write-up of this work.

List of Figures

List of Tables

Chapter 1

Introduction

Galaxy formation models provide a framework through which we can understand the structure and evolution of the Universe. Galaxies trace the underlying dark matter distribution in an unknown way, which is the result of the interplay of a large number of complex, often nonlinear processes. Due to the inherent difficulty of the modelling, compounded by the range of length and time scales involved, reproducing the properties of the observed galaxy population using physically motivated *ab initio* models of galaxy formation has proven to be challenging. Nevertheless, the current state-of-the-art in galaxy formation models has enjoyed some success in matching some of the general properties of the galaxy population, and has provided key insights into the processes which must be driving the evolution of galaxies (see for example the reviews by Baugh 2006; Benson 2010; Somerville & Davé 2015).

The earliest simulations used a handful of light-bulbs to simulate the gravitational interactions during the collision of two galaxies (Holmberg, 1941). Since then, galaxy formation simulations have advanced to be able to simulate the formation and evolution of tens of thousands of galaxies in representative volumes of the universe (e.g. Lacey et al., 2016; Schaye et al., 2015). These simulations are able to track the evolution of the dark matter and gas, the formation of stars, and the chemical evolution of the stars and gas. These calculations provide important insights into the galaxy formation process through comparison to observation.

1.1 ΛCDM and hierachical structure formation

The prevailing theoretical framework for understanding the formation of large-scale structure and the formation of galaxies in the Universe is the ΛCDM model (Peebles, 1980). The ΛCDM model incorporates two fundamental components; the dark energy that is thought to be responsible for the recent acceleration of the cosmic expansion, and a cold form of dark matter, non-baryonic matter crucial for the gravitational assembly of structure.

The ΛCDM model is supported by decades of observational and theoretical cosmological evidence. Measurements of the Cosmic Microwave Background (CMB) temperature and polarization fluctuations, reflecting the density fluctuations in the early universe, align closely with theoretical predictions of the ΛCDM model (Bond & Efstathiou, 1984; Spergel et al., 2003; Planck Collaboration et al., 2016). The large-scale distribution of galaxies as measured by surveys such as the Sloan Digital Sky Survey (SDSS; York et al. 2000a), is consistent with hierarchical structure formation within a ΛCDM paradigm (Coles & Lucchin, 1995; Percival et al., 2001; Eisenstein et al., 2005; Alam et al., 2017). Gravitational lensing measurements of galaxy clusters and X-ray images of their hot gas have also indicated that there is a large discrepancy between the inferred mass and the observed luminous matter, providing indirect evidence of a large missing matter component (Allen et al., 2011).

Observations of Type 1A supernovae also offer support to the ΛCDM, finding that they are dimmer than would be expected in a matter dominated universe. Inferring cosmological parameters from these supernovae, the expansion of the universe was found to be accelerating, providing evidence for a significant dark energy contribution to the present energy density of the universe (Riess et al., 1998; Perlmutter et al., 1999).

Importantly, within ΛCDM, the dark matter is taken to be cold. Though any candidate dark matter has so far eluded direct detection, to match observations, the dark matter component is required have a low thermal velocity dispersion, leading to a short free streaming length and preservation of density perturbations on small scales (Davis et al., 1985). Lighter dark matter particles with longer free-streaming lengths and higher velocity dispersions (hot or warm dark matter) would smooth-out small scale perturbations (indicated by a steeper fall in the matter power spectrum at shorter wavelengths). A potential candidate for a cold dark matter particle is the Weakly Interacting Massive Particle or WIMP (Peebles, 1982), though as stated, there have been no direct detections of dark matter candidate particles.

In the ΛCDM model, structure emerges through a bottom-up process, where small density fluctuations in the early universe grow under the influence of gravitational instability into progressively larger structures. Over cosmic time, small-scale density perturbations collapse under their self-gravity to form dark matter halos, which subsequently merge and accrete matter to build larger structures. Quantum fluctuations in the early universe are expanded to large scales by inflation, and form the basis for the formation of structure in the Universe. The initial growth of these perturbations, in the linear regime, is well understood analytically, but their later collapse and the process of structure formation is much more complex and non-linear. Nevertheless, this non-linear growth of structure has been investigated and largely understood via N-body simulation (e.g. Springel et al., 2006). These early perturbations collapse to form dark matter halos once their self-gravity overcomes the expansion of the universe. The halos act as the scaffolding for galaxies, as baryonic matter condenses inside the halo.

Gas cools through radiative processes within the dark matter halos, and the cooling gas retains the angular momentum of the halo (which is generated by tidal torques due to the irregular density field around a halo) to form a disc. Once the gas is cold enough, it is able to form stars. These stars follow evolutionary tracks which have been modelled and studied in detail through observation of local stellar populations and populations in nearby galaxies (Conroy, 2013). When larger mass stars die, they inject large amounts of energy into the interstellar medium, ejecting gas out of the galaxy, and playing a crucial role in shaping the observational properties of galaxy populations (e.g White & Frenk, 1991). As dark matter halos merge and form larger structures, the galaxies that reside within them are also subject to mergers which is key in giving rise to the wide array of galaxy shapes and sizes (e.g. Kauffmann, 1996).

Early attempts at building physical models of galaxy formation within this hierarchical model (White & Rees, 1978) found that in low-mass halos too much gas was cooling, and too many stars were being formed to match measurements of e.g. the local K-band luminosity function. Fundamentally, this was due to a discrepancy between the luminosity function of galaxies and the halo mass function. At the low-mass end of the halo mass function, observationally, far fewer stars are formed than would be implied by a naive model based on the abundance of low-mass halos and assuming an equal mass-to-light ratio across halo masses. A model of galaxy formation attempts to link the evolution of galaxies to their dark matter by positing physical explanations for such discrepancies. A supernova feedback term, dependent on circular velocity, was invoked to eject gas more efficiently from lower mass galaxies, so suppressing star formation and more accurately matching the faint-end of the observed galaxy luminosity function (LF) (Lacey & Silk, 1991; Cole, 1991; White & Frenk, 1991). Later, problems appeared at the bright-end of galaxy LF within simulations with updated cosmological parameters, where too much gas was now cooling in high-mass halos (Benson et al., 2003). A number of prescriptions were suggested to remedy this. For example, Baugh et al. (2005) used a "superwind" term in their calculation of supernova feedback to permanently eject gas from high-mass galaxies, preventing star formation in high-mass halos, and (Bower et al., 2006) introduced the idea of feedback by active galactic nucleae to suppress star formation in the most massive galaxies. This leads us to a discussion of the kinds of models used to investigate galaxy formation within the hierarchical structure formation framework.

1.2 Modelling Galaxy formation

Among the more sophisticated *physical* models of galaxy evolution, two threads have emerged. The class of model that enjoyed the first successes, dubbed semi-analytic models (SAM), tracks the evolution of baryons using a halo-merger tree, which nowadays are extracted from large, high-resolution dark matter-only N-body simulations (Lacey et al., 2016; Somerville & Davé, 2015; Lagos et al., 2018; Lu et al., 2014). The models apply observationally and theoretically motivated prescriptions to describe key processes in galaxy formation including—but not limited to—gas cooling, star formation, the feedback from supernovae and active galactic nuclei, and chemical evolution. These models, due to their modular nature where one or many processes can simply be switched on and off, allow us to easily test the importance of various theoretical processes which might allow a model to better match the observed galaxy population, and have uncovered several key effects now considered centrally important in all models of galaxy formation. For example, White & Frenk (1991) and Cole et al. (1994) demonstrated the importance of supernova feedback in governing the slope of the faint-end of the low-redshift galaxy luminosity function. Baugh et al. (1996) and Kauffmann (1996) illustrated the role of mergers in shaping the Hubble sequence. Bower et al. (2006) and Croton et al. (2006) demonstrated the need for AGN feedback in allowing the models to match the sharp break at the bright end of the low-redshift luminosity function.

More recently, hydrodynamic simulations which track the dark matter and gas distributions more directly have increased in popularity and accuracy. This is due to the availability of ever increasing computing power and improved algorithms, which have resulted in the simulations being able to probe cosmologically representative volumes at the required spatial resolution, and in some cases have begun to match basic statistical galaxy populations with a similar accuracy to that displayed by SAMs (Vogelsberger et al., 2014; Schaye et al., 2015). These simulations are attractive because they provide a more explicit description of galaxy formation in terms of gas and dark matter particles, although the analysis of such models is therefore more complex (such as the need to define what is meant by a galaxy). Whilst the hydrodynamic simulations can relax some of the assumptions and approximations made in SAMs, such as removing the need to assume spherical symmetry when computing the rate at which gas cools, the simulations nevertheless are still restricted on two levels: i) they have a finite resolution and ii) the lack of a detailed physical knowledge of all of the processes behind galaxy formation. As a result, many phenomena are still dealt with using "sub-grid" physics in hydrodynamical simulations, though both the scale at which the "hand over" from direct simulation takes place, and the level of information involved differ from those available in SAMS. A prime example is the interaction of SNe with galactic gas. In the semi-analytical model used in this thesis, the SNe feedback is handled by specifying the mass loading of a wind in terms of parameters and the circular velocity of a model galaxy (see Chapter 2). On the other hand, the Illustris simulation described by Vogelsberger et al. (2014) covers a volume of 106.5 Mpc on a side, with a gravitational force softened on a scale of 710 pc. The hydrodynamic scheme used is a deformable mesh code called AREPO (Springel 2010) and the smallest mesh elements are 48 pc, comparable to the size of a giant molecular cloud. Hence, the formation of stars cannot be directly resolved, even if we had a physical, first principles model to give us the equation for the star formation rate. The heating of the interstellar medium (ISM) by supernovae is made more efficient by temporarily turning off the hydrodynamic coupling between the wind and the ISM by hand.

1.3 Model parameters

Both kinds of galaxy formation model outlined above, semi-analytic models and hydrodynamic simulations, require a number of adjustable parameters to be set which govern various processes of the model, for example, the strength of supernova winds and feedback from active galactic nuclei (AGN). These processes have been demonstrated to be very important in governing the observed properties of the galaxy population (Bower et al., 2006; Croton et al., 2006), but operate on scales below the current scope of large-scale galaxy formation modelling. For this reason, and often because we lack an accurate physical model of the process, these effects are treated using parameterised prescriptions. The parameters are then tuned to match a subset of observational datasets, often called calibration data. In the original models, the setting of the model parameters was done in a simplistic way, varying parameters one at a time to build intuition, and performing a "chi-by-eye" comparison to the calibration data. The modernisation of this tuning process is the primary concern of this thesis, and has been the focus of a number of previous works. This effort has generally taken two forms: direct exploration of the model parameter space, and emulation. Although SAMs are orders of magnitude cheaper than hydrodynamic simulations, direct exploration of their parameter space is computationally expensive due to the sheer number of model runs required for a formal search; often this will take a prohibitive length of time except for the case of tuning the parameters to a small number of datasets. This approach has been investigated in a number of papers. Kampakoglou et al. (2008) implemented Markov-Chain Monte Carlo (MCMC) techniques to calibrate a SAM to multiple datasets. Henriques et al. (2009) again used MCMC to calibrate the `L-GALAXIES` SAM to a number of a datasets, finding that the choice of datasets altered the values of the best-fitting parameters, pointing to deficiencies in their model. Martindale et al. (2017) expanded on this to include the HI mass function as a constraint, leading to a change in the best-fitting parameters. Lu et al. (2011, 2012) constrained the parameter space which gave acceptable fits to the K-band luminosity function (LF), and expanded this to include the HI mass function in Lu et al. (2014). Ruiz et al. (2015) used particle swarm optimization to calibrate a SAM to the K-band LF.

More sophisticated approaches (e.g. Benson & Bower, 2010) have been emulator-based. That is, they construct a statistical model of the galaxy formation model itself in order to search the parameter space. The model emulator has a much lower computational cost than running the full model, and produces similar output, which allows a more extensive search of the parameter space to be carried out, either in terms of the number of models run (and hence the information extracted from the parameter search, such as the range of acceptable models) or the number of parameters permitted to vary. The methodology developed in Vernon et al. (2010) has been applied widely to galaxy formation models (e.g. Rodrigues et al., 2017; van der Velden et al., 2021). In this approach, an emulator is constructed from a wide sampling of the parameter space, and iteratively refined on smaller subsets of the space based on a measure of implausibility, as defined in Vernon et al. (2010), until a sub-space of models which is deemed to give an acceptable fit to the observational datasets is reached. In this thesis, we consider alternative emulator-based approaches and optimisation methods to speed up the process of model calibration and parameter space exploration, which do not require user intervention.

1.4 Machine Learning

Machine learning methods are computational methods for building predictive models directly from data (e.g. Baron, 2019; Emmert-Streib et al., 2020). That is, given a set of inputs $\mathbf{x}$ to some unknown function $f(\mathbf{x})$, and the corresponding outputs $\mathbf{y}$, machine learning methods aim to construct a statistical model of f, $\hat{f}$, which approximates the mapping from $\mathbf{x}$ to $\mathbf{y}$. Certainly, the emulator methods applied in Vernon et al. (2010) already amount to a kind of machine learning, where a model is constructed from the data itself, without a strongly defined physical model.

The field of machine learning comprises a set of techniques to optimally construct these models so that our approximation of the function is as accurate as possible (Bishop, 2007). For example, a neural network—a network of neurons which take in inputs, perform a dot product and apply an activation function, and feed these operations forward to produce an output—is able to approximate any smooth function. Choices of activation function include the Rectified Linear Unit (ReLu), which is defined as $\sigma(x) = \max(0, x)$ and the sigmoid activation function $\sigma(x) = 1/(1 + e^{-x})$. These functions introduce non-linearity into the computation and govern the smoothness and extrapolation properties of the network. For example, the sigmoid function will suppress very large input values of x, whereas the piece-wise linear ReLu function will not, instead extrapolating linearly. The field of machine learning equips us with techniques for ensuring that these methods provide us with approximate functions that are able to be generalized beyond the data on which they are trained, for example, techniques such as cross-validation might be used to ensure that any model constructed also generalizes to unseen samples.

1.5 Emulation and optimisation

This thesis is primarily concerned with how we can understand and calibrate semi-analytic models quickly, effectively and transparently. In this context, semi-analytic models can be thought of as black-boxes which take a set of input parameters $\mathbf{x}$, and return a population of galaxies corresponding to that formulation of the model. We then compute some statistical properties of these galaxies, and adjust the parameters until we match a subset of the observations of the same statistical properties calculated with data from the actual Universe.

Within this framework we represent the galaxy formation model as a function

$$\mathbf{y} = f(\mathbf{x}) + \epsilon, \tag{1.1}$$

where $\mathbf{x} \in X$ is a set of parameters within a larger space of explorable parameters X, and ϵ is a noise term, usually the shot-noise which results from the finite simulation volume used (we typically sample a subset of the merger histories available from the full N-body simulation, which, for example, makes predictions of the luminosity function subject to noise at the brightest magnitudes). Generally, we then want to calculate some measure of the error, $f_{\mathrm{e}}(\cdot)$, between the predictions from the simulation and the real Universe, which will be a function of $(\mathbf{y} - \mathbf{y}_{\mathrm{obs}})$, the uncertainty on the observations σ_{obs} (and perhaps also the uncertainty on the emulator σ_{em}).

We aim to find a set of parameters $\mathbf{x}^* \in X$ such that $f_{\mathrm{e}}(\cdot)$ is minimised, or some distribution over x which describes the possible values of these parameters given the observational data $\mathbf{y}_{\mathrm{obs}}$, $p(\mathbf{x}|\mathbf{y}_{\mathrm{obs}})$, i.e. the set of parameters which produce results that are statistically compatible with the calibration data. Hence, the result of the model calibration can be a range of best-fitting models, rather than a single model.

The galaxy formation model is, in general, expensive to evaluate. that is, in the interest of minimizing our computing time, we would like to perform as few evaluations of the full model as possible. Often therefore, we would like to build a surrogate model or emulator of the full model which we can evaluate much more cheaply (and many orders of magnitude more quickly) which will approximate evaluating the full model as well as possible. That is, we want to estimate the error $f_{\mathrm{e}}(\cdot)$, as a function of the input parameters $\mathbf{x}$, using a statistical model.

To do this we have a number of choices. Ultimately, we want to infer the correct choice of parameters given a set of observed data. That is, we want to approximate the probability of a given set of parameters given some observational data $p(\mathbf{x}|\mathbf{y}_{\mathrm{obs}})$. The most intuitive approach is to simply build a statistical model of the galaxy formation model itself. For example, we may have a set of evaluations at parameter-space points $\mathbf{x}$, and the corresponding synthetic galaxy properties $\mathbf{y}$. We then build a simple surrogate model $\tilde{f}(\mathbf{x})$, which is able to predict $\mathbf{y}$. Once we are convinced that our emulator accurately approximates the full model, we could then use a simple MCMC routine to infer the probability density of the parameters $p(\mathbf{x}|\mathbf{y}_{\mathrm{obs}})$. Often, fully approximating the galaxy formation model requires a reasonably large number of full model evaluations.

Another approach, which we also explore, is to directly model the error between our galaxy formation model's predictions and the observational datasets considered. Given a parameter space X and a sample from this space, $\mathbf{x}$, we calculate some error which is a function of $\mathbf{x}$, $f_{\mathrm{e}}(\mathbf{x})$. Our goal is then to approximate $f_{\mathrm{e}}(\mathbf{x})$ *directly* using machine learning techniques, and find the parameter space sample $\mathbf{x}^*$ which minimizes the function $f_{\mathrm{e}}(\mathbf{x})$. In this way, we are not constructing an emulator of the full output for the galaxy formation model for the observational datasets we are considering (e.g. the values of the K-band luminosity function), but instead some measure of the error between our predicted values and the observations.

1.6 Thesis Outline

This thesis investigates the application of machine learning and optimisation techniques to explore and calibrate a semi-analytic model of galaxy formation. The aim is to replace the traditional method of setting the model parameters, which involved developing a feel for an important subset of the full model parameter space and devoting effort to building intuition about how the model predictions responded to changes in these parameters. This approach has a number of drawbacks: it is hard to reproduce, the comparison to observations is approximate, and it is impossible to carry out a methodical search of even the subset of the parameter space chosen, let alone to investigate a more plausible parameter space. The methodology we present in this thesis can be applied to any complex scientific model which is computationally expensive to evaluate and which contains an appreciable number of parameters (or order tens of parameters).

In Chapter 2, we introduce the `GALFORM` semi-analytic galaxy formation model (Cole et al., 2000a; Lacey et al., 2016)), outlining the processes followed and explain how they are modelled. We also introduce the associated model parameters and put them into context within the galaxy formation modelling framework. In

Chapter 3, we apply a deep learning approach to optimising the GALFORM model parameters over a wide range of datasets. We also apply sensitivity analysis techniques (previously explored in Oleśkiewicz & Baugh (2020)), to quantify the relative importance, effect and interactions between the different model parameters. We use our model calibration pipeline to identify tensions which arise from the choice of calibration data. We present a new version of the model introduced by Lacey et al. (2016) calibrated against a broad range of observational data, and quote ranges on the parameter values which correspond to statistically acceptable models. In Chapter 4, we apply optimisation methods to investigate the choice of initial mass function within the GALFORM model, where we directly model the error of the galaxy formation model and the observational data rather than building a full model, and use Bayesian optimisation to select candidate parameter samples. Previously, using the old fashioned and burdensome model calibration process described above, Baugh et al. (2005) argued that a top-heavy stellar initial mass function was needed in bursts of star formation to accommodate the observed number and redshift distribution of dusty star forming galaxies in the model, at the same time as reproducing local observations. Our new approach allows us to convincingly demonstrate that there is no unexplored corner of a high dimensional parameter space that would allow the model to match all of these constraints *simultaneously* without making this assmuption, and that, within the limitations of the current GALFORM model, a top-heavy IMF is needed.

Finally, in Chapter 5, we present our conclusions.

Chapter **2**

Theoretical Background

2.1 GALFORM

GALFORM is a state-of-the-art *ab initio* physically motivated semi-analytical model of galaxy formation. The model tracks the merger histories of dark matter haloes, the cooling of gas to form galactic disks, quiescent star formation in the disk, bursts of star formation associated with mergers or disk instabilities, the resultant feedback and gas ejection driven by supernovae, the role of heating by AGN in inhibiting gas cooling, and the chemical enrichment of stars and gas (for a full description of GALFORM see Cole et al., 2000b; Lacey et al., 2016).

Briefly, the model tracks

- the collapse of dark matter into DM-halos, and their subsequent merging
- the heating and cooling of gas inside these halos, and the formation of galactic disks
- star formation in both the disk and starbursts
- feedback effects such as the ejection of gas by supernovae (and its subsequent return to the hot gas halo), and the heating of gas by AGN to prevent it from cooling

- galaxy mergers and galactic disk instabilities
- calculation of the sizes of disks and spheroids based on hydrostatic equilibrium, conservation of angular momentum and halo contraction
- chemical enrichment of stars and gas
- the effect of galactic dust on the starlight emitted by galaxies

In the follow sections, we give a description of each of the processes which will be explored in this work.

2.1.1 Quiescent star formation in disks

The model uses an empirical star formation law formulated by Blitz & Rosolowsky (2006) (and implemented in GALFORM in Lagos et al., 2011) based on observations of nearby star-forming disk galaxies. The star formation rate in the disc is

$$\psi_{\rm disk} = \nu_{\rm SF} M_{\rm mol,\ disk}, \tag{2.1}$$

where $M_{\rm mol,\ disk}$ is the mass of molecular gas in the disk, and $\nu_{\rm SF}$ is a constant which we treat as an adjustable parameter within a reasonable range suggested by observations (Bigiel et al., 2011). The fraction of cold gas in the molecular phase depends on the gas pressure in the mid-plane of the disc.

2.1.2 Supernova feedback

Supernova feedback causes gas to be ejected from galaxies and out of the halo. The model assumes that this mass ejection is proportional to the instantaneous star formation rate, ψ, with a mass loading factor which depends on the circular velocity of the galaxy, $V_{\rm c}$:

$$\dot{M}_{\mathrm{eject}} = \left(\frac{V_{\mathrm{c}}}{V_{\mathrm{SN}}} \right)^{-\gamma_{\mathrm{SN}}} \psi, \tag{2.2}$$

where both V_{SN} and γ_{SN} are model parameters. We can further separate V_{SN} into $V_{\mathrm{SN,\ disk}}$ and $V_{\mathrm{SN,\ burst}}$, allowing for different mass loadings in quiescent star formation and bursts, although these parameters have generally been assumed to be equal in most previous versions of the model (see Benson & Bower 2010 for an exception). Gas ejected from the halo is assumed to gradually return from a reservoir beyond the virial radius of the halo to the hot gas reservoir at a rate given by

$$\dot{M}_{\mathrm{return}} = \alpha_{\mathrm{ret}} \frac{M_{\mathrm{res}}}{\tau_{\mathrm{dyn,halo}}}, \tag{2.3}$$

where $\tau_{\mathrm{dyn,halo}}$ is the dynamical time of the halo, M_{res} is the mass in the reservoir beyond the virial radius, and α_{ret} is a free parameter.

2.1.3 Galaxy mergers

In the model, galaxy mergers can trigger bursts of star formation and destroy galactic disks. We define two thresholds, f_{ellip} and f_{burst} which are used to classify mergers and which govern their outcomes. When a satellite galaxy with baryonic mass $M_{\mathrm{b,\ sat}}$ merges with a central galaxy with baryonic mass $M_{\mathrm{b,\ cen}}$ two types of mergers may occur. First, if $M_{\mathrm{b,\ sat}}/M_{\mathrm{b,\ cen}} \geqslant f_{\mathrm{ellip}}$ the merger is classified as a *major* merger, and the disk component of the galaxy is destroyed and forms a spheroid. The cold gas in the disk is assumed to be consumed in a burst of star formation which adds new stars to the spheroid. Second, if $M_{\mathrm{b,\ sat}}/M_{\mathrm{b,\ cen}} < f_{\mathrm{ellip}}$, the merger is classified as *minor*, and the disk survives the merger. In this case, the cold gas is consumed in a starburst if a second condition is met, $M_{\mathrm{b,\ sat}}/M_{\mathrm{b,\ cen}} \geqslant f_{\mathrm{burst}}$. Both f_{burst} and f_{ellip} are treated as free parameters. In the improved galaxy merger model of Simha & Cole (2017), once a subhalo can no longer be resolved, an analytic calculation of the merger time is made based on dynamical friction arguments.

2.1.4 Disk instabilities

Galactic disks dominated by rotational motion can become unstable to bar formation if their degree of self-gravity is too large. The model follows the work of Efstathiou et al. (1982), and assumes that disks become unstable if the criterion

$$\frac{V_{\rm c}(r_{\rm disk})}{(1.68GM_{\rm disk}/r_{\rm disk})^{1/2}} \leq F_{\rm stab} \tag{2.4}$$

is met, where $M_{\rm disk}$ is the total disk mass and $r_{\rm disk}$ is the disk half-mass radius. Numerical simulations of a suite of exponential stellar disks by Efstathiou et al. (1982) suggested a value of $F_{\rm stab} \approx 1.1$, while Christodoulou et al. (1995) found a value of 0.9 for gaseous disks. A value of 0.61 or below corresponds to universally stable disks, since this is the value of the left hand side of Eqn. 2.4 for a completely self-gravitating disk. We allow this parameter to vary within a reasonable range (see Table 4.1). We assume that unstable disks are disrupted by bar instabilities on a sub-resolution timescale such that all the mass is instantaneously transferred to the spheroid and any gas present takes part in a burst of star formation.

2.1.5 Starbursts

Bursts of star formation, triggered by mergers or bar instabilities in dynamically unstable disks, are assumed to form stars at a rate

$$\psi_{\rm burst} = \nu_{\rm SF,\ burst} M_{\rm cold,burst} = \frac{M_{\rm cold,burst}}{\tau_{*\rm burst}}, \tag{2.5}$$

where the time scale is given by

$$\tau_{*\rm burst} = \max[f_{\rm dyn}\tau_{\rm dyn,\ bulge}, \tau_{*\rm burst,min}]. \tag{2.6}$$

Here the bulge dynamical time, $\tau_{\text{dyn, bulge}}$ is defined as $\tau_{\text{dyn, bulge}} = r_{\text{bulge}}/V_{\text{c}}(r_{\text{bulge}})$, where the velocity is the effective circular velocity at the half-mass radius of the bulge. The minimum timescale of the burst, $\tau_{*\text{burst,min}}$, is treated as an adjustable parameter in the range 1-100 Myr. f_{dyn} is held at the value of 20 used by Lacey et al (2016).

2.1.6 SMBH growth and AGN feedback

Supermassive black holes can inject energy into the halo gas, inhibiting gas cooling. Hot halo accretion, BH-BH mergers, as well as starbursts can increase the mass of the black hole (Bower et al., 2006; Griffin et al., 2019). In the case of starbursts, the mass accreted onto the SMBH is a fraction f_{SMBH} of the mass of stars formed, where f_{SMBH} is an adjustable parameter. AGN accretion is assumed to occur if both of the following conditions are met: (1) that the gas halo is in quasi-hydrostatic equilibrium, that is the condition

$$\tau_{\text{cool}}/\tau_{\text{ff}} > 1/\alpha_{\text{cool}}, \tag{2.7}$$

is met, where τ_{cool} is the cooling time of the gas, τ_{ff} the free-fall time, and α_{cool} is an adjustable parameter; (2) The AGN power required to balance the radiative cooling luminosity is below a fraction f_{Edd} (fixed at 0.05) of the Eddington luminosity of the SMBH.

2.1.7 Stellar initial mass function

The stellar initial mass function (IMF) gives the mass distribution of newly formed stars, and strongly affects the evolution of a the stars and their luminosity, as well as the metal and gas content of the galaxy. The IMF, $\Phi(m)$ is defined such that the number of stars with mass m is $dN = \Phi(m)\text{dln}m$, and $\Phi(m)$ is normalised such that $\int_{m_{\text{L}}}^{m_{\text{H}}} \Phi(m)\text{dln}m = 1$ between some maximum (m_{H}) and minimum (m_{L}) stellar mass.

In some of the models considered later we will allow the slope of the IMF to depend on the mode of star formation, assuming a solar neighbourhood IMF for quiescent star formation that takes place in disks and a power law IMF for bursts of star formation. The slope of the power law IMF is then a model parameter:

$$\Phi(m) := dN/d\ln m \propto m^{-x}. \tag{2.8}$$

To accommodate this change, we must self-consistently calculate the recycled fraction (i.e. the fraction of mass returned to the ISM from mass lost by stars over their lifetime), given by

$$R = \int_{1M\odot}^{m_H} (m - m_{\rm rem}(m))\Phi(m){\rm d}\ln m, \tag{2.9}$$

where $m_{\rm rem}$ is the mass of the remnant left by a dying star of birth mass m. We also calculate the yield, i.e. the fraction of the initial mass synthesised into metals and ejected, p, as

$$p = \int_{1M\odot}^{m_H} p_{\rm Z}(m) m\Phi(m){\rm d}\ln m, \tag{2.10}$$

where $p_{\rm Z}(m)$ is the fraction of mass ejected as metals by a star of mass m. For reference, the solar neighbourhood IMF assumed in quiescent star formation tends to a power law slope of $x = 1.35$ above one solar mass and turns over with a log-normal form below one solar mass (Chabrier, 2003). Most previous `GALFORM` models used the Kennicutt (1983) form of the solar neighbourhood IMF.

2.1.8 Absorption and reradiation of starlight by dust

Within galaxy disks, a two component dust model is assumed with diffuse and molecular cloud components. In a burst, the dust is assumed to be in the form of molecular clouds. In both cases, the dust is mixed in with the stars, although these components can have different scale heights (see Granato et al. 2000). The mass of dust in a galaxy is assumed to be a constant fraction $\delta_{\rm dust}$ of the mass of metals in the cold gas, $M_{\rm dust} = \delta_{\rm dust} Z_{\rm cold} M_{\rm cold}$, where $\delta_{\rm dust} = 0.334$ and $Z_{\rm dust}$ is the metallicity of the cold gas (Silva et al., 1998). The optical depth is then calculated as

$$\tau_{\rm dust,\lambda} = 0.043 \left(\frac{k_\lambda}{k_{\rm V}}\right) \left(\frac{\Sigma_{\rm gas}}{M_\odot {\rm pc}^{-2}}\right) \left(\frac{Z_{\rm cold}}{0.02}\right), \tag{2.11}$$

where $\Sigma_{\rm gas}$ is the surface density of the gas and the k quantities are related to the extinction curve.

The model assumes that an adjustable fraction $f_{\rm cloud}$ of the dust is in clouds of mass $m_{\rm cloud}$ and radius $r_{\rm cloud}$. The mass and radius of the clouds are assumed to be constant and are based on observations of local galaxies (Granato et al., 2000). Stars are assumed to form inside molecular clouds and escape over an adjustable timescale $t_{\rm esc}$. The optical depth in clouds therefore scales as $\tau_{\rm cloud} \propto f_{\rm cloud} m_{\rm cloud}/r^2{}_{\rm cloud}$, and the optical depth of the diffuse component as $\tau_{\rm diffuse} \propto (1 - f_{\rm cloud}) M_{\rm cold} Z_{\rm cold}/r^2{}_{\rm diffuse}$. Here, $r_{\rm diffuse}$ is taken to be the disk radius, $r_{\rm disk}$, for stars in the disk, and $r_{\rm bulge}$, for stars in the bulge. Attenuation by diffuse dust is calculated by interpolating the tabulated results of radiative transfer runs by Benson (2018) using the `HYPERION` code of Robitaille (2011). These models are higher resolution versions of the extinction tables previously used in `GALFORM`, which were based on the radiative transfer calculations by Ferrara et al. (1999), and extend over a wider range of optical depth values, making the interpolation of model galaxy properties more accurate.

The dust re-radiates the photons it absorbs from the starlight at infra-red and sub-millimeter wavelengths. In the model, the emission from the diffuse and cloud components are treated separately. We calculate the total stellar luminosity absorbed by the dust in a galaxy, and assume the dust radiates this energy as a modified black body

$$L_\lambda^{\rm dust} \propto M_{\rm dust}\kappa_{\rm d}(\lambda)B_\lambda(T_{\rm dust}), \tag{2.12}$$

where $M_{\rm dust}$ is the mass and $T_{\rm dust}$ the temperature of the dust component, $B_\lambda(T)$ is the Planck function, and $\kappa_{\rm d}$ is the dust opacity per unit mass. Integrating $L_\lambda^{\rm dust}$ over wavelength and setting the result equal to the absorbed luminosity allows us to solve for the dust temperature, $T_{\rm dust}$, for each component. The dust opacity is approximated as a broken power law

$$\kappa_{\rm d} \propto \begin{cases} \lambda^{-2} & \lambda < \lambda_{\rm b} \\ \\ \lambda^{-\beta_{\rm b}} & \lambda > \lambda_{\rm b}, \end{cases} \tag{2.13}$$

where we allow the adjustable exponent $\beta_{\rm b}$ for bursts to vary within the range 1.5 - 2.0 (see e.g. Silva et al. 1998) and $\lambda_{\rm b} = 100\mu$m. For emission from dust in the disk, there is no break in the power law for $\kappa_{\rm d}$.

2.1.9 The Lacey2016 model

Since the model of Cole et al. (2000a), there have been a number of iterations on the `GALFORM` code. The Lacey2016 `GALFORM` model (Lacey et al., 2016), which includes a different IMF slope in bursts to that used in quiesent star formation and the AGN feedback treatment developed in Bower et al. (2006) is a comprehensive calibration of the model applied to the Millennium WMAP-7 N-body simulation. This simulation has a side length $500h^{-1}$Mpc and halo mass resolution $1.87 \times 10^{10} h^{-1} M_\odot$, and a dark matter particle resolution of $9.36 \times 10^{8} h^{-1} M_\odot$. The simulation output consists of 64 snapshots at varying redshifts. These outputs are used to construct the dark matter halo merger trees. Halos are first identified using a friends-of-friend (FoF) percolation algorithm, and then `SUBFIND` is run on the resulting particle lists to identify subhalos within these structures Springel et al. (2001). The subhalo merger trees are processed through the `DHALOS` algorithm to ensure that the halo mass increases monotonically, and to apply a set of conditions to determine if subhalos have merged (Jiang et al., 2014). The `DHALOS` algorithm groups subhalos into dark matter halos with strictly hierarchical formation histories, overcoming issues of artificial mergers and tenuous bridges often encountered with traditional FoF halos.

Lacey et al. (2016) provides a comprehensive exploration of the model predictions, illustrating how they change when varying key model parameters. This model shows good agreement with a large range of observational datasets across both redshift and wavelength. In particular, the model was designed to reproduce local calibration data, such as the K-band galaxy luminosity function, and to match the properties of high redshift galaxy populations, including the number counts and redshift distribution of galaxies seen through their emission at submillimetre wavelengths, and the luminosity function of galaxies in the ultra-violet. The model is therefore set up to reproduce dusty star forming galaxies seen through their dust emission and star forming galaxies seen through their attenuated emission at UV wavelengths.

2.1.10 The Baugh19 model

The Baugh et al. (2019) model is a recalibration of the Lacey et al. (2016) model for implementation in a higher-resolution dark matter only N-body simulation, the P-Millennium simulation. This simulation also uses updated values of the Λ-cold dark matter model parameters from the Planck mission. The P-Millennium simulation is both larger and has an improved dark matter halo resolution, with side length 800 Mpc, halo mass resolution of $2.12 \times 10^9 h^{-1} M_{\odot}$, and dark matter particle mass of $1.06 \times 10^8 h^{-1} M_{\odot}$, representing about a factor of almost 10 increase in the halo mass resolution compared to the WMAP-7 simulation. This has important implications for the model, and so required a recalibration (albeit a simple one). Ultimately, Baugh et al. (2019) adjusted only two parameters, $\gamma_{\rm SN}$, which controls the strength of supernova feedback, and $\alpha_{\rm ret}$, which controls the rate of gas return following ejection by supernovae. These parameters were changed to match the agreement with the local galaxy luminosity function enjoyed by the Lacey2016 model. The choice of parameters to be adjusted was made after building intuition about how different parameters affect the galaxy luminosity function, and performing a series of runs varying these parameters. This is an illustration of the original way in which model parameters were set, and is discussed further in Chapters 3 and 4.

2.2 Updates to the GALFORM model

Here we describe updates to the GALFORM model which we used in this work versus previous iterations of the model.

2.2.1 Dust modelling

In this iteration of the model, we use an improved scheme for the modelling of the reprocessing of starlight by galactic dust. Previous iterations of the model were based on the ray-tracing calculations of Ferrara et al. (1999). In the model, attenuation factors are calculated by interpolating in the tables in which the ray-tracing calculations from Ferrara et al. are stored, which record the attenuation as a function of optical depth, the ratio of spheroid to disk scale-lengths, wavelength, and inclination. The Ferrara et al. (1999) tables are sparsely populated, and updated models of dust grains are now available (Draine, 2003), and far finer tabulations covering a much wider range of the variables of interest have been made (Benson, 2018).

Previously, the model relied on interpolations and power-law extrapolations of the variables to calculate attentuation factors for the model galaxy population. With these updated tables we aim to make the interpolations more accurate, by including more outputs of the radiative transfer calculation, and to avoid extrapolation by producing outputs for a wider range of parameter values. We have therefore implemented the high-resolution radiative transfer runs supplied in Benson (2018),

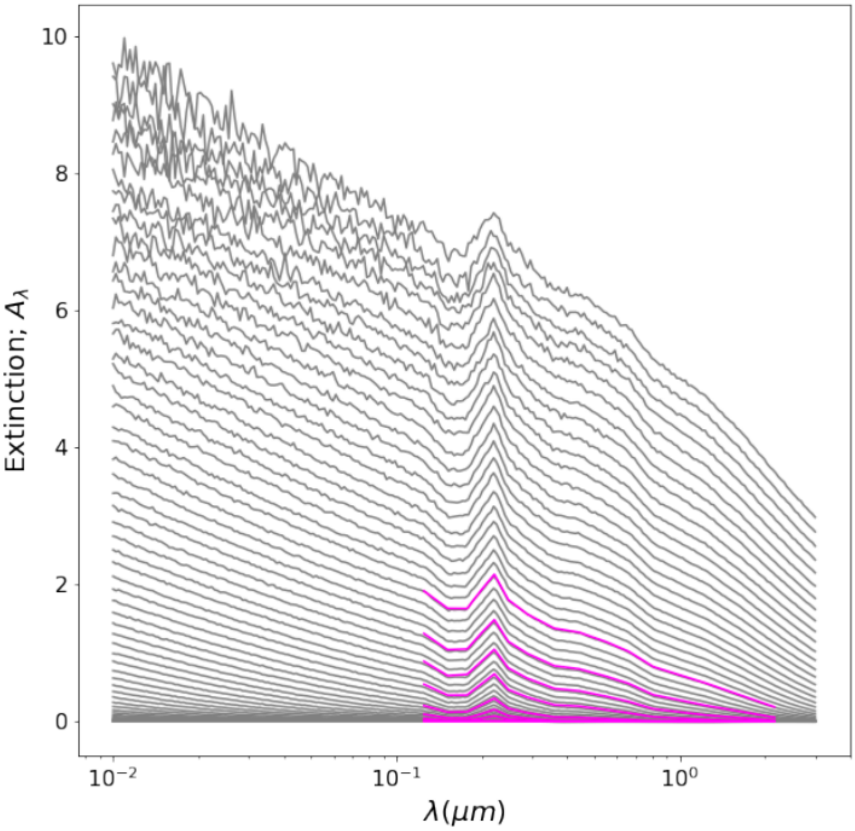


Figure 2.1: Comparison between the Benson (2018) disk extinction curves (*grey*) and the values tabulated by Ferrara et al. (1999) (*magenta*) for fixed optical depth and varying inclination. (Lower value attenuation curves correspond to the disk being face-on.)

which allows the user to specify a number of different dust prescriptions. A comparison between the tables provided by Benson (2018) and Ferrara et al. (1999) is shown in Fig 2.1.The magenta lines show the tabulation provided by Ferrara et al. (1999) for a galactic disk at varying inclinations, and the grey lines show the updated calculations of Benson (2018). The Benson curves cover a much wider range of wavelength and attenuation values.

Benson (2018) also calculated extinction curves for a variety of model dust grains. Fig 2.2 shows a few choices which can now be selected by the user. The blue line shows the KMH model (Kim et al., 1993), the model of Draine (2003) is shown in red, and the updated calculations with the same underlying dust grain model as

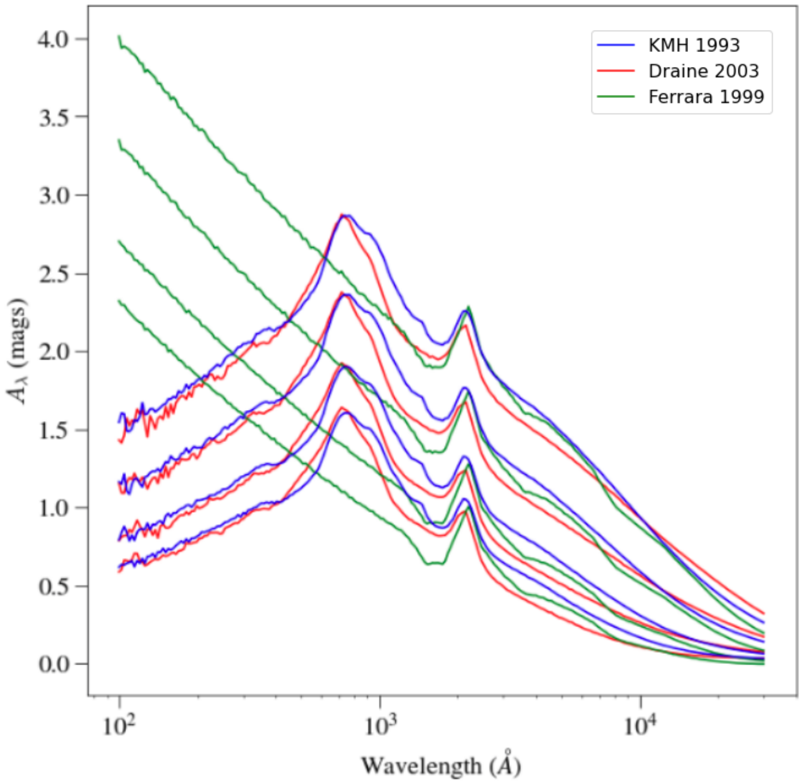


Figure 2.2: Extinction curves as calculated by Benson (2018) assuming 3 different dust models. *Blue*: The model of Kim et al. (1993). *Red*: the model of Draine (2003). *Green*: the model of Ferrara et al. (1999). We show extinction curves for the disk component, assuming a fixed optical depth and varying inclination.

Ferrara et al. (1999) are shown in green. As before, here we show the extinction curves for a galactic disk with varying inclination, though similar calculations exist for the bulge component. We tested the effect of varying these models on the low-redshift K-band and TH1500 (a top-hat filter centered at 1500Å) luminosity functions using the Baugh et al. (2019) model, but found little impact on the model predictions. It would be interesting to extend these tests in the future to examine the effect on the predictions for the UV luminosity function at high redshifts; however, the main differences seen in Fig. 2.2 are at shorter wavelengths than probed by the UV luminosity functions, which sample the curves at wavelengths around the 2175 Å bump or longer.

2.2.2 Gas cooling

In this thesis, we use the updated cooling model introduced into `GALFORM` by (Hou et al., 2018). This model aims to address some of the shortcomings of earlier iterations of the GALFORM cooling scheme. Briefly, previous schemes tracked an outward-moving cooling front which was reset at artificial 'halo formation events', deemed to be when the mass of the halo doubled. Between these formation events, the contraction of the gas profile was not modelled. Hou et al. (2018) introduced an improved model which tracks the gas cooling more smoothly. Rather than calculating a single advancing cooling radius, $r_{\rm cool}$, two radii are used $r_{\rm cool}$ and $r_{\rm cool,\ pre}$. At a given timestep, $r_{\rm cool}$ is calculated and compared to the previous cooling radius $r_{\rm cool,\ pre}$, and the gas between these two radii cools during the timestep, and is added to a cold gas component with new radius $r_{\rm cool}$. The updated model calculates both the dynamical contraction of the hot halo due to gravity, and the expansion of the cooling radius due to gas cooling by radiation. The use of this new cooling model does not produce a significant qualitative change in the model predictions, but does mean that the corresponding parameters in a best-fitting model will be somewhat different from those obtained using the previous cooling scheme.

2.2.3 Stellar population synthesis models

Compared to recent versions of the `GALFORM` code, which largely used the stellar population synthesis (SPS) models of (Maraston, 2005), we use the flexible SPS model (FSPS) of (Conroy, 2013), which makes use of the Padova isochrones (Alongi et al., 1993; Bressan et al., 1993; Fagotto et al., 1994) and the BaSeL semi-empirical spectral library (Lejeune et al., 1998; Westera, P. et al., 2002). The use of the FSPS code rather than pre-computed tables allows us to generate SPS models for a range of IMFs. There are a large number of uncertainties inherent in the SPS modelling process. For example, core-convection overshooting is a significant source of uncertainty in the colour evolution of simple stellar populations in the age range

0.1 - 1.0 Gyr, with the Maraston (2005) model not including overshooting in its treatment of core-convection.

Importantly, the Conroy (2013) models include adjustable parameters $\Delta_{\rm T}$ and $\Delta_{\rm L}$ which correspond to offsets in the position of the TP-AGB track on the Herzsprung-Russell diagram. These parameters were calibrated by Conroy (2013) to better match color-age relations inferred from the Magellenic Cloud, achieving significantly better agreement than the Maraston (2005) models.

Chapter 3

Deep learning emulation of GALFORM

Summary: We implement a sample-efficient method for rapid and accurate emulation of semi-analytical galaxy formation models over a wide range of model outputs. We use ensembled deep learning algorithms to produce a fast emulator of an updated version of the GALFORM model from a small number of training examples. We use the emulator to explore the model's parameter space, and apply sensitivity analysis techniques to better understand the relative importance of the model parameters. We uncover key tensions between observational data sets by applying a heuristic weighting scheme in a Markov chain Monte Carlo framework and exploring the effects of requiring improved fits to certain data sets relative to others. Furthermore, we demonstrate that this method can be used to successfully calibrate the model parameters to a comprehensive list of observational constraints. In doing so, we re-discover previous GALFORM fits in an automatic and transparent way, and discover an improved fit by applying a heavier weighting to the fit to the metallicities of early-type galaxies. The deep learning emulator requires a fraction of the model evaluations needed in similar emulation approaches, achieving an out-of-sample mean absolute error at the knee of the K-band luminosity function of 0.06 dex with fewer than 1000 model evaluations. We demonstrate that this is an

extremely efficient, inexpensive, and transparent way to explore multidimensional parameter spaces, and can be applied more widely beyond semi-analytical galaxy formation models.

3.1 Introduction

Galaxy formation is a complex and non-linear process, involving the interplay of gravitational, radiative, thermal, and fluid processes. Semi-analytical modelling is an approach used to improve our understanding of this problem by reducing it to its key ingredients using simplified mathematical relations motivated by physical and geometric arguments (e.g. Baugh, 2006; Benson, 2010). These relations take the form of coupled differential equations and simple algebraic relations describing processes such as star formation, gas cooling, and bar instabilities in galactic disks. Semi-analytical models provide a comprehensive theoretical framework with which to understand and develop intuition about galaxy formation, and have produced a number of insights (e.g. White & Frenk, 1991; Benson et al., 2003; Bower et al., 2006; Croton et al., 2006; Lacey et al., 2016).

However, the semi-analytical approach has sometimes attracted scepticism for a number of reasons. The mathematical relations which describe the physical processes in the model often contain adjustable parameters, and a model is defined by a particular choice for the parameters values (analogous to the parametrised sub-grid models employed in hydrodynamic simulations, e.g. Crain et al., 2015; Somerville & Davé, 2015) These parameters are sometimes set by theoretical or observational considerations, but in many cases they are less well specified (for example, in the case of the parameters governing the strength of feedback due to supernovae - SNe).

There is a perception–which we believe to be misplaced–that these ‘free’ parameters allow semi-analytical models (SAMs) to fit any arbitrary combination of datasets, therefore eliminating their predictive and explanatory power. We hope to dispel this view by demonstrating that the majority of the variance in the model output is contributed by just a few parameters which have clear physical interpretations (such as the strength of feedback due to SNe or AGN), and that these dominant parameters preclude arbitrary fitting.

Another major source of the scepticism towards SAMs arises from the seemingly opaque procedures that have commonly been used to calibrate the model parameters. This process often follows a 'chi-by-eye' methodology, in which the operator adjusts the parameters by hand, interprets the effect on the model output, and adjusts the parameters again to improve the match of the model output to an observable. This requires a high level of expertise and familiarity with the SAM, and the operator often makes trade-offs between fits to different constraining datasets in a way which is poorly defined; model predictions are often judged to be good fits when formally they would be rejected. This makes the process of setting the model parameters hard to reproduce. There is also no guarantee that the by-eye approach will produce the best fit to the calibration datasets; the model parameters may interact in a non-linear way, which can be difficult to conceptualize. This, coupled with the large parameter space, makes it unlikely that such a search will find the best-fitting parameters. We aim to side-step these issues by developing a method to rapidly and robustly perform an exhaustive search of the parameter space, calibrate the SAM in an automatic way without the need for significant human intervention, and quantify the relative importance of the parameters. In this way, we hope to make the model calibration process transparent and reproducible, especially by researchers with less experience of running SAMs. Although the cosmological parameters are now well constrained, SAMS must still be re-tuned for simulations with different resolutions and cosmologies, such as $f(R)$ gravity simulations, or when a new implementation of a process is included. The question of how to set the model parameters therefore remains a relevant one.

Here, we aim to emulate an updated version of the `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), but which also includes an improved model for gas cooling in halos (introduced by Hou et al., 2018).

We focus specifically on using deep learning to build our emulator (for an introductory review, see e.g. Emmert-Streib et al., 2020). This sub-field of machine learning uses stacked neural layers (hence *deep*) to build flexible function approximators which are able to uncover non-linear relations in data without the need for a strongly pre-defined model, and have proven to be highly successful in astronomical applications (e.g. Ravanbakhsh et al., 2016; Schmit & Pritchard, 2018; Perraudin et al., 2019; He et al., 2019; Cranmer et al., 2019; Yip et al., 2019; De Oliveira et al., 2020; Ntampaka et al., 2019). We demonstrate that deep learning algorithms can be applied to accurately emulate SAMs over the full range of model outputs, and require a relatively small number of training examples to achieve good accuracy when compared to other methods. Since the deep learning emulator can be evaluated orders of magnitude faster than the time taken to run GALFORM, we are able to run many simple MCMC chains to explore the parameter space, and investigate how calibration to different datasets constrains the model parameters. We achieve this by minimizing the absolute error between the emulator output and the data, and employing a heuristic weighting scheme to the different observational datasets to mimic the process employed by model practitioners. In this way, we hope to elucidate and automate the calibration process, as well as exhaustively search the parameter space of the model. A similar approach has been explored in Forbes et al. (2019), applied to a semi-analytical model of galactic disks, though our sampling scheme and MCMC implementation are different.

This approach has a number of advantages over previous work. Non-emulation approaches such as MCMC and particle swarm optimization offer a powerful way to quantify parameter uncertainty and fit the model to a particular observable, but are limited in terms of exploring and understanding the full parameter space, and come at significant computational expense. Previous emulation approaches, though informative, also do not fully address our aims; they have focused on reducing the parameter space based on measures of implausibility (a measure which incorporates information about the emulator prediction and target data, and their variances, to rule out regions of parameter space). By iteratively refining more approximate emulators over a number of waves of model runs, these methods hone in on a region of parameter space which could plausibly contain good fits to a predefined set of just a few observables. Here, we focus on producing an emulator of the GALFORM model which is accurate across the entire parameter space. This allows us to explore the full parameter space of the model and fit to a wide range of observables, and to consider more diverse combinations of observables than has been attempted in previous work. We also aim to reduce the requirement for a large number of GALFORM evaluations. Rodrigues et al. (2017), for example, used 7 waves of 5000 runs each to hone in on the region of parameter space which gave acceptable fits to the local galaxy stellar mass function; here, we limit ourselves to < 1000 full GALFORM runs. In doing so we intend to develop a general method for investigating, understanding, and calibrating SAMs in an inexpensive, flexible, and reproducible way.

We also apply sensitivity analysis techniques to the model parameters, as recently applied to GALFORM by Oleśkiewicz & Baugh (2020). This allows us to judge the importance of different parameters by quantifying the proportion of the model variance due to a given parameter through sensitivity indices. We are also able gauge the degree of interaction between parameters, giving us important insight into the model.

The layout of the chapter is as follows. In § 3.2 we review the theoretical background. The key processes of GALFORM that are relevant to this work are described in § 2.1. In § 3.2.1 we give a brief review of the deep learning approach and our emulator design, and in § 3.2.2 we give a description of the sensitivity analysis method. In §3.2.3 we describe the observational constraints under consideration, and in § 3.2.4 we discuss how we find best-fitting parameters using MCMC. In

§ 3.3 we present our results. In §3.3.1 we review the predictive performance of the emulator, in § 3.3.2 we show the results of our sensitivity analysis and model exploration, and in § 3.3.3 we present our model calibration results. In §3.4 we discuss the merits of our methods and outline potential future work, and conclude in § 3.5.

3.2 Theoretical background

Here we describe the process of building a deep learning emulator and motivate our specific choice of model (§ 3.2.1). We then briefly describe sensitivity analysis (§ 3.2.2), the observational datasets considered (§ 3.2.3), and our calibration scheme (§ 3.2.4).

3.2.1 Deep learning emulator

Before we consider observational data, we aim to construct a fast emulator of the `GALFORM` model using the `tensorflow` deep learning framework (Abadi et al., 2015). We formulate the problem from the perspective of supervised learning. We treat the `GALFORM` model as an unknown function $\hat{f}(\cdot)$ that takes some input vector $\mathbf{x}$, representing a set of values for the model parameters, and produces an output vector $\mathbf{y}$, representing one or many binned statistical properties of the resulting synthetic galaxy population (e.g. the values of the K-band luminosity function in given luminosity bins). Our goal is then to develop a fast and accurate approximation to the function $\hat{f}(\cdot)$ by training an emulator to predict $\mathbf{y}$ given $\mathbf{x}$.

Since `GALFORM` is computationally expensive to run (at least in comparison to a potential deep learning emulator), we are limited in how many evaluations of the code we can perform, and so limited in the number of input-output pairs, $(\mathbf{x}_i, \mathbf{y}_i)$, we have to train our emulator. To sample the parameter space evenly and efficiently, we use Latin hypercube sampling (as described in e.g. Bower et al.,

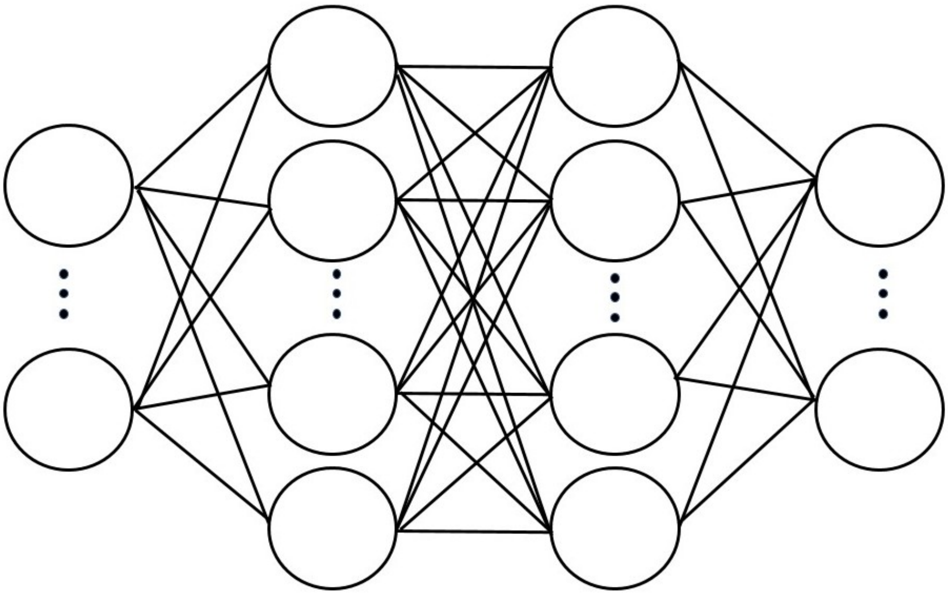


Figure 3.1: A schematic diagram showing a neural network with 2 hidden layers. The neurons on the left hand side represent the input layer, the central two layers of neurons are the hidden layers, and the right-hand neurons comprise the output layer.

2010) to generate the model input parameters. This method aims to fill the target parameter space evenly given a fixed number of samples. After evaluating `GALFORM` at these points, we are therefore left with the pairs of vectors $(\mathbf{x}_i, \mathbf{y}_i)$, corresponding to the input and output of the model. We separate the samples randomly into three sets: the training set, the validation set, and the holdout set. The training and validation sets will be used to train the emulator, and the holdout set will be kept separate so it can be used for evaluating the emulator's performance on out-of-sample data. The different roles of these sets are discussed further below.

The task of emulating `GALFORM` is therefore reduced to a regression problem. The deep learning emulator is comprised of stacked neural layers as shown in Fig. 3.1. Here we see a neural network with an input layer on the left, two *hidden* layers, and an output layer on the right. Note that the output from each neuron is passed to every neuron in the following layer. The network is defined by a set of weights and biases, W; the i-th neuron in the j-th layer contains an adjustable weight vector $\mathbf{w}_{ij}$ and an adjustable bias term b_{ij}. When we propagate inputs through our network to produce a prediction, the input layer first passes the inputs to every neuron in the first hidden layer. Each neuron i in each subsequent layer j, starting with the first hidden layer, takes in the outputs from the previous layer and calculates its own output z_{ij} by performing the computation

$$z_{ij} = \hat{\sigma}(\mathbf{z}_{j-1} \cdot \mathbf{w}_{ij} + b_{ij}), \tag{3.1}$$

where we have taken the dot product between the vector of all the neuron outputs in the previous layer $\mathbf{z}_{j-1}$ and the i-th neuron's weights $\mathbf{w}_{ij}$, and added the bias term b_{ij}. An activation function $\hat{\sigma}(\cdot)$ is then applied. This is generally a non-linear function such as the sigmoid or hyperbolic tangent function. The neuron outputs z_{ij} are then passed to the next layer and the process is repeated until we reach the final layer. The output from the final layer is then the prediction of the network for these inputs. Usually, the neurons in the final layer only apply a linear activation function. Therefore, since the network outputs are linear sums of non-linear functions of the input parameters, we can think of this method as estimating non-linear basis functions from training data.

The weights and biases associated with each neuron are adjusted during training by seeking to minimise an error function between the emulator predictions and the true values. In our case, given a set of input parameters, we want to minimise the error between our emulator's prediction of the `GALFORM` output and the actual `GALFORM` output. We choose to use the mean absolute error function (hereafter MAE)

$$\mathrm{MAE} = \frac{1}{n} \sum_{k=1}^{n} |\hat{\mathbf{y}}_k - \mathbf{y}_k|, \tag{3.2}$$

where $\hat{\mathbf{y}}_k$ is the model emulator prediction for the k-th of n samples and $\mathbf{y}_k$ is the true value. Since both $\hat{\mathbf{y}}_k$ and $\mathbf{y}_k$ are vector quantities, the modulus signs represent the L1 norm (i.e. the sum of absolute errors of the vector components); we choose this metric as it gives less weight to outliers than the more commonly used L2 norm (i.e. the sum of squared errors of the vector components). If we denote the function represented by the neural network as f, parameterised by weights and biases W, we therefore attempt to find a function f_* such that

$$f_* = \arg\min_W \{\text{MAE } (f(\mathbf{x}),\ \mathbf{y})\}. \tag{3.3}$$

The training is performed iteratively in steps known as *epochs.* During each epoch, the model weights and biases, W, are adjusted by an optimizer to minimise the MAE of the network's predictions for the training set. The optimizer is an algorithm which calculates how best to adjust the model weights by seeking minima on the error surface, usually by some form of gradient descent. We use the AMSGRAD variation of the Adam optimizer (Kingma & Ba, 2015; Reddi et al., 2018). Adam is a momentum-based optimizer and AMSGRAD aims to improve the performance of Adam around minima on the error surface. At the end of each epoch, the adjusted model is evaluated on the validation set, to ensure the model generalises to unseen data. If the performance on the validation set has improved (as measured by the MAE), we save the model weights and continue training. If the performance does not improve, we do not save the weights and continue training. This process is repeated until the performance on the validation set has not improved for 30 epochs at which point we halt the training. We then perform a final fine-tuning step using the RMSprop optimizer (Tieleman & Hinton, 2016); this optimizer uses stochastic gradient descent and treats the error surface as a quadratic bowl. For this step, we use a very low learning rate of 10^{-5}, allowing us to take small gradient-steps toward the minima of the error surface. We find this works well in boosting the performance of our emulator. We then evaluate the model on the holdout set to ensure its performance generalises to entirely unseen data (since we selected model weights which perform best on the validation set, the validation set itself is not a good test of out-of-sample performance).

Table 3.1: The GALFORM parameters under investigation. See § 2.1 for the equations which define the symbols in the first column.

Parameter	Range	Process
$f_{\rm stab}$	0.61 – 1.1	Disk instability
$\alpha_{\rm cool}$	0.2 – 1.2	AGN feedback
$\alpha_{\rm ret}$	0.2 – 1.2	SN feedback
$\gamma_{\rm SN}$	1.0 – 4.0	SN feedback
$V_{\rm SN,\ disk}$ [kms^{-1}]	100 – 550	SN feedback
$V_{\rm SN,\ burst}$ [kms^{-1}]	100 – 550	SN feedback
$f_{\rm burst}$	0.01 – 0.3	Mergers
$f_{\rm ellip}$	0.2 – 0.5	Mergers
$\nu_{\rm SF}$ [Gyr^{-1}]	0.2 – 1.7	Quiescent star formation
$f_{\rm SMBH}$	0.001 – 0.05	SMBH accretion

3.2.1.1 Inputs and outputs

The aim of our emulator is to map an input vector $\mathbf{x}$, the GALFORM parameters, onto an output vector $\mathbf{y}$, the statistical galaxy properties that we wish to predict. Our choice of input parameters is informed by previous analyses (e.g. Lacey et al., 2016; Oleśkiewicz & Baugh, 2020), and we aim to emulate the effects of the parameters associated with the key processes outlined in §2.1. These parameters and their ranges are shown in Table 4.1. We train our emulator to predict a wide range of statistical galaxy properties calculated from the output of GALFORM. These are the K- and r-band LFs at $z = 0$, the early- and late-type galaxy sizes, the HI mass function, the early-type fraction with r-band magnitude, the I-band Tully-Fisher relation, the bulge-black hole mass relation, and the metallicities of early-type galaxies.

3.2.1.2 Model architecture

We find that a simple architecture is sufficient to accurately emulate GALFORM. We use a densely-connected neural network, meaning that every neuron is connected to every neuron in the previous layer. We use two hidden layers, each with 512 neurons and sigmoid activation functions, and linear activations on the output layer. We investigated a number of other architectures, such as stacking long short term memory (LSTM; Hochreiter & Schmidhuber 1997) and 1D convolutional layers to try to exploit features of the data, but found limited improvement at the cost of slower evaluation speed.

3.2.1.3 Ensembling

Training a neural network is a stochastic process. The network weights are often initialized according to some distribution (e.g. Glorot & Bengio, 2010), and the optimizer traverses the weight space using gradient steps calculated on mini-batches of the full dataset (i.e. a small subset of the whole training set at a time), and so is inherently stochastic. This means that training a single network is sub-optimal. Since the error surface is likely to contain many local minima we are unlikely to find the best possible network weights with one network alone, and each network will develop its own idiosyncrasies in how it fits the data. Neural networks also contain a vast number of parameters, and are therefore prone to over-fitting. One way to address these problems is ensembling. This involves training a handful of networks with different weight initializations and combining the individual predictions. We can also shuffle the validation and training sets for each model in the ensemble, so that each model is exposed to a different distribution of input-output pairs. In general, this allows for a more robust prediction. Individual models may over- or under-fit to different features of the data, and combining predictions averages over these individual behaviours.

We therefore train 10 models as described above, each with the same model architecture. Our emulator is then the simple average of the predictions of this ensemble of models. We must note however that this is a rich avenue for exploration in future work (for a review of popular ensembling methods, see Maclin & Opitz, 2011). For example, it may be possible to ensemble different machine learning algorithms and combine the individual model predictions with a weighting scheme, or even another machine learning algorithm.

3.2.2 Sensitivity analysis

Once we have trained a deep learning emulator of GALFORM, we can apply sensitivity analysis techniques (e.g. Saltelli et al., 2010; Saltelli, 2017) to understand the contribution of the different parameters to the bin-wise variance in the emulator outputs. For a full description of calculating sensitivity indices see Oleśkiewicz & Baugh (2020), who first applied this type of analysis to a model of the entire galaxy population. Here we provide a brief overview of the sensitivity indices and what they describe.

Since GALFORM is deterministic, all the variance in the output Y will be due to the effects of the input parameters X. Assuming the input parameters are independent, we can calculate the first-order variance due to parameter X_i by integrating the variance over the i-th dimension. Furthermore, we can calculate the variance due to interactions between parameters X_i and X_j by integrating the variance across the i-th and j-th dimensions, and subtracting the corresponding first-order effects of parameters X_i and X_j. This can be continued to account for the interactions between many parameters.

As explained in Oleśkiewicz & Baugh (2020), assuming Y is a scalar output of our model (for example, the value of a luminosity bin in our K-band luminosity function) and assuming that the input parameters X to the model are independent, the variance attributable to the model parameter X_i can be calculated as:

$$E_i = E_{X\sim i}(Y|X_i) = \frac{\int_{X_i} Y(X_i)\,\mathrm{pdf}(X_i)\prod_{i\neq j} dX_j}{\int \mathrm{pdf}(\mathbf{X})d\mathbf{X}}, \tag{3.4}$$

$$V_i = \mathrm{Var}_{X_i}(E_i) = \frac{\int_{X_i}(E_i - E(Y))^2\,\mathrm{pdf}(X_i)dX_i}{\int \mathrm{pdf}(\mathbf{X})d\mathbf{X}}, \tag{3.5}$$

where V_i is the variance integrated over dimension i, and E_i is the mean Y value calculated by integrating in all dimensions except i, as indicated by $E_{X\sim i}$. In our case, we can assume that $\mathrm{pdf}(X_i)$ is a uniform distribution. We can then define the first-order sensitivity indices S_i as

$$S_i = \frac{V_i}{\mathrm{Var}(Y)}, \tag{11}$$

calculating the first-order variance constribution of parameter X_i divided by the total model variance $\mathrm{Var}(Y)$. We can then measure the variance due to second order interactions between parameters by calculating the variance integrated across dimensions i and j, and subtracting the first order variances

$$V_{ij} = \mathrm{Var}_{X_{ij}}\left(E_{X\sim ij}(Y|X_i, X_j)\right) - V_i - V_j, \tag{3.6}$$

and $S_{i,j}$ can be calculated in the same way as S_i. The total variance of the model output Y can therefore be decomposed as

$$\sum_{i=1}^{d} \mathrm{Var}_i + \sum_{i<j}^{d} \mathrm{Var}_{i,j} + ... + \mathrm{Var}_{1,2...d} = \mathrm{Var}(Y) \tag{3.7}$$

where Var_i represents the variance due to the i-th of the d parameters, the sum over $\mathrm{Var}_{i,j}$ represents the variance due to interactions between the parameters X_i and X_j, and $\mathrm{Var}(Y)$ is the total variance in the model output Y. This can be normalised to give the sensitivity indices of all orders

$$\sum_{i=1}^{d} \mathrm{S}_i + \sum_{i<j}^{d} \mathrm{S}_{i,j} + ... + \mathrm{S}_{1,2...d} = 1. \tag{3.8}$$

This can be separated into S_1, the first order sensitivity index, which describes the proportion of the variance due to the i-th parameter, and S_T, which encapsulates the proportion of variance due to the i-th parameter and all higher order interactions between the i-th parameter and all other parameters.

Given the low computational cost of our emulator, we can evaluate it at a large number of points in the parameter space following Saltelli sampling. This sampling method aims to both evenly sample the space and minimise the model discrepancy (a concept whose full explanation is beyond the scope of this work, but is described in Saltelli et al., 2010), allowing for sample-efficient calculation of the sensitivity indices. For this analysis, we use the `SALib` python package (Herman & Usher, 2017).

3.2.3 Calibration and comparison datasets

We will use our emulator to calibrate `GALFORM` using a number of datasets. For the most part, we adopt the datasets used for model calibration in Lacey et al. (2016), but with a focus on low-redshift observations. The key change we make is to the choice of LF data. We use the K- and r-band LFs from the GAMA survey (Driver et al., 2012); we choose these datasets as they correspond to the same survey volume and the same analysis methods are used for each band, with consistent k-corrections to $z = 0$ bands. The measured LFs should therefore be as consistent as possible, allowing our model to fit both. We apply a number of selection criteria to the `GALFORM` output to replicate the observational samples of the calibration datasets.

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

1. For the K-band LF, we calibrate to data from Driver et al. (2012) and also compare to data from Kochanek et al. (2001).

2. For the r-band LF, we calibrate to Driver et al. (2012).

3. For the early- and late-type sizes, we calibrate to data from Shen et al. (2003). We define early types in the model as galaxies with bulge-to-total r-band luminosities of $(B/T)_{\mathrm{r}} > 0.5$ and late types as $(B/T)_{\mathrm{r}} < 0.5$. Since the half-light radii of late-type galaxies are measured in circular apertures

projected on the sky, the late-type galaxy sizes are corrected to face-on values by multiplying the median sizes by a factor of 1.34 (as in Lacey et al., 2016).

4. For the HI mass function, we calibrate to data from Zwaan et al. (2005) and compare to the estimate from Martin et al. (2010).

5. For the early-type fraction, we calibrate to data $(B/T)_r$ derived from Moffett et al. (2016) (A. Moffett, private communication). Here, the $(B/T)_r$ ratio was calculated from GAMA using the disk/bulge decomposition method outlined in Lange et al. (2016). We also compare to data from González et al. (2009), which uses concentration indexes calculated from SDSS data (York et al., 2000b). Again, early types are defined to have $(B/T)_{\rm r} > 0.5$.

6. For the I-band Tully-Fisher relation we compare to a subsample of Sb-Sd galaxies from the Mathewson et al. (1992) catalogue, as selected by de Jong & Lacey (2000). Model galaxies are selected with $(B/T)_{\rm B} < 0.2$ and gas fractions $M_{\rm cold}/M_* > 0.1$, where $M_{\rm cold}$ is the cold gas mass and M_* is the stellar mass.

7. For the Bulge-BH mass relation, we compare to data from Häring & Rix (2004). To match the bias toward early-types in the sample, we choose model galaxies with $(B/T)_{\rm r} > 0.3$.

8. For the early-type metallicity, we compare to data from Smith et al. (2009). We select model galaxies which reside in dark matter halos with $M_{\rm halo} > 10^{14}h^{-1}M_\odot$ and define early-types as before. The observed metallicities are corrected for metallicity gradients as described in Lacey et al. (2016).

9. Finally, we explore the model predictions for data in a very different redshift range to our calibration datasets. We test the calibrated model predictions against observational estimates of the star formation rate density (SFRD) with redshift. We compare to data from Burgarella et al. (2013); Cucciati et al. (2012); Oesch et al. (2013); Sobral et al. (2013) and Gunawardhana

et al. (2013). Since the observationally derived SFR values depend on an assumed initial mass function, and our model assumes a mildly top-heavy initial mass function in starbursts, we account for this in the observational comparison by applying an approximate correction in which we weight the starburst SFR by a factor of 1.9 (see Lacey et al. (2016) for further details).

3.2.4 Parameter fitting

Once we have trained our emulator, we use Markov-Chain Monte Carlo (MCMC) to explore the effect of calibration to different datasets with a simple implementation of the Metropolis-Hastings algorithm (e.g. Robert, 2016). The complication here is that the observational errors on the datasets cannot be combined straightforwardly. For example, if we aimed to minimise χ^2, and the error bar on a particular data point in the constraining observational dataset was very small, this point would dominate the total error measure. Our MCMC chain would simply be trying to find the best fit to this one data point, without fitting to the others. We therefore aim to minimise the absolute error between the emulator output and the observational constraints, without considering the observational errors. This allows us to combine and fit to multiple datasets, without having to worry about the robustness of the associated observational error bars, and hence to avoid the complications described above.

We also wish to have the flexibility to give more consideration to a selected observational constraint over the others. This will allow us to investigate the effect of requiring better fits to some datasets, and to see how this affects the fit to other datasets, as well as how the optimal parameter choices change as a result. We therefore include a vector of heuristic weights, $\mathbf{W}$, which can be varied to increase the contribution of the residuals from one constraint to the total error,

$$\mathrm{MAE}_{\mathrm{obs}} = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{n_i^{\mathrm{obs}}} W_i |\hat{\mathbf{y}}_i - \mathbf{y}_i^{\mathrm{obs}}|, \tag{3.9}$$

where the sum is over the n observational constraints, W_i represents the weighting of the contribution to the total error of the i-th constraint, $\hat{\mathbf{y}}_i$ represents the emulator prediction for a set of model parameters, and $\mathbf{y}_i^{\text{obs}}$ is the observational data for the i-th constraint with n_i^{obs} datapoints. Since $\hat{\mathbf{y}}_i$ and $\mathbf{y}_i^{\text{obs}}$ are vector quantities, the modulus signs represent the L1 norm. As the constraining datasets have a variety of values, we scale each one by a constant factor and apply a constant offset so that the range of each $\mathbf{y}_i^{\text{obs}}$ is [0,1]. We apply the same scaling to the emulator predictions $\hat{\mathbf{y}}_i$ before calculating Eqns. 3.2 and 3.9. Note that since different datasets contain different numbers of datapoints, we divide the i-th dataset's error by the number of datapoints n_i^{obs} so that each contributes equivalently to the mean error. In later sections when considering observational data, we shall refer to Eqn. 3.9 as just the mean absolute error (MAE). We have checked that using the more common L2 norm instead of L1 moves attention to outliers and degrades the overall performance of the emulator.

We implement the Metropolis-Hastings algorithm as follows: we initialize each chain at a (different) random point in the parameter space, $\mathbf{x}$. We then draw the next sample in the chain, $\mathbf{x}'$, from independent Laplacian distributions, $\mathcal{L}(x_i'|\mu_i, b_i) = \frac{1}{2b_i}\exp(-|x_i' - \mu_i|/b_i)$ with $\mu_i = x_i$ and the scale parameter for the i-th model parameter, b_i, taken to be 1/20th of the parameter ranges given in Table 4.1. We then calculate the *acceptance ratio*, α, by taking the likelihood ratio of the emulator predictions to the observational data for the parameter sets $\mathbf{x}$ and $\mathbf{x}'$ under a Laplacian likelihood with scale parameter $b_{\text{obs}} = 1/20$ (i.e. the ratio $\mathcal{L}(f_*(\mathbf{x}')|\boldsymbol{\mu}, b_{\text{obs}})/\mathcal{L}(f_*(\mathbf{x})|\boldsymbol{\mu}, b_{\text{obs}})$, where $\boldsymbol{\mu}$ represents the values of the observational data and $f_*(\cdot)$ the emulator, and recalling we are using the modified absolute difference given in Eqn. 3.9). We next generate a uniform random number $u \in [0, 1]$; samples are *accepted* if $u \leq \alpha$, in which case we draw the next sample from Laplacians centered on $\mathbf{x}'$, or *rejected* if $u > \alpha$, in which case we draw the next sample from Laplacians centered on the original point $\mathbf{x}$. Therefore, if the error between the emulator predictions for the parameter set $\mathbf{x}'$ and the observational data is less than or equal to the error for the predictions for $\mathbf{x}$, we accept the sample. If the error for $\mathbf{x}'$ is not an improvement over the previous sample, we accept it with probability α. The density of accepted samples should then trace the regions in the parameter space which give the best fits to the observational data. We discard the first 50% of accepted samples to allow for burn-in. We test a number of values of the sampling Laplacian widths b_i in the range $0.05 - 0.2$, in conjunction with the likelihood width $b_{\rm obs}$, and find that these parameters have little effect on the convergence of the chains, and larger $b_{\rm obs}$ simply increases the proportion of accepted samples. We ran longer chains up to 100,000 samples and found that they quickly converged to their minimum MAE (as given by Eqn. 3.9) within the first 10,000 samples, and so choose this as our chain length.

3.3 Results

Here we present our main results, starting with a demonstration of the accuracy of the emulator (§ 3.3.1), a sensitivity analysis of the model parameters (§ 3.3.2), and closing with a discussion of the calibration of the model parameters and the tensions that arise when using different combinations of datasets (§ 3.3.3).

3.3.1 Emulator performance

Having trained our emulator as described in § 3.2.1, we evaluate its ability to predict the output of `GALFORM` at unseen points in the parameter space. We use a set of 930 `GALFORM` runs. The emulator was trained as described in § 3.2.1 with 80% of the runs used as the training set (i.e. 744 combinations of parameter values), a 93 sample validation set, and a 93 sample holdout set. For each model in the emulator ensemble (i.e. each version of the neural network), the training and validation sets were shuffled. Fig. 3.2 shows the emulator prediction vs. the true `GALFORM` output for the holdout set. Generally, the emulator follows a tight relation on the $y = x$ line, indicating that the emulator is accurately predicting the `GALFORM` output for

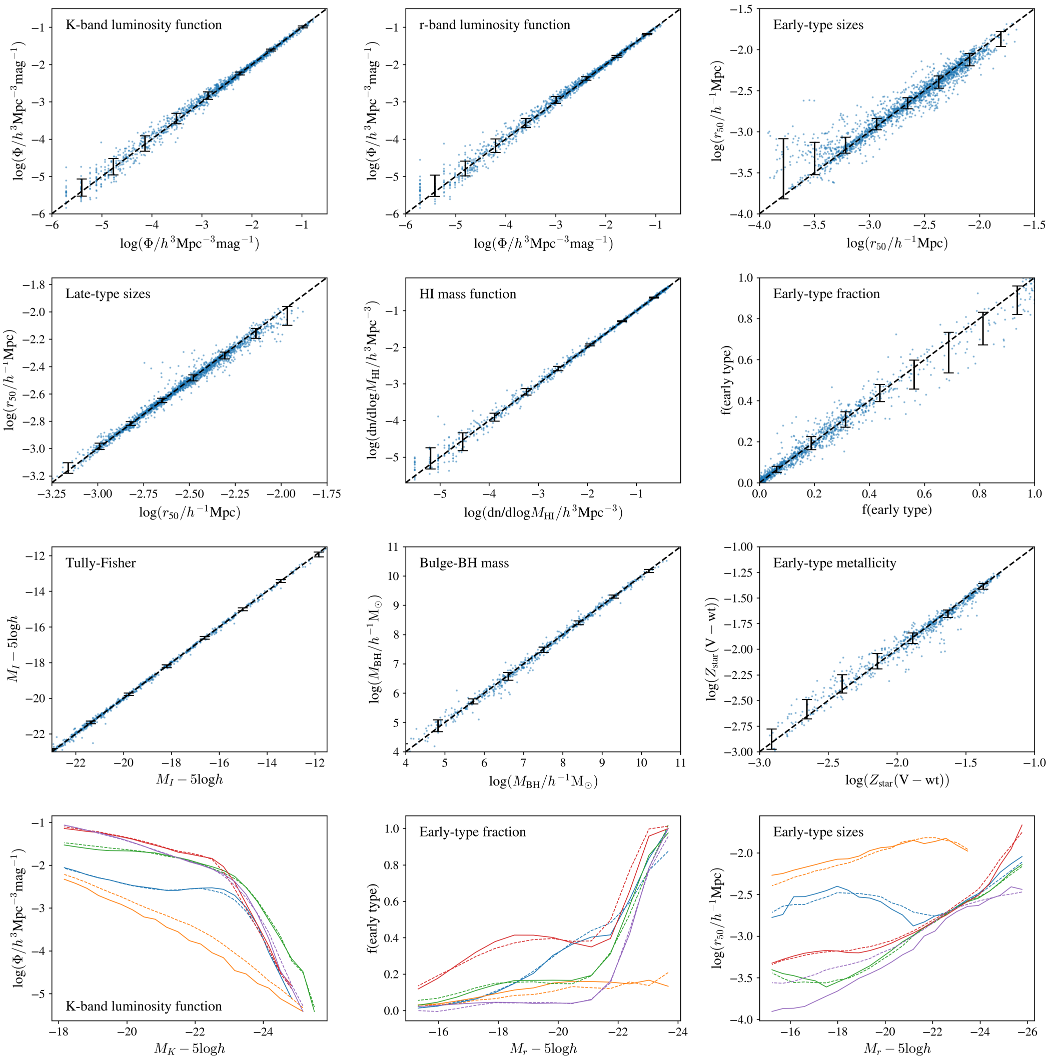


Figure 3.2: Emulator performance across nine statistics computed from the model output for out-of-sample parameter sets. These statistics are either number densities or median values in luminosity or mass bins, and are the same ones used for the observational comparisons. The first three rows show the emulator output (y-axis) vs. the `GALFORM` output (x-axis). Black error bars indicate the 10-90th percentile range of the residuals. The bottom row shows a random draw of emulator outputs (*dotted*) and true `GALFORM` outputs (*solid*) for the K-band LF, early-type fraction, and early-type sizes, reading from left to right. In these panels different colours denote different parameter sets.

the parameters sets in the holdout set. The HI mass function, Tully-Fisher relation, and Bulge-BH mass relations are accurately predicted, as well as the faint end of the luminosity functions and late-type galaxy sizes. The uncertainty is greater for the predictions of the bright-end of the LFs, and for the early-type sizes, fraction of early-type galaxies with luminosity, and the early-type metallicity. The lower panel of Fig. 3.2 sheds some light on the source of inaccuracies in the early-type predictions, notably the early-type sizes, which exhibit noisy behaviour for some choices of parameters, and for a few cases (e.g. the purple line) the lower luminosity sizes are not well predicted. For the early-type fraction, while the error bars look large, inspection of the lower panels shows that such errors are generally in the brighter bins. We are nevertheless able to discriminate between parameter sets at the fainter magnitudes as the overall shape is well captured.

We see that the emulator is able to characterise a wide range of behaviour in the LFs, with the majority tightly predicted. In the bottom row of Fig. 3.2, the orange curves in the K-band panel show a substantial discrepancy between the true and predicted outputs; this usually indicates that the training data did not contain sufficient examples of this behaviour. The emulator constructs the function $f_*(\cdot)$ by fitting to the training examples, and in doing so should build a function which can interpolate between points in the parameter space. However, in sparsely sampled regions of the space, such as at the edges of our parameter bounds, the interpolation is less reliable. Indeed, if a point in the holdout set is an extrapolation with respect to the training set, performance can be affected. This is why we aim to fill the parameter space as evenly as possible using the Latin hypercube sampling method. We expect that such disagreements will decrease on increasing the number of training samples.

We can also judge from the distribution of predictions for the K- and r-band LFs in Fig. 3.2 that the emulator slightly over-predicts the bright end of the LF. This is a consequence of the emulator training; in the interest of computing speed, we run `GALFORM` on only a sub-region accounting for 1% of the full volume of the

P-MILL simulation. This leads to sampling effects at low galaxy number densities, and for different choices of parameters the LF is cut off at different luminosities. Since the output of our emulator must be fixed-length, during training we mask any points beyond this luminosity cut-off when computing the loss. This means that in the brighter luminosity bins the emulator is only fitting to a small number of runs which are biased towards having higher values of ϕ in these brighter bins. There is therefore a tendency to over-predict at these luminosities. This should only be a minor problem in terms of our fitting routine, since the Driver et al. (2012) data we are fitting to does not sample ϕ to very low number densities. We also see a quantisation effect in the brighter LF bins, again due to the discrete sampling of galaxies. These problems could be removed by evaluating `GALFORM` on a larger fraction of the P-Mill simulation volume, though this would be more computationally expensive.

3.3.1.1 Scaling with training set size

We train three emulators with 250, 500 and 750 samples of parameters respectively (split into training and validation sets with 10% of the samples being used for validation) to investigate the scaling of the emulator performance with the number of full `GALFORM` calculations carried out. The emulators consist of an ensemble of 10 networks each trained on the same (shuffled) training and validation data and the same holdout set of 93 samples. Fig. 3.3 shows the scaling of the emulator performance on the holdout set (as measured by the MAE) with the number of training samples N. The dashed line shows average performance of the individual networks, and the solid line shows the performance of the ensemble. The model scales well with increasing training samples, and ensembling affords an almost constant improvement in performance (we find that at ~ 10 models, the performance increase from adding more models to the ensemble saturates).

We test the ability of the emulator to generalise to unseen data by evaluating the version of the emulator trained with 500 samples in Fig. 3.3 on the remaining

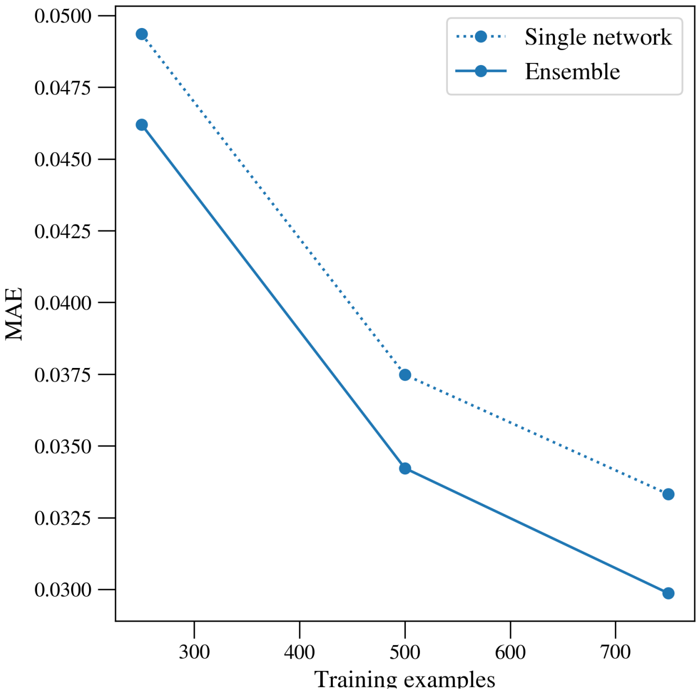


Figure 3.3: Emulator mean absolute error with the number of training examples of full `GALFORM` runs for the ensemble (solid line) and single (dotted line) networks. The emulators were trained on 250, 500 and 750 samples and performance evaluated on the same holdout set of 93 samples. Recall that the emulator outputs are scaled as described in § 3.2.4.

430 unseen samples. We find very little variation in the accuracy of the model between the two holdout sets. The MAE on the 93 sample holdout set was 0.034, and on the full 430 available holdout samples was 0.032. Further, we perform a 10-fold cross validation with the training, validation and holdout sets as described in § 3.3.1. We find a mean MAE of 0.030, with a range between 0.027 and 0.034. This gives us confidence that the model is able to learn a function which provides a very good approximation to `GALFORM` across the full parameter space.

The impressive scaling of the emulator error with number of training samples is encouraging. SAMs are used to build mock catalogues for upcoming surveys, and some of these have stringent requirements on fits to certain datasets, such as the redshift distribution of galaxies. We can envisage using this technique to produce high accuracy parameter estimates for fits to such datasets by increasing the number of training samples, or using ‘zoom-in’ training samples as in previous work (e.g. Bower et al., 2010) to focus in on a particular region of parameter space which is deemed to give acceptable fits to the constraining datasets. Nevertheless, we find that our current emulator is sufficiently accurate to facilitate calibration and model exploration.

3.3.2 Sensitivity Analysis

We apply the techniques described in § 3.2.2 to calculate the contribution of each parameter to the variance in each bin of the 9 constraints. The results are shown in Fig. 3.4. The open circles indicate the first order sensitivity index, S_1, which quantifies the proportion of the variance due to just one parameter. The total order sensitivity, S_T, is shown as solid lines, and indicates the proportion of the variance contributed by one parameter and its interactions with the other parameters. We can interpret the difference between the first order and total order sensitivity as a measure of the strength of the interaction between a given parameter and the other parameters. For clarity, we exclude parameters which never contribute more than 10% of the variance to any bin. Both f_{burst} and f_{ellip} meet this condition, and so do not appear in the plots.

We see that the dominant parameters for the majority of the model outputs are, perhaps unsurprisingly, the supernova feedback parameters. $V_{\mathrm{SN,\,disk}}$ and γ_{SN} account for the majority of the variance at the faint end of the K- and r- band LFs. At the bright end, α_{cool}, the parameter governing the strength of AGN feedback, contributes the largest proportion of the variance. The majority of the variance in the late- and early-type sizes, the Tully-Fisher relation, as well as the HI mass function is also contributed largely by the same two or three parameters.

The early-type fraction is dominated by the threshold for disk instability, f_{stab}, up until $M_r - 5\log h \approx -21$. At brighter magnitudes, disk instabilities become unimportant as mergers takes over as the main channel for building spheroidal components (see Huško et al., 2021, for an exploration of the relative importance

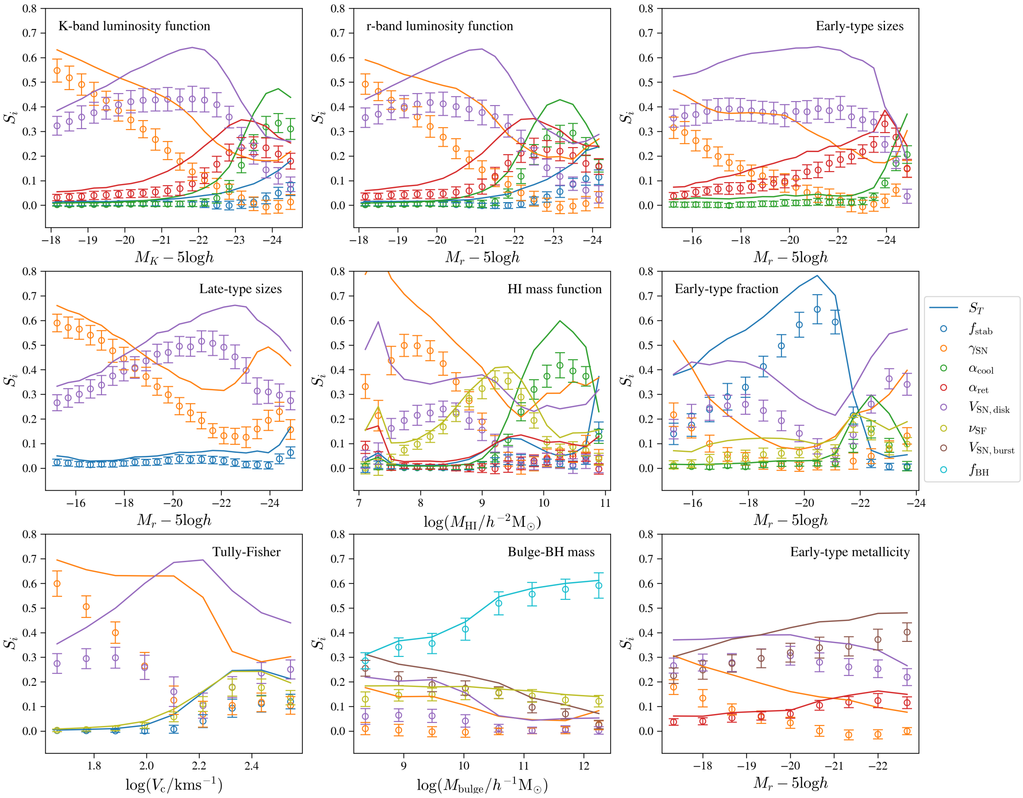


Figure 3.4: The emulator sensitivity to different parameters for each of the observables considered in this work; each panel shows a different observable, as labelled. Open circles indicate S_1 as described in the text, and solid lines represent S_T. For clarity, error estimates are shown for the S_1 calculation but not for S_T, although they are similar. Sensitivities for parameters which never exceed more than 10% of the variance in any bin are not plotted.

of different channels for the growth of galaxy stellar mass).

The sensitivity analysis hence dispels one of the myths surrounding SAMs as it shows that the model cannot be made to fit to any arbitrary combination of datasets. To match the faint end of the K-band LF, we are strongly constrained in our choice of supernova feedback parameters, which contribute the vast majority of the variance to these bins. Our predictions of early- and late-type galaxy sizes, the HI mass function, the Tully-Fisher relation, and the bright end early-type fraction are also then largely constrained, since the supernova feedback parameters dominate these outputs too. This is in line with the analysis performed by Bower et al. (2010), which reached similar conclusions.

The parameters also have clear physical interpretations, and are analogous to the parameters used in the subgrid physics models in hydrodynamic simulations (e.g. Crain et al., 2015; Weinberger et al., 2016; Pillepich et al., 2017). The parametric model for supernova feedback can indeed be tuned to give a good match to the late-type galaxy sizes, but in doing so we are strongly constraining our fits to other datasets; the model does not include arbitrary parameters which allow for fine-tuning to an individual dataset without physical motivation or consequences for the fits to other datasets.

3.3.3 Calibration and dataset tensions

We now apply the methods described in § 3.2.4 to calibrate the model to the datasets described in § 3.2.3, focusing on uncovering any tensions that exist between datasets. First, we aim to replicate a known tension in the model discussed in Bower et al. (2010) and Lacey et al. (2016). This is the tension between reproducing late-type galaxy sizes and the galaxy LFs; these datasets have been found to prefer different values for the supernova feedback parameters. We can investigate this by adjusting the weightings applied to the residuals between our emulator prediction and each dataset (as in Eqn. 3.9), and then performing an MCMC parameter search to see how the best-fitting parameter choices respond.

In Fig. 3.5, we show the emulator predictions for three sets of best-fitting parameters. In the first case, shown by the blue line, we weight only the residuals for the K-band LF. For the orange line, we weight only the size-luminosity relation for late-type galaxies, and the green line shows the results when weighting both datasets equally (i.e. both datasets have equal influence over the best-fitting parameter values). The shaded region is shown only around the fit to the K-band LF for clarity, and represents the 10-90th percentile error of the emulator when predicting similar values in the holdout set (this gives a rough idea of the uncertainty of the emulator, but is certainly not an exact measure). We can clearly see the tension between these two datasets uncovered in an automatic and objective way;

matching the sizes of faint late-type galaxies leads to an over-prediction of the LF at all luminosities by up to an order of magnitude. When matching both the K-band LF and the late-type galaxy sizes, we see an over-prediction in the faint-end of the LFs, and the sizes of faint late-types are over-predicted by a factor of ~ 2. The early-type sizes and Tully-Fisher relation are also shown in Fig. 3.5. Although no weighting was applied to these datasets in this exercise, we can see improved matches emerge naturally when we fit to the late-type galaxy sizes. We can gain some intuition for this behaviour from Fig. 3.4. As discussed, the Tully-Fisher relation, early- and late-type galaxies sizes, and the faint-end of the galaxy LF are highly sensitive to the choice of supernova feedback parameters, $\gamma_{\rm SN}$ and $V_{\rm SN,\, disk}$ (which together account for $\sim 90\%$ of the variance in the faint-end LFs and the sizes of faint late-type galaxies). Therefore we might expect that some tension would arise in trying to fit to a number of the above datasets at the same time.

It is also informative to investigate how the acceptable regions of parameter space change as we introduce weightings to other datasets. We demonstrate this for the tension between the LF/late-type sizes in Fig. 3.6. The shaded regions represent accepted samples from our 20 MCMC chains, each 10,000 steps in length, with the first 50% of each chain discarded to allow for burn-in. The red region corresponds to a fit to the K-band LF, and the blue region to fits to both the K-band LF and late-type galaxy sizes. The shading gives a sense of the density of accepted samples i.e. the darker colours correspond to the more favoured parts of parameter space in this projection. The darkest regions correspond to the 25th percentile, and the lighter regions to the 50th and 75th percentiles. Also shown in Fig. 3.6 are 1D histograms of the density of accepted samples. We find that, as in previous analyses, a reasonably large range of parameter values result in acceptable fits to a given constraint. This can be best understood (as explained in Bower et al., 2010) as the effect of the high dimensionality of the parameter space; though when plotted in projection down to 1 or 2 dimensions the space appears widely sampled, the higher dimensional acceptable region is reduced significantly. Also, some of the parameters

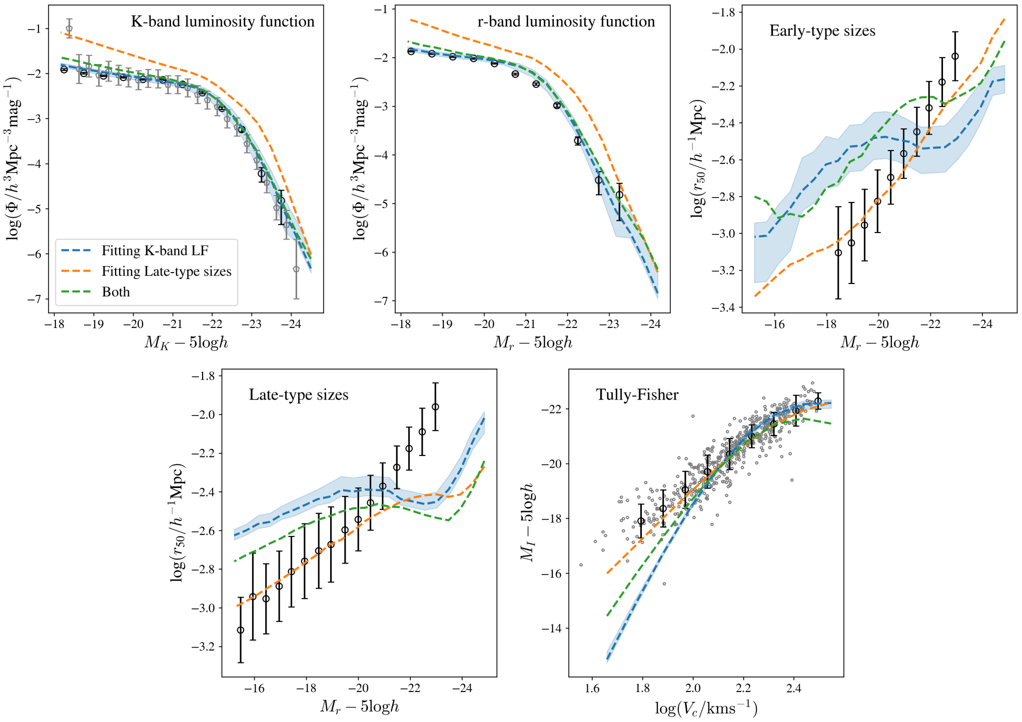


Figure 3.5: A comparison of the emulator predictions for fits to the K-band luminosity functions, the late-sizes, and a combination of the two (represented by different colour dashed lines). We fit to the data from Driver et al. (2012) (*black*) for the K-band LF, and Shen et al. (2003) for the late-type sizes. The emulator predictions correspond to the best fit found from 20 MCMC chains, each 10,000 steps in length. The blue shading represents the 10-90th percentile errors when predicting a similar value in the holdout set. The black and grey datapoints represent the calibration data described in §3.2.3. For the K-band LF, we also compare to data from Kochanek et al. (2001) (*grey*). For the r-band LF, we compare to data from Driver et al. (2012). For the early-type sizes we compare to data from Shen et al. (2003), and for the Tully-Fisher relation we compare to data from de Jong & Lacey (2000).

produce degenerate effects (see for example Fig. A.1 in Appendix A, where we show the degenerate effects of the f_{stab} and $V_{\mathrm{SN,\,burst}}$ parameters). Nevertheless, we see that the K-band LF fit prefers somewhat higher values of $\gamma_{\mathrm{SN}} \approx 3.6$ and lower values of $V_{\mathrm{SN,\,disk}} \approx 200\mathrm{kms}^{-1}$, in contrast to the fit to both the K-band LF and late-type sizes, where we find a preferred value of $\gamma_{\mathrm{SN}} \approx 2.3$ and $V_{\mathrm{SN,\,disk}}$ at the top of the explored range at $\sim 550\mathrm{kms}^{-1}$. Interestingly, there seems also to be a preference for lower values of ν_{SF} to match the late-type galaxy sizes. We can understand this crudely by investigating the first-order effect associated with the ν_{SF} parameter.

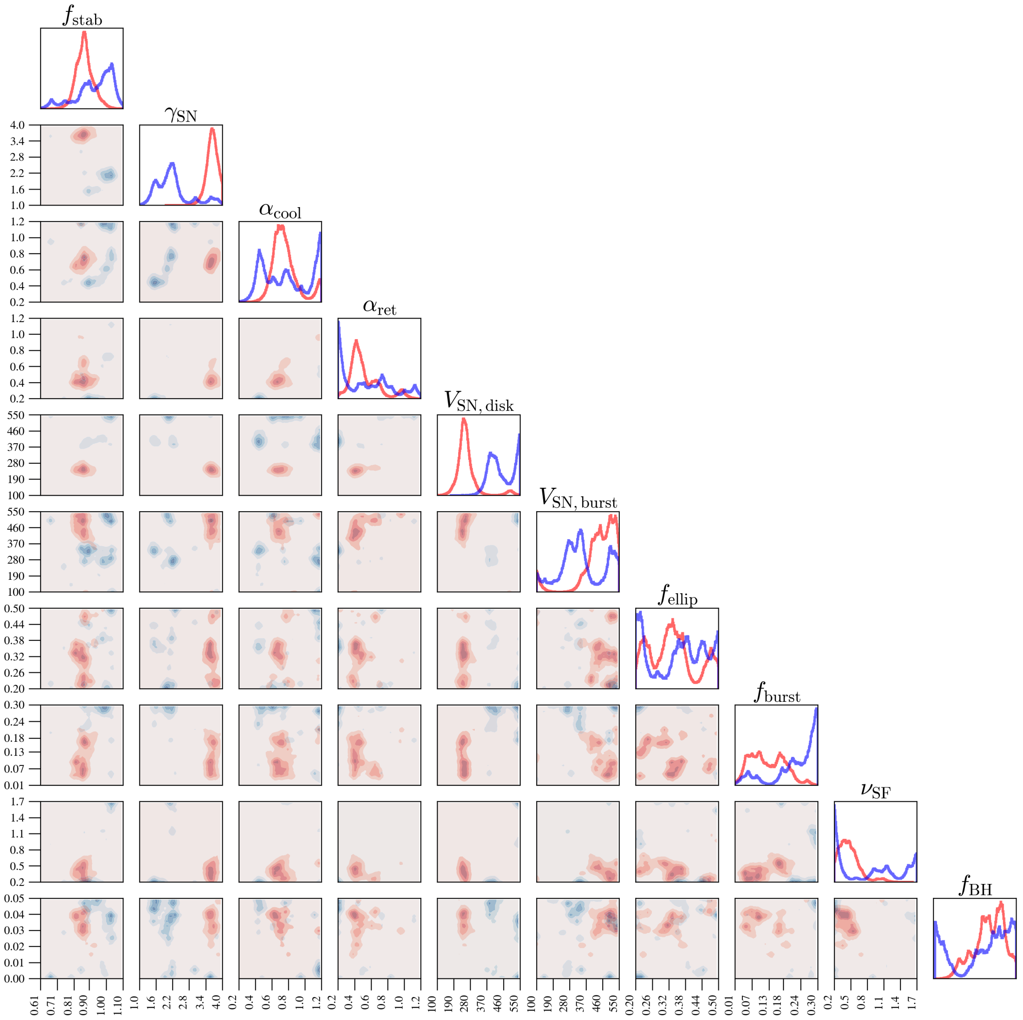


Figure 3.6: Accepted samples from 20 MCMC chains for fits to the K-band LF (*red*), and both the K-band LF and the late-type galaxy sizes (*blue*). The first 50% of samples were discarded to allow for burn-in. The histograms show the marginalised distribution of the parameters. The ranges on each axis are the same as those quoted in Table 4.1. The shading gives a sense of the density, 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.

Inspecting Fig. A.2 in Appendix A, we see that the $\nu_{\rm SF}$ parameter has a some effect on the bright-end of the K-band LF. This counteracts the enhancement from the higher value of $V_{\rm SN,\,disk}$, and also marginally improves the fit to the late-type galaxy sizes.

Another tension arises between the HI mass function and the bright end of the K- and r-band LFs. This is shown in Fig. 3.7. As before, the blue line corresponds

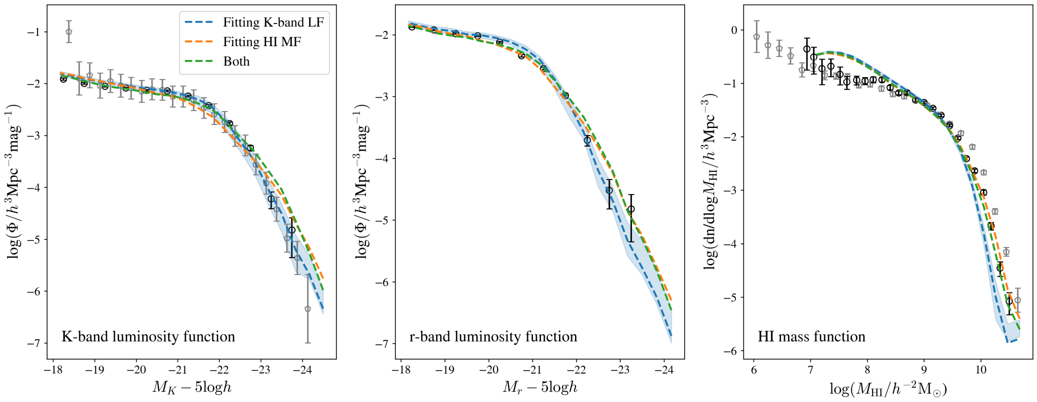


Figure 3.7: A comparison of the emulator predictions for fits to the K-band luminosity functions, the HI mass function, and a combination of the two (represented by the different colour dashed lines). The black and grey datapoints represent the calibration data described in §3.2.3. For the HI mass function, we fit to data from Zwaan et al. (2005) (*black*) and include data from Martin et al. (2010) (*grey*) for comparison.

to the fit to the K-band LF alone, the orange line to the fit using the HI mass function alone, and the green to a fit to both datasets. We can again propose (from our plot of the sensitivity indices, Fig. 3.4) that the main cause of this discrepancy is a tension in the choices for the AGN feedback parameter, α_{cool}, and the supernova feedback parameters. Indeed, when fitting the observational constraints individually, the fit to the K-band LF prefers a higher value for the AGN feedback parameter, with $\alpha_{\mathrm{cool}} \approx 0.8$, whereas the fit to the HI mass function prefers $\alpha_{\mathrm{cool}} \approx 0.5$. We can also investigate how calibrating to both datasets shifts the parameter values. We do this as before with an MCMC exploration of the parameter space (see Fig. A.3 in Appendix A). Fitting to both the K-band LF and the HI mass function (as compared with a fit just to the K-band LF) causes a shift in the preferred $V_{\mathrm{SN,\,disk}}$ to higher values. ν_{SF}, the parameter which controls the rate of quiescent star formation, shifts to the lowest values in the explored range, and the parameter α_{ret}, which is involved in gas return to halos following supernova feedback, becomes more strongly peaked, with the peak shifted to slightly higher values.

To understand this further, we investigate the first-order effects of the parameters

($V_{\rm SN,\,disk}$, $\nu_{\rm SF}$, and $\alpha_{\rm ret}$), perturbed around the fit to the K-band LF. We show the results in Fig. 3.8. We vary the parameters individually ('one-at-a-time') across their explored range, with lighter colors corresponding to lower parameter values. We can begin to understand the changes in the preferred parameter choices in terms of these transformations. When fitting both the HI mass function and the K-band LF, we find that there is a slight over-prediction of the bright-end of the LF. From these one-at-a-time plots we can see that the increase in $V_{\rm SN,\,disk}$ causes an over-prediction at the bright-end of the LF, and a reduction in amplitude at the faint-end, but more accurately matches the high-mass end of the HI mass function. The HI mass function can be better matched at intermediate masses by a decrease in $\nu_{\rm SF}$. In GALFORM, reducing $\nu_{\rm SF}$ has the effect of decreasing the rate of quiescent star formation in disks. As a result, lower values of this parameter provide a better fit to intermediate masses of the HI mass function, while simultaneously reducing the number density of the most luminous galaxies in the K-band LF, and so counteracting the enhancement due to the increase in the $V_{\rm SN,disk}$ parameter. We can further improve the match of the prediction for the LF to the observational data by increasing $\alpha_{\rm ret}$, which has little impact on the HI mass function but reverses some of the 'flattening' of the LF caused by the increase in $V_{\rm SN,disk}$. In previous galaxy formation models, using the WMAP-7 cosmological parameters, this tension has not been so apparent, but can also be seen between the $b_{\rm J}$-band LF and the HI mass function in Baugh et al. (2019).

Our approach also allows us to uncover a significant tension between the bright end of the LFs, the early-type fraction, the HI mass function, and the early-type metallicity. We demonstrate this in Fig. 3.9, where we compare a fit found by calibrating to the K-band LF, HI mass function, and the early-type fraction with and without including the early-type metallicity constraint (note that we do not fit to datasets shown in grey). Including the early-type metallicity has a significant effect on the best-fitting parameter values; it generally improves the fits to the galaxy sizes, and degrades the fit to the early-type fraction (at least when considering the

Moffett et al. (2016) data) and the HI mass function. We investigate the impact on the acceptable region of parameter space in Fig. 3.10, where we show the key changes induced by including the early-type metallicity constraint. The red region shows the fit to the K-band LF, HI mass function, and early-type fraction, and the blue region also includes the early-type metallicity. We find that there is a reconfiguration of the supernova feedback parameters, γ_{SN}, and $V_{\mathrm{SN,\ burst}}$ to match the early-type metallicity. This reconfiguration provides better fits to the galaxy sizes, while degrading the fit to the HI mass function, which is also very sensitive to the choice of γ_{SN}. The fits found when we choose not to include the early-type metallicity constraint are very similar to those found in Lacey et al. (2016); Baugh et al. (2019), with over-predictions for the sizes of faint early-type galaxies, good fits to the HI mass function, and an under-prediction of the metallicity of faint early-type galaxies. Including the early-type metallicity constraint, however, moves us to a different region of parameter space for this updated version of the `GALFORM` code.

Another key shift is in the preferred value of f_{stab}; the preference for lower values of f_{stab} leads to a suppression of the early-type fraction at intermediate luminosities. At these luminosities, disk instabilities are the main channel for building up spheroid components and decreasing f_{stab} limits the number of disk instabilities (see **?**). Although f_{stab} does not appear in the early-type metallicity sensitivity analysis (as shown in Fig. 3.4), this is because the sensitivity indices are dominated by the strong effects of the supernova feedback parameters. A lower f_{stab} does increase the early-type metallicity but to a far lesser extent than the supernova feedback parameters, and so gives a more exact match to the observational data.

In our analysis so far, we are perhaps making the mistake of attempting to understand a non-linear model in terms of just first order, one-at-a-time changes to the parameters. Indeed, this is one of the key weaknesses of traditional 'chi-by-eye' parameter fitting. However, as shown in Fig. 3.4, we can justify this mode of investigation; the majority of the variance due to a given parameter is generally due to just its first-order effect. ν_{SF}, α_{ret}, α_{cool} and f_{SMBH} only have weak higher order variance contributions. In the cases where this assumption is less valid, for example in the case of the parameter γ_{SN} and $V_{\mathrm{SN,disk}}$, this can be understood straightforwardly with reference to Eqn. 2.2; these parameters directly interact in the implementation of supernova feedback. It is striking how much of the variance is due to the parameters' first order effects. The outlier is f_{stab}, which has strong higher-order interactions and is not directly coupled to the other parameters in any equation.

3.3.3.1 Best-fitting model

We can now re-calibrate the `GALFORM` model across all constraints to produce an estimate of the best-fitting parameters. As we have seen, there is no single choice of parameters which can reproduce all of the constraints, and we have to decide during the calibration which datasets we would like to give more or less weighting. The ideal of automatically calibrating a semi-analytic model is therefore a difficult one to realise; we will always have to make trade-offs in how we fit to the various datasets. As described in § 3.2.3, we can do this in a semi-automatic way using the heuristic weighting scheme.

We have seen that there are a number of trade-offs or tensions to consider when aiming to find a best-fitting model. Fitting to the late-type galaxy sizes, the Tully-Fisher relation, or the HI mass function generally degrades the fit to the K- and r-band LFs. We have also seen that trying to reproduce the early-type metallicities worsens the fit to the Moffett et al. (2016) data for the early-type fraction, and worsens the fit to the high-mass end of the HI mass function. On the other hand, other observational constraints are more easily fitted; since the bulge-BH mass relation is largely dependent solely on the f_{SMBH} parameter, and this has very little influence on other observables, fitting this constraint is trivial.

With these considerations in mind, we choose heuristic weights such that the r-

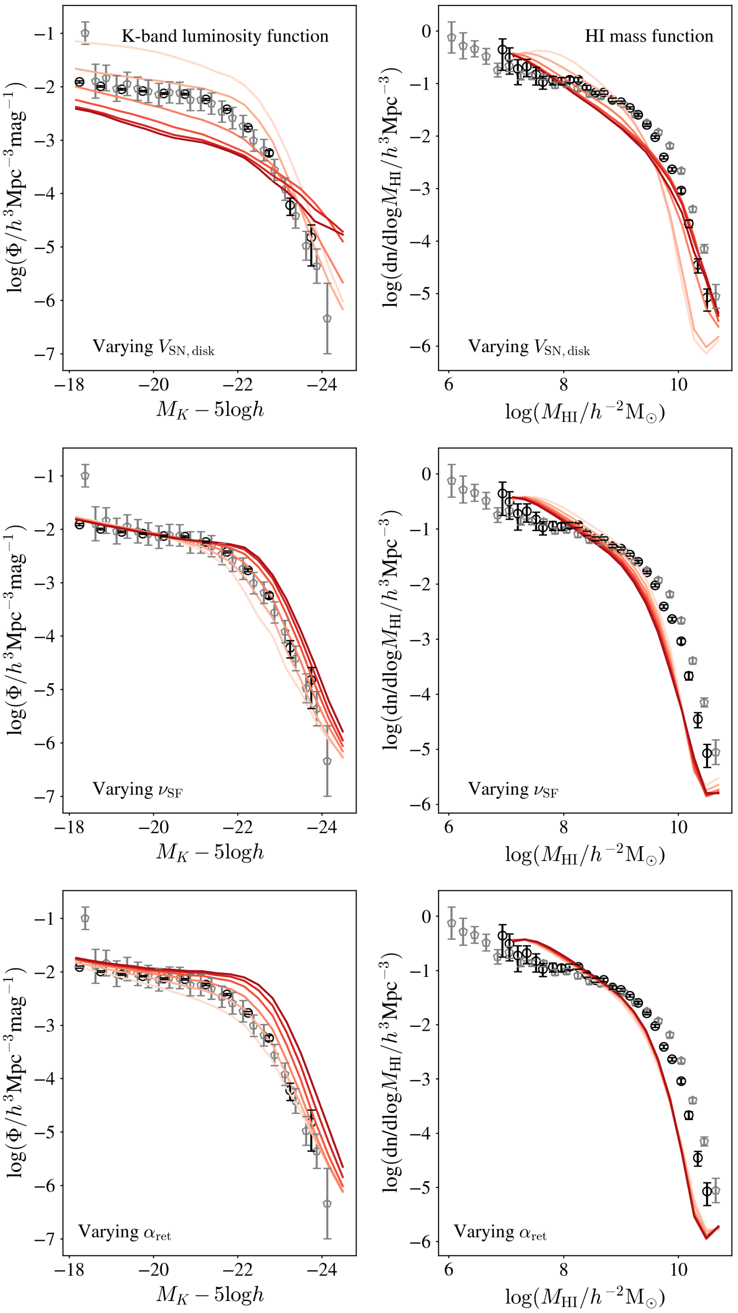


Figure 3.8: Emulator predictions for perturbing three key parameters around a fit to the K-band LF. The top row shows the result of varying the parameter $V_{\mathrm{SN,\ disk}}$ between 100 and 550 kms^{-1}, the middle row varies ν_{SF} between 0.2 and 1.7 Gyr^{-1}, and the bottom row varies α_{ret} between 0.2 and 1.2. Darker colours correspond to higher values of the varied parameter.

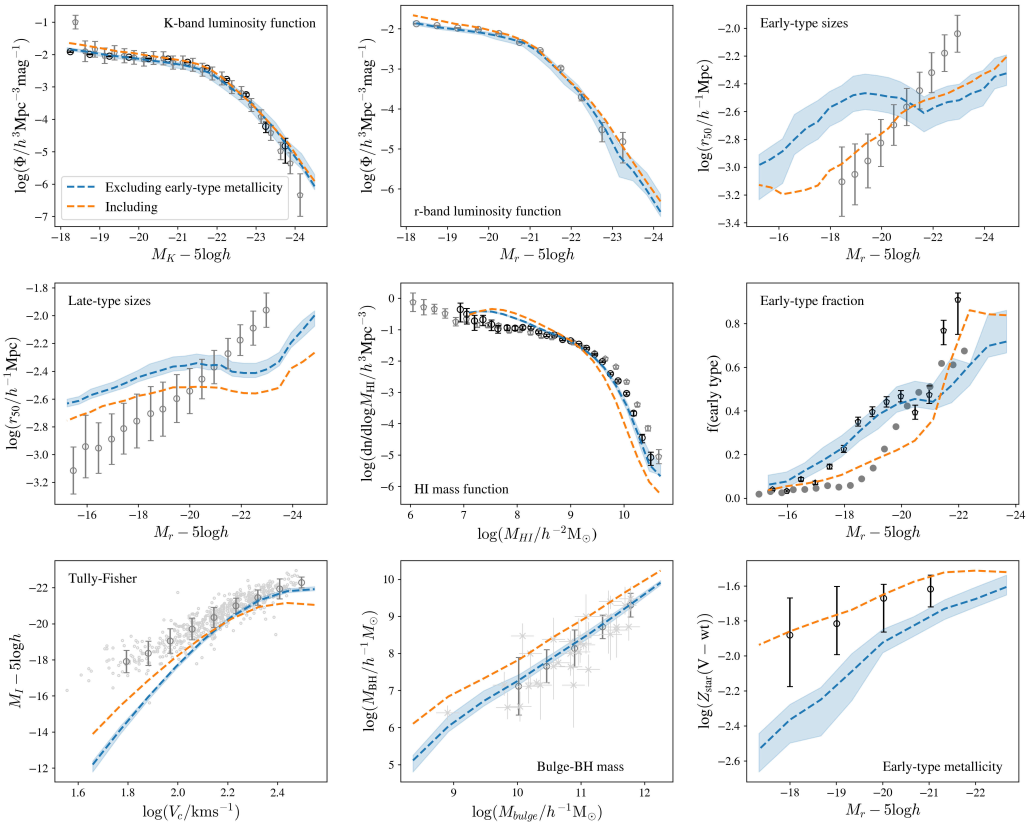


Figure 3.9: A comparison of the emulator predictions for fits to the K-band LF, the HI mass function, and the early-type fraction with and without including the early-type metallicity constraint (represented by different colour dashed lines, as labelled in the top left panel). The emulator predictions correspond to the best fit found from 20 MCMC chains, each 10,000 steps in length. In both cases, all included constraints were equally weighted. The data described in §3.2.3 is shown in black and grey. For the Bulge-BH mass relation we compare to data from Häring & Rix (2004), for the early-type fraction we fit to data from Moffett et al. (2016) and compare to data from González et al. (2009), and for the early-type metallicity we compare to data from Smith et al. (2009). Black data points indicate that the data was used for fitting, grey data points are included for comparison.

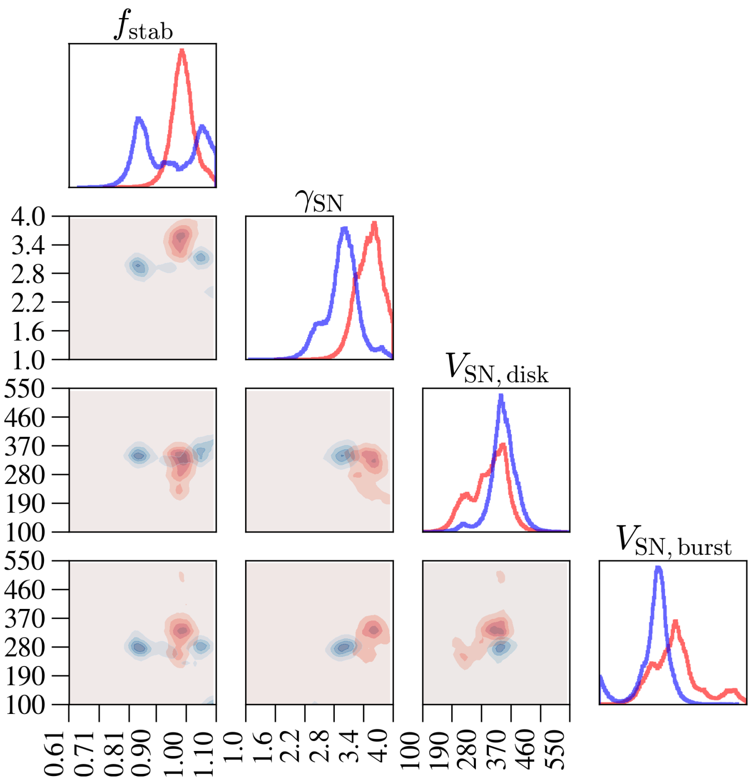


Figure 3.10: Accepted samples from 20 MCMC chains for fits to the K-band, the HI mass function, and the early-type fraction with (*blue*) and without (*red*) including the early-type metallicity constraint for a few key parameters. The shading gives a sense of the density of the samples, and the histograms show the distribution of each parameter in 1D projection. The darkest regions correspond to the 25th percentile, and the lighter regions to the 50th and 75th percentiles.

and K-band LFs are strongly weighted. We know from our previous analysis that there will be trade-offs between both the bright- and faint-ends of the luminosity functions, but we require good fits to both. Therefore we doubly weight both of these constraints when calculating the MAE given in Eqn. 3.9 (i.e. by setting $W_i = 2$ for each observable). Since the late-types sizes, early-types sizes, and the Tully-Fisher relation are important constraints, but lead to compromised LF fits, we apply single weighting to all these constraints (i.e. $W_i = 1$). We also give a single weighting to the early-type metallicity since this trades-off against the bright end of the luminosity function and the high mass end of the HI mass function. Since the HI mass function is an important constraint, but as we are aware that it generally degrades the fit to the bright end of the luminosity function, we give this constraint

Table 3.2: The best-fitting parameters (as measured by MAE, Eqn. 3.9) found by using MCMC combined with our emulator. For reference the last column gives the parameter values used by Baugh et al. (2019). The first column indicates the set of parameters with the lowest MAE, and the second column indicates the parameter ranges of the 50 best runs of the 100 MCMC chains, again selected by MAE as described in the text.

Parameter	This work	Range	Baugh19
$f_{\rm stab}$	0.79	0.73 – 1.00	0.90
$\alpha_{\rm cool}$	0.84	0.66 – 1.16	0.80
$\alpha_{\rm ret}$	0.59	0.32 – 0.86	1.00
$\gamma_{\rm SN}$	2.24	2.05 – 2.72	3.40
$V_{\rm SN,\ disk}$ [kms^{-1}]	489	368 – 541	320
$V_{\rm SN,\ burst}$ [kms^{-1}]	284	230 – 292	320
$f_{\rm burst}$	0.25	0.12 – 0.30	0.05
$f_{\rm ellip}$	0.20	0.20 – 0.39	0.30
$\nu_{\rm SF}$ [Gyr^{-1}]	0.20	0.20 – 0.33	0.74
$f_{\rm SMBH}$	0.003	0.001 – 0.004	0.005

a triple weighting. This is to ensure that more total weight is applied to the K- and r-band LFs in combination. We apply a single weighting to the early-type fraction; we have seen that this fit is in strong tension with the early-type metallicities and sizes.

We run 100 MCMC chains with our emulator, each 10,000 steps in length. We find that the minimum MAEs (as computed using the emulator) obtained with each chain lie in the range $\sim 0.15 - 0.20$; since this range is similar to the out-of-sample accuracy of the emulator, and so in principle we cannot discern which parameter sets give the best fit to the observational data with the emulator alone, we evaluate these 100 minimum MAE parameter sets with the `GALFORM` code.

The best-fits are shown in Fig. 3.11. Here we plot the best 50 sets of parameters from the 100 MCMC chains, as evaluated with the `GALFORM` code. These runs have very similar MAEs, covering the range $0.16 - 0.18$, while the runs not shown cover the range $0.18 - 0.22$, which is slightly wider than the range predicted by the emulator, but within the expected emulator error (0.04 in this weighting scheme). The solid red line indicates the run with the lowest MAE, and the blue lines show the remaining 49 runs. The shading on these lines indicates the size of the residuals between the model and the HI mass function, with darker lines indicating smaller residuals, and demonstrates that the parameter choices which provide the best fits to the HI mass function over-predict the bright-end of the LFs. The black dashed line shows the statistical galaxy properties of the model presented in Baugh et al. (2019) (hereafter Baugh19). In Table 3.2 we show the set of parameters with the lowest MAE to the observational data (corresponding to the red line in Fig 3.11), the parameter range of the best 50 parameter sets, and compare with the parameters adopted in Baugh19 for an older version of the model. We reiterate, however, that the best-fit parameters are just one realization out of many possible choices due to the degeneracies between the parameters, and the effect of calibrating to multiple datasets. Also, the ranges shown in Table 3.2 do not indicate that any choice of parameters within these ranges will yield an equivalent fit; the value of one parameter will constrain the choices for the other parameters, hence the reason for giving an example of a best-fitting set of parameters. We find that some parameters, such as $\nu_{\rm SF}$ and $\gamma_{\rm SN}$ are constrained to a tight range of values, whereas others, such as $f_{\rm stab}$ can be drawn from a large fraction of the explored range.

Calculating the mean absolute error of the best-fitting model, and the Baugh19 model, using the same procedure as described in §3.2.4 (and recalling that we scale each output so that the data lie in the range [0,1]), we find that at least under this metric the new model is a better fit to the data. Over all the datasets, the new best-fit found in this work gives an MAE of 0.16 vs. an MAE of 0.20 for the Baugh19 model. We note that the MAE for the model used in Baugh19 is within the range of the minimum MAE reached by the 100 MCMC chains. The reduced MAE of the new best fitting model compared to the Baugh19 model is mainly due to large improvements in the fits to the early-type galaxy sizes and their metallicities, while the fits of the new model to the early-type fraction and Tully-Fisher relation are slightly worse.

As shown in Fig 3.11, we find that our model provides a slightly better fit to the

K- and r-band LFs than the Baugh19 model*. For the updated model presented in this work, we find an MAE of 0.05 vs. 0.08 for the Baugh19 model in the K-band and 0.04 vs. 0.06 in the r-band. The galaxy sizes are an improvement over previous iterations of the GALFORM model, particularly the early-types, which are now more qualitatively similar to the observational data in that they are monotonically increasing with luminosity (at least in the range of the data), whereas the Baugh19 model features a marked dip at intermediate magnitudes and significant over-prediction at fainter magnitudes, differing from the observed sizes by a factor of ~ 3. The MAEs in this case are also significantly lower for the new model: for the late-type galaxies we find an MAE of 0.14 in this work vs. 0.21 for the Baugh19 model, and 0.09 vs. 0.39 for the early-type sizes. This difference is largely due to the different choices for the $\gamma_{\rm SN}$ parameter. Here, we find a preference for much lower values of $\gamma_{\rm SN}$, in the range $2.05 - 2.72$, vs. 3.40 for the Baugh19 model. Reducing this parameter significantly weakens the effect of supernova feedback in low-mass galaxies, leading to smaller sizes (see figure C10 of Lacey et al., 2016). Interestingly, the preferred $\gamma_{\rm SN}$ we recover is much closer to the value expected from energy conservation arguments, $\gamma_{\rm SN} = 2$ (Larson, 1974; Lagos et al., 2013).

The fit to the HI mass function is slightly worse than the fit found in the Baugh19 version of the model (with MAEs of 0.09 vs 0.08); a better fit would come at the expense of a more severe over-prediction of the bright-end of the luminosity function as previously discussed, and as shown by the shading of the blue lines in Fig. 3.11. As we have seen in Fig. 3.9, we are able to produce better matches to the HI mass function and the luminosity functions if we exclude the early-type metallicity and galaxy sizes constraints (the fits found in this case are much more similar to the Baugh19 model, with similarly high $\gamma_{\rm SN}$ in the range $\sim 3.2 - 3.8$, as shown in Fig. 3.10). Our fit to the early-type metallicities is an improvement over the prediction of the Baugh19 version of the model, where the MAE of our model is 0.15 vs. 0.55 for the Baugh19 model. However, our early-type metallicities fit

*Baugh et al. concentrated on reproducing the b_J-band luminosity function, and the HI mass function, and did not consider the r-band LF.

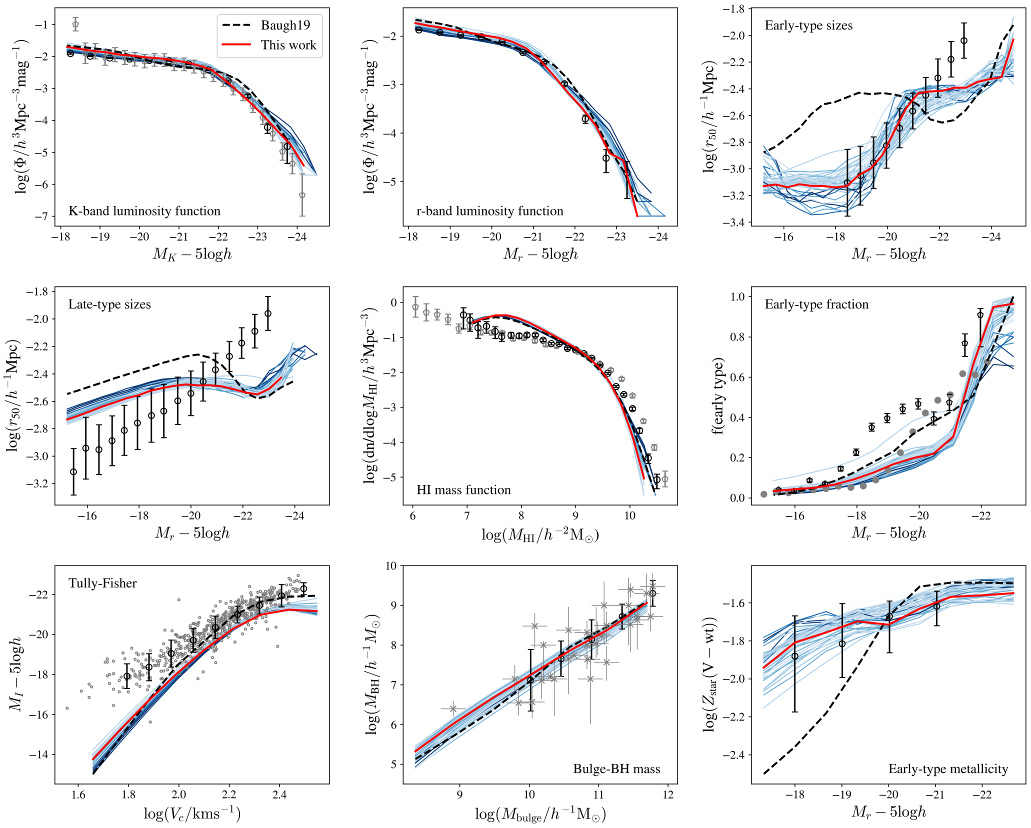


Figure 3.11: The `GALFORM` evaluations of the best-fitting parameters found with 100 MCMC chains, each 10,000 samples in length, using the constraint weightings described in the text. Here we plot a sample of the best 50 runs, as measured by MAE. The red line indicates the parameter set with the lowest MAE. The remaining 49 runs are plotted in blue, with darker shades indicating small residuals to the HI mass function. Note therefore that runs with the smallest residuals to the HI mass function over-predict the bright-end of the K- and r-band LFs. The black dashed line shows the Baugh19 model. The data described in §3.2.3 is shown in black and grey. We calibrate to the data shown in black.

comes at the cost of slightly degrading the fit to the early-type fraction (0.13 vs. 0.10). Our fit to the Tully-Fisher relation is worse than in the Baugh19 model, with an MAE of 0.28 vs. 0.17, though we have demonstrated that we can retrieve a fit more similar to Baugh19 by giving less weight to the early-type metallicity constraint (again as shown in Fig. 3.9).

3.3.3.2 Predictions for cosmic star formation history

We have calibrated GALFORM to low-redshift constraints and now investigate the predictions for the evolution of the star formation rate density (SFRD) with redshift. To do this, we evaluate the SFRD with redshift for the sets of parameters corresponding to the GALFORM runs shown in Fig. 3.11. Fig. 3.12 shows the SFRD predictions for these parameter choices. Since GALFORM assumes a mildly top-heavy initial mass function (IMF) for stars formed in starbursts, we apply an approximate correction to give the SFR which would be inferred assuming a Kennicutt IMF (Kennicutt, 1983) by weighting the starburst SFR by a factor of 1.9 (as in Lacey et al., 2016). The curves therefore represent an apparent SFRD which can be compared with observational estimates which assume a solar neighbourhood IMF. Interestingly, we see that the spread of the model predictions only increases slightly as we move out to larger redshifts. This suggests that the low-redshift calibration datasets actually constrain the redshift evolution of the model reasonably well.

3.4 Discussion

We have presented a method for efficiently calibrating and exploring a SAM of galaxy formation across a wide range of outputs. In doing so, we have uncovered a number of tensions between datasets: for example, in Fig. 3.9, we found that on relaxing the requirement for a good fit to the early-type metallicities, we recovered a fit very similar to those found in Baugh et al. (2019) and Lacey et al. (2016). By increasing the weight given to the early-type metallicity constraint, we moved to a new region of parameter space, changing our fit to the early-type fraction and early-type sizes. Tensions such as this point to either deficiencies in the model, or a discrepancy between the observational datasets. For example, again in Fig. 3.9, we see that the early-type fraction fit to the Moffett et al. (2016) data (shown in black) degrades when we include the early-type metallicity constraint. However, in this case the fit is then in better agreement with the González et al. (2009) data

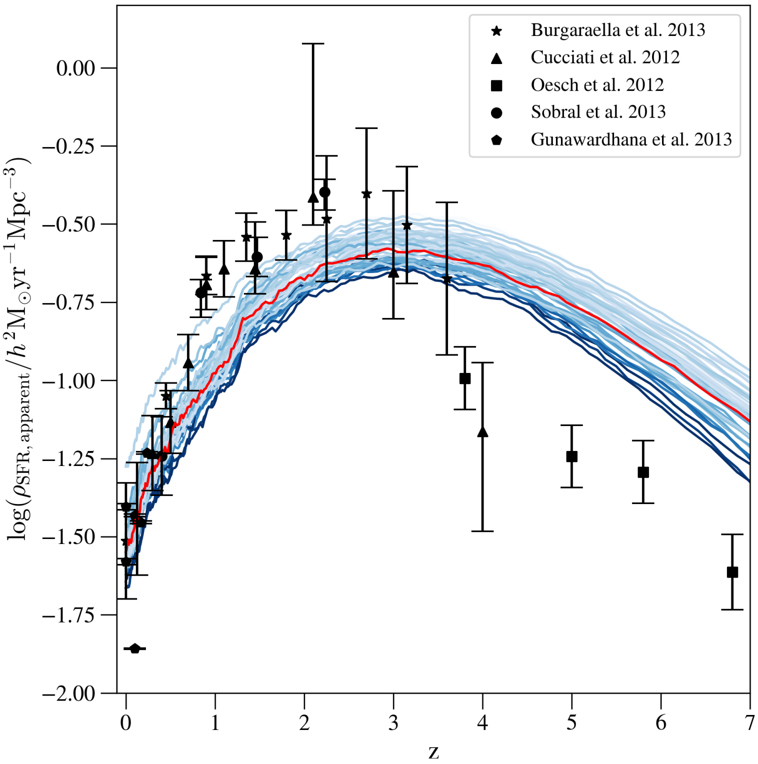


Figure 3.12: The apparent SFRD predictions for the GALFORM model evaluations shown in Fig. 3.11. The red line indicates the predictions for the best-fit parameters (as calculated by MAE), while the blue lines indicate the remaining 49 runs. These lines are shaded according to the model's residuals to the HI mass function, with darker shades indicating smaller residuals. We compare to observational data from Burgarella et al. (2013); Cucciati et al. (2012); Oesch et al. (2013); Sobral et al. (2013); Gunawardhana et al. (2013). Note that these data were not used in the fitting. A correction has been applied to the predicted SFRD in bursts to give an apparent SFRD, as described in the text.

(shown in grey). Similarly, for the HI mass function, the Zwaan et al. (2005) and Martin et al. (2010) datasets do not agree with one another, differing by up to a factor of five in abundance at high masses.

In other cases, we can see a clearer deficiency in the GALFORM predictions. For example, in Fig. 3.5 we show the effect of fitting to the K-band LF or the late-type galaxy sizes, or both together. We see that even when we fit only to the late-type sizes constraint, we are still not able to recover the observed monotonic increase in radius with increasing luminosity. Clearly, this suggests that the treatment of the galaxy disk-sizes in GALFORM needs to be improved.

The emulation method presented here contrasts with previous work; most emulators have focused on reducing the parameter space by using more approximate emulators, but with robust uncertainty measures, to iteratively reduce the volume of parameter space which could plausibly produce good fits to the data. van der Velden et al. (2021), for example, used a total of 3000 runs over three waves to calibrate the `MERAXES` SAM to the stellar mass function. We have focused instead on maximizing the accuracy of our emulator of `GALFORM` across the whole parameter space. Our aim is to build an emulator which allows us to explore a wide range of calibration datasets, and different combinations of these datasets. As shown in Fig. 3.2, our emulator performs well: most of the key constraints are tightly predicted.

In this vein, we have discounted the observational error bars to facilitate model exploration. In § 3.3.3.1 we calibrated our model to the full set of observational datasets under consideration. However, since we did not include observational errors and used an absolute error metric, it is difficult to give meaningful error bars around our estimates of the best-fitting parameters quoted in Table 3.2. As previously mentioned, SAM calibration involves making trade-offs between certain observational constraints; often the best-fitting model is calibrated in a way which is poorly defined. We have attempted to reproduce and elucidate this process in an automatic way through a heuristic weighting scheme. We aim to investigate a more robust calibration analysis in the future with an improved treatment of the observational errors.

Similarly, our approach could be extended to include a more robust measure of the emulator's uncertainty in reproducing `GALFORM` outputs. When emulating a set of model outputs, we should ideally account for *epistemic* and *aleatoric* uncertainty. *Epistemic* uncertainty refers to the uncertainty associated with the emulator's parameters (in this case, the weights of the neural network), and *aleatoric* uncertainty refers to uncertainty inherent in the data generating process (for example, the sampling noise on the `GALFORM` outputs). Our approach does not currently model the epistemic uncertainty on the emulator's weights, but instead acts to reduce it by averaging over a number of individual estimates provided by the neural networks in our ensemble. It is possible therefore that we are discarding regions of the parameter space which potentially contain reasonable fits to the data. However, we are somewhat protected from this scenario in that the regions which are most difficult for our emulator to model are regions which produce 'unusual' or 'undesirable' outputs (e.g. such as LFs without a clear exponential break), which are unlikely to be good matches to the observations anyway. Nevertheless, ideally we would like our emulator to return an estimate of its uncertainty (both the uncertainty in the emulator's weights and uncertainty inherent in the data-generating process). `GALFORM` is a deterministic code, but we are still limited by the noise associated with sampling from a relatively small population of galaxies at high masses. Bayesian neural networks (Neal, 1994; Bishop, 1997) are a class of models which seek to incorporate epistemic and aleatoric uncertainty into the deep learning framework; these networks often apply independent Gaussian prior distributions over model weights, and model the outputs themselves as distributions. We believe this may be a promising line of inquiry to combine the power of the neural network's adaptive basis functions with the uncertainty quantification of a full Bayesian analysis.

Another appealing method is the deep kernel learning approach (Wilson et al., 2016). Here, a deep neural network is employed to transform the inputs to the kernel of a Gaussian process regression, and it has been shown to outperform both the plain Gaussian process model and the plain deep neural network in a number of cases (e.g. Wilson et al., 2016; Patacchiola et al., 2020) while also providing robust uncertainty estimates. Here, the deep neural network can be thought of as a feature extractor which reduces the number of features input into the Gaussian process kernel and so allowing it to better generalize to higher dimensional inputs.

In Fig. 3.3, we demonstrated that we could improve the performance of our emulator as much as 10% by averaging over 10 neural networks, rather than using just one. It may be interesting to investigate this avenue further. Our method used a simple average, but if a selection of machine learning algorithms are able to give errors which are not strongly correlated (i.e. some fit better to certain examples than others), it may be possible to use a more sophisticated approach to incorporate the respective advantages of a number of different algorithms (see e.g. Maclin & Opitz, 2011).

We have proposed a number of ways to investigate the GALFORM model with our emulator. We can use sensitivity analysis techniques to evaluate the effect of different parameters, and since the emulator is extremely fast, we can manually explore the outputs in detail. It may also be possible to use symbolic regression such as the proprietary software EUREKA (as described in Dubčáková, 2011) or sparse regression-based methods (e.g. Rudy et al. (2019)) to generate closed-form expressions of the neural network outputs if desired (i.e. an estimate of the functional form of the outputs). Cranmer et al. (2019), for example, applied symbolic regression techniques in conjunction with graph neural networks to extract equations from cosmological simulations.

3.5 Conclusions

We have implemented a deep learning approach to emulate the GALFORM SAM. We trained an ensemble of deep learning algorithms to approximate the full model using just 930 evaluations of GALFORM. We used this to explore the parameter space of GALFORM, and to calibrate the model parameters to a wide array of observations. Typically the exploration of a model parameter space and the determination of a best-fitting set of parameter requires many more than 930 explicit full calculations. Our emulator is remarkably accurate, particularly in regions of the parameter space for which the model gives outputs which are close to matching the observed Universe.

We used sensitivity analysis to quantify the influence of different parameters on the model outputs, to better understand which parameters are of greatest importance in fitting to different observations (see Oleśkiewicz & Baugh 2020). Here, as shown in Fig. 3.4, we found that the majority of the variance is due to just a few key parameters, which leads to tension when trying to calibrate to multiple observational datasets.

We explored the tensions between the use of different observational datasets further, using MCMC to fit the emulator output to observational data with a heuristic weighting scheme. This allowed us to reproduce the known tension between the faint-end galaxy LFs in the K- and r-bands and the late-type galaxy sizes, and to uncover a number of others. Furthermore, we used the same technique to find a global fit to the observational datasets, finding a set of parameters which provide an improved fit to the early-type galaxy sizes and metallicities as compared with an earlier version of the GALFORM code presented in Baugh et al. (2019).

We intend to apply our emulation approach to calibrate GALFORM using the observed galaxy redshift distribution to generate mock galaxy catalogues for the DESI bright galaxy survey (Aghamousa et al., 2016). This requires model outputs over a large number of redshifts, which makes running GALFORM more computationally expensive. We are motivated therefore to reduce the required number of model evaluations as much as possible; calibrating the model across this redshift range would be prohibitively expensive for direct MCMC methods, and very difficult to achieve by-eye. Our emulator is ideally suited to this task; we have demonstrated that we require very few runs to achieve good accuracy, and that we are able to emulate over a wide range of outputs.

We believe our approach to be an inexpensive, intuitive and accurate alternative to other emulation techniques in the literature, and that this method will serve as an invaluable tool in quickly exploring and calibrating SAMs, and for the rapid assessment of the implications of changes to the underlying model.

Chapter **4**

Calibrating GALFORM to SMG constraints

Summary: The nature of galaxies detected by their emission at sub-millimetre wavelengths (SMGs) remains controversial, with conflicting claims made about how these galaxies fit into hierarchical structure formation. We revisit this question using Bayesian optimization to perform an exhaustive search of the parameter space of the `GALFORM` semi-analytical galaxy formation model. This model has been used to argue that a top-heavy stellar initial mass function (IMF) is needed in bursts of star formation to reproduce the number counts and redshift distribution of SMGs, whilst also matching the observed number of bright galaxies locally. Our new approach to finding the best-fitting model parameters converges to a solution with as few as 200 full model evaluations, even when varying 15 parameters. We test the ability of the model to match, simultaneously, the observed $z = 0$ K-band luminosity function, and the SMG number counts and redshift distribution, both with and without a top-heavy IMF in bursts of star formation. Although the model can match the sub-millimeter counts and redshift distribution at some level when assuming a solar neighbourhood IMF in all star formation, it is not possible for this variant to also match the local K-band luminosity function. This model also requires much higher rates of quiescent star formation than is usually predicted as

well as requiring disk instabilities to be switched off altogether. We find that a variant with a top-heavy IMF in bursts of star formation is able to simultaneously match the low-redshift K-band LF as well as the SMG constraints.

4.1 Introduction

Observations of galaxies in the high-redshift Universe can impose strong constraints on galaxy formation models, particularly when used to test the models alongside local observations. Sub-millimeter galaxies (SMGs), first discovered using the SCUBA[*] instrument on the James Clerk Maxwell Telescope (Smail et al., 1997; Hughes et al., 1998) are a population of galaxies undergoing dust-obscured star formation (for a review, see Casey et al., 2014). As the sub-mm emission from these galaxies undergoes a negative k-correction, for a fixed total infrared luminosity, their flux is almost unchanged over a wide range of redshifts, providing a useful window through which to study galaxy evolution. The bright SMGs are estimated to contribute up to half the star formation rate density at $z \sim$ 2-3, while having a space density of $\sim 10^{-5}$cMpc^{-3} (e.g Chapman et al., 2005; Smith et al., 2017), have large stellar masses of $\sim 10^{11} M_{\odot}$ (e.g Swinbank et al., 2004; Da Cunha et al., 2015), and inhabit massive halos $\sim 10^{13} M_{\odot}$ (e.g. Blain et al., 2004). If the SMG emission is assumed to be powered by star formation with a solar neighbourhood IMF, the resulting star formation rates are intense and the episode of star formation corresponding to the SMG phase could be responsible for a sizeable fraction of the mass of a present day, bright elliptical galaxy.

Any viable galaxy formation model should aim to reproduce both the number counts and the redshift distribution of SMGs, at the same time as reproducing observations of local galaxies, but so far this has proven challenging. Early semi-analytic models under-predicted the number counts of SMGs by over an order of magnitude (Granato et al., 2000; Somerville et al., 2012). Baugh et al. (2005)

[*] Submillimetre Common User Bolometer Array

argued that adopting a stellar initial mass function (IMF) in bursts of star formation with a larger proportion of massive stars than in a solar neighbourhood IMF (i.e. $dn/d\log m \propto m^{-x}$ with $x = 0$ adopted) allowed the model to match both the number counts and redshift distribution of SMGs, without compromising the model predictions at low redshift. Later iterations of this model (Lacey et al., 2016; Baugh et al., 2019; Cowley et al., 2019), which include feedback by AGN heating to modulate the abundance of bright galaxies, are able to reproduce the properties of SMGs and local galaxies with a somewhat less extreme but still top heavy IMF in bursts, with $x \sim 1$ (note that a Chabrier (2003) IMF, has the form $dn/d\log m \propto m^{-1.35}$ above one solar mass).

Some semi-analytic models have claimed success in matching the observed SMG redshift distribution and number counts. Safarzadeh et al. (2017), using the semi-analytic model of Lu et al. (2014), find a reasonable match to the number-counts and redshift distribution of SMGs (albeit predicting too low a median redshift compared to more recent observations). However, this work is based on using a fitting formula to relate the $850\,\mu m$ flux of a given galaxy to its dust mass and SFR; the formula was found by Hayward et al. (2011) from a small number of non-cosmological hydrodynamic simulations and does not *self-consistently* calculate the dust temperature and emission as in Baugh et al. (2005) or Lacey et al. (2016). The SHARK semi-analytic model (Lagos et al., 2018, 2019) produces a reasonable match to the faint-end of the SMG number-counts but over-predicts of the bright end by a factor of ~ 5. Again, the median redshift of SMGs predicted by `SHARK` is lower than the most up-to-date estimates (Dudzevičiūtė et al., 2020).

Other models have also made predictions for SMGs. Predictions from the EAGLE hydrodynamic simulation (Schaye et al., 2015), in which a Chabrier (2003) IMF is assumed for all star formation, were presented in McAlpine et al. (2019); these authors found some agreement with the redshift distribution of these galaxies but greatly under-predicted the number counts, with the simulation producing almost no sources above 5mJy. The SIMBA hydrodynamic simulation (Davé et al., 2019; Lovell et al., 2021), again assuming a Chabrier IMF, similarly finds agreement with the redshift distribution of the submillimeter galaxies but under-predicts the SMG counts by a factor of 3-10 at the bright end, depending on the assumptions about blending, and by greater than an order of magnitude at the faint end. Hayward et al. (2021) compares the sub-millimeter predictions for the Illustris and IllustrisTNG simulations (Nelson et al., 2015; Pillepich et al., 2018).Illustris reproduces the number counts reasonably well, but predicts a significantly lower median redshift for bright sources than is observed. IllustrisTNG produces a redshift distribution that is in better agreement with observations, but under-predicts the number counts by greater than an order of magnitude at bright fluxes. We note that these predictions are also not based upon a self-consistent calculation of the dust temperature, but instead again use the fitting-formula for sub-millimeter flux from Hayward et al. (2011).

Here, we revisit the need for a top-heavy IMF in hierarchical galaxy formation models. Allowing the form of the IMF to change depending on the mode of star formation is seen as controversial (see, for example, Bastian et al. 2010). Nevertheless other authors have considered such a possibility. Using the strength of emission lines and the Balmer decrement measured from star-forming galaxies in the Galaxy and Mass Assembly Survey, Gunawardhana et al. (2011) argued that the slope of the IMF varies with the star formation rate (SFR), with the IMF becoming more top-heavy as the SFR increases, reaching $x \approx 0.9$ in the most intensely star-forming galaxies, similar to the value adopted by Lacey et al. (2016). (Note the solar neighbourhood Chabrier IMF has a slope of $x = 1.35$ above $1 M_{\odot}$.) Romano et al. (2017) inferred a top-heavy IMF slope, with $x \approx 0.95$ in nearby starburst galaxies. Schneider et al. (2018), studying massive stars in the Large Magellanic Cloud, found an IMF of $x = 0.9 \pm 0.3$, and subsequent analysis by Farr & Mandel (2018) found an IMF of $x = 1.05 \pm 0.14$, both top-heavy compared to the solar neighbourhood IMF. Nevertheless, though many studies provide evidence for variations in the IMF, the exact nature of the variation is much more uncertain. Some studies (for example, Conroy & van Dokkum 2012; Smith 2020) infer evidence for a bottom-heavy IMF in high mass galaxies, and Weidner et al. (2013) propose a time-dependent IMF that favours massive stars at early times and low mass stars at late times, compared with a solar neighbourhood IMF.

All galaxy formation models, whether they are hydrodynamic gas simulations or semi-analytic codes, involve a large number of parameters which govern the strength of various sub-grid processes, such as feedback from supernovae and active galactic nuclei (AGN) (Baugh, 2006; Benson, 2010; Crain et al., 2015; Somerville & Davé, 2015). These parameters must be adjusted to match some sub-set of observations before the model can be used to make predictions for other observables. One criticism of the inclusion of a top-heavy IMF in the GALFORM model is that, given the relatively large parameter space, another, unexplored choice of parameters might be able to fit the observations without requiring this assumption. Our aim is to find out if there is a set of parameters with which the model, assuming a universal solar neighbourhood IMF, is capable of matching the observational constraints from SMGs while maintaining reasonable fits to low-redshift datasets such as the K-band luminosity function. As a secondary aim, we investigate the slope of the IMF in bursts, when the IMF is allowed to change depending on the mode of star formation. Will a more extensive search of the model parameter space reveal the possibility of invoking a less ‘extreme’ IMF in bursts?

To achieve this, we draw on techniques from the field of Bayesian optimization (see e.g. Frazier 2018). This class of methods is used to calibrate the values of the parameters in a model, also called model optimisation, by searching for the best-fitting set of parameters, as judged by how well the model reproduces a set of target datasets. The metric to determine the best-fitting model could be based on a measure of the distance between the model predictions and the calibration data. In general, we do not know how the metric depends on the model parameters, which makes this a so-called ‘black-box’ problem. Bayesian optimisation is typically used for models that are computationally expensive to run, and we need an optmization method that does not require a large number of model evaluations. In Bayesian optimisation, the metric is known at every point in the parameter space, with varing degrees of certainty. Here we use Gaussian processes (GP; see e.g. Rasmussen & Williams 2006) to describe the metric. As more model evaluations are performed during the optimization, the GP are updated. Once there is no further significant improvement in the metric judging how well the model reproduces the calibration data, the best-fitting model is the one with the lowest value of the distance metric. We find that this method is able to find suitable fits to the data in a fraction of the number of model evaluations required by other (often more elaborate) approaches (see e.g. Kampakoglou et al., 2008; Henriques et al., 2009; Bower et al., 2010; Vernon et al., 2010; Benson & Bower, 2010; Lu et al., 2011, 2012; Ruiz et al., 2015; Martindale et al., 2017). However, with this approach we only get a limited sense of the uncertainty on the parameters in exchange for an order of magnitude fewer evaluations of the full model. Bayesian optimisation has been used in other problems in astronomy that involve running a computationally expensive model over a parameter space with a high number of dimensions (e.g. interpreting supernovae light curves Leclercq 2018; the Lyman-α forest power spectrum Rogers et al. 2019; Takhtaganov et al. 2021; looking for signatures of inflation in cosmic microwave background data Hamann & Wons 2022)

The calibration of semi-analytical models and the exploration of their parameter spaces has been investigated in a number of recent papers. In general, two approaches have been taken to tackle calibration: direct evaluation of the model parameter space across a large number of parameter choices, and emulation, in which a much smaller number of full model runs are used with the bulk of the parameter sampling being done by a surrogate or proxy which uses greatly reduced computational resources.

The direct exploration approach has been investigated in a number of papers. Both Kampakoglou et al. (2008) and Henriques et al. (2009) implemented Markov-Chain Monte Carlo (MCMC) techniques to calibrate their semi-analytical models to a range of datasets, and found that the choice of dataset effected the choice of best-fitting parameters. Martindale et al. (2017) used the same semi-analytical model as Henriques et al. but also included the HI mass function as a constraint, again changing the best-fitting parameters. Similarly, Lu et al. (2011, 2012, 2014) applied this approach to fit parameters to the K-band luminosity function and the HI mass function. Ruiz et al. (2015) used particle swarm optimization to calibrate a SAM to the K-band LF.

The emulation method instead involves building a statistical approximation to the full galaxy formation model. Bower et al. (2010) and Vernon et al. (2010) employed a Bayesian emulation technique – Bayes linear – (as developed by Goldstein & Wooff, 2007) fit the GALFORM model to the K- and b_J-band LFs, and included further observational datasets to further constrain this reduced parameter space in Benson & Bower (2010). Rodrigues et al. (2017) also applied this method to calibrate GALFORM to the local galaxy stellar mass function. Van der Velden et al. (2021) recently used the Bayes linear methodology to calibrate Meraxes galaxy formation model at high redshift.

The Bayes linear approach used by Bower et al. (2010) makes more approximations and assumptions about the functions being minimized than we do here, and so is not strictly a black-box method like the Bayesian optimisation used here. The Bower et al. method involved searching the parameter space in waves, with the space redefined at each wave to make the search adaptive. Bower et al. used 5500 runs of the GALFORM model in their search of a similar szied parameter space to the one considered here; we will use fewer than 200 runs of the full model.

We presented a new framework for the automated calibration of the GALFORM model using local observational data as used in Chapter 3. The first step was to run a sensitivity analysis to determine which model parameters were mainly responsible for shaping the model predictions for the calibration data (see Oleśkiewicz & Baugh 2020 for the first application of sensitivity analysis to the GALFORM model). This process allowed us to identify a subset of ten model parameters that were the most important for determining the form of the calibration data. We next ran the full GALFORM model for 1000 parameter combinations sampled from this ten dimensional space using a Latin hypercube (Stein, 1987). These runs allowed us to build an emulator of GALFORM using an artificial neural network. The emulator was then used to make an extensive Monte-Carlo Markov Chain search of the parameter space, returning a best-fitting parameter set for the chosen calibration data, along with an indication of the range of acceptable model predictions. The speed of this approach, which made possible the extensive exploration of a parameter space with a large number of dimensions, allowed us to explore the best-fitting models that resulted from different combinations of calibration datasets. Differences in the resulting best-fit models point to possible deficiencies in the model or in the compatibility of the observational measurements with each other.

Here we have a more challenging problem to address. First, the parameter space we search is even larger than in Chapter 3, with 15 dimensions rather than 10. The focus of the parameter space search is now: “Can we find *any* example of a model that works, under the assumptions?”, rather than simply finding a best-fitting parameter set along with the associated range of acceptable models. Second, the computational overhead for the model runs associated with each parameter set are much higher. We need predictions for the number counts and redshift distribution of galaxies, which require, because of the way the GALFORM code is set up, the model to be run for many redshifts rather than simply just for $z = 0$. This is because the GALFORM model tracks galaxy SEDs through filters, rather than tracking the full SED. Since the definition of observer frame filter changes with redshift, the model must be run separately for each output redshift. Moreover, some of the predictions are sensitive to rare events such as starbursts so we need to simulate many more examples of dark matter halo merger histories to get robust predictions.

Hence, to overcome these challenges we investigate the application a of new approach to model calibration using Bayesian optimisation. Unlike previous work, which has generally used the emulator approach (i.e. building a surrogate model which is a statistical approximation of the full galaxy formation model, with a much smaller computational overhead), or expensive MCMC routines to infer posterior parameter distributions, here we are primarily interested in whether there exists parameter choices which are capable of matching the datasets under consideration: the constraints being the number counts and redshift distribution of SMGs and the low-redshift K-band LF. Bayesian optimisation is a global optimization technique which efficiently searches the parameter space using a Gaussian process prior, and is capable of searching high-dimensional parameter spaces efficiently for global minima. Here, we aim to demonstrate that these methods are applicable to galaxy formation models in general, and then use the method to test whether there exists a set of parameters which is capable of matching the SMG observations and low-redshift constraints simultaneously without including a top-heavy IMF. Past explorations of the parameter space, though usually performed manually, have suggested that it is not possible to simultaneously match these three constraints when assuming a universal solar neighbourhood IMF. Our contribution is to validate or invalidate this conclusion with a more sophisticated parameter search, over a comprehensive list of relevant parameters and a wide search space. Such an approach represents an enormous improvement over the old-fashioned, one-at-time parameter searching originally used to argue for a top-heavy IMF.

This Chapter is set out as follows: In § 4.2, we review the theory and practical considerations behind Bayesian optimisation, listing the observational datasets selected for calibration in § 4.2.5 and, importantly, validating our method on a surrogate model in § 4.2.6. In § 4.3, we present the results of the model calibration, for different assumptions about the form of the IMF and about the importance placed on reproducing various datasets. We give a summary in § 4.4 and conclusions in § 4.5. The processes and parameters of the `GALFORM` model have been reviewed in § 2.1. The list of model parameters varied, and the range considered for each parameter is given in Table 4.1.

Table 4.1: The parameters explored in this work and the range of values over which they are varied.

Name	Process	Range
$F_{\rm stab}$	Disk instability	0.5 - 1.2
$\gamma_{\rm SN}$	SN feedback	1.0 - 4.0
$\alpha_{\rm cool}$	AGN feedback	0.2 - 4.0
$\alpha_{\rm reheat}$	SN feedback	0.2 - 3.0
$V_{\rm SN,\ disk}$ $({\rm km\,s^{-1}})$	SN feedback	10 - 800
$V_{\rm SN,\ burst}$ $({\rm km\,s^{-1}})$	SN feedback	10 - 800
$f_{\rm ellip}$	mergers	0.2 - 0.5
$f_{\rm burst}$	mergers	0.01 - 0.3
$\nu_{\rm SF}$	Quiescent star formation	0.1 - 4.0
$f_{\rm SMBH}$	BH growth	0.001 - 0.05
$\tau_{*\rm burst,\ min}$ (Gyr)	Burst star formation	0.01 - 0.05
$f_{\rm cloud}$	Dust	0.2 - 0.8
$t_{\rm esc}$ (Gyr)	Dust	0.0001 - 0.01
$\beta_{\rm burst}$	Dust	1.5 - 1.2
x	Initial mass function	0 - 1.35

4.2 Bayesian optimization

We use Bayesian optimisation (see Frazier 2018 for a review) to set the model parameters so that `GALFORM` reproduces as closely as possible the observational datasets used for model calibration. Here we explain why we take this approach and set out its background. We start in § 4.2.1 with an overview of the problem, the exploration of a high dimensional parameter space of a computationally expensive model, and explain how Bayesian optimisation addresses this challenge. The nature and role of Gaussian processes, the tool used to convey our knowledge of the parameter space is described in § 4.2.2. The kernel function is an important part of the Gaussian process description and is explained in § 4.2.3. The method used to choose where to add new calculations with the full model to the parameter space is presented in § 4.2.4. The datasets used to calibrate `GALFORM` are listed in § 4.2.5. A simple validation of our approach is given in § 4.2.6. We close this section in § 4.2.7 by describing the practical application of Bayesian optimization to `GALFORM`.

4.2.1 An overview of Bayesian Optimization

In a traditional parameter space exploration, a model is evaluated at a series of points in the parameter space and each parameter set is ranked by a metric, f, which measures the discrepancy or distance between the model predictions and the calibration data. In low dimension parameter spaces, a full grid search may be feasible. Often, however, there are too many parameters for such a search to be possible and a more efficient method is used, such as a Monte Carlo Markov Chain (Robert, 2015). New positions or samples of the parameter space are accepted by the chain if certain conditions are met, such as the proposed point having a smaller value of f than the current point.

In our application, the `GALFORM` code is expensive to run many times. As we are searching a high dimension (15) parameter space, a prohibitively large number of model evaluations would be required to calibrate the model using traditional methods. One option would be to bypass evaluating the model for every new set of parameters by producing an emulator to mimic the effect of running the model, which we discuss in § 4.2.6. This was the approach we took in Chapter 3, where we used several hundred runs of the full model to train an artificial neural network to produce output that was, for most parameter sets, close to that obtained by running the full model. This option is not feasible in the present application, as each set of model parameters requires many redshifts to be run, rather than just one, as we are comparing the model predictions to galaxy number counts and redshift distributions; hence, the computational costs of each model evaluation is higher than in Chapter 3. Also, the dimensionality of the parameter space is higher in this Chapter than in Chapter 3, implying even more sets of parameters would have to be run to train an emulator (note that the emulator we use to validate our method in § 4.2.6 is too approximate to be used to calibrate the model accurately).

Bayesian optimization replaces the emulator with a surrogate description of the metric function, f, using a Gaussian process (GP). With a small number of evaluations of the full model (compared to other methods), the GP describes the value of f and uncertainty on this value at *each* position in the parameter space, x (where x can be multi-dimensional in practice, but is one dimensional for the illustrative plots, Figs. 4.1 and 4.2). For values of x close to a position in the parameter space at which there is a full model evaluation, the uncertainty on the predicted value of f is small. The uncertainty grows as we move away from the locations in x where the full model has been run. So we have knowledge of f at any position in the parameter space, just that in some places this knowledge is better than others. The knowledge of f can be improved by carefully choosing the next position in the parameter space to run a full calculation of the model. This process is dealt with in § 4.2.4.

So after on the order of a few tens of full model calculations, at locations in the parameter space sampled using a Latin hypercube, the search for a minimum value of f begins, and, as we show later, can be completed following a few hundred full model runs.

Since we do not have any information about the form of f, nor do we compute its derivatives (which could be used in a gradient descent method to find an extreme point), this is technically referred to as a black-box, derivative free, global optimisation problem (Frazier, 2018).

4.2.2 Gaussian processes

Gaussian process (GP) regression is a non-parametric way to calculate a posterior distribution, starting from the assumption that the underlying distribution is continuous (see e.g. Rasmussen & Williams 2006). The GP gives the value of the metric function (sometimes called the objective function) at any point in the parameter space, $f(x)$, along with an estimate of the uncertainty on this value.

To specify the GP, a mean value and a covariance matrix or kernel function are required. The mean is generally taken to be a constant; here we set this value to zero since the target value for the metric function is as close to zero as possible. The covariance matrix describes the correlation between the values of f at nearby points in the parameter space, given a definition of a characteristic length scale in the parameter space x. Note that x can be a multi-dimensional parameter space. Points which are considered close together in the parameter space, according to some measure of distance, will tend to have similar values of the objective function, f. This behaviour follows from the assumption that f is a smooth, continuous function.

The GP functions are modelled as being drawn from a multivariate normal distribution

$$f(x') \sim N(0, K(\theta, x, x')), \tag{4.1}$$

where $K(\theta, x, x')$ is a covariance matrix of all pairs of points in the parameter space (x, x') given by an assumed kernel function (see next subsection) parameterised by θ, and we have already assumed a mean value of 0. The hyperparameters θ can then be optimized by maximizing the log-likelihood

$$\log\, p(f(x')|\theta, x) = -\frac{1}{2} f(x)^T K^{-1}_{\theta,x,x'} f(x') - \frac{1}{2} \log\, \det(K_{\theta,x,x'}) - \frac{n}{2} \log\, 2\pi. \tag{4.2}$$

Note that the function f is known at both points, x and x'. If we wish to make predictions for a new location in the parameter space, x^*, the posterior mean and variance at this point are given by

$$\mu^* = K_{\theta,x^*,x} K^{-1}_{0,x,x'} f(x) \tag{4.3}$$

$$\sigma^{2*} = K_{\theta,x^*,x^*} - K_{\theta,x^*,x} K^{-1}_{\theta,x,x'} K^{T}_{\theta,x^*,x} \tag{4.4}$$

which are the standard formulae for conditioning a multivariate normal distribution.

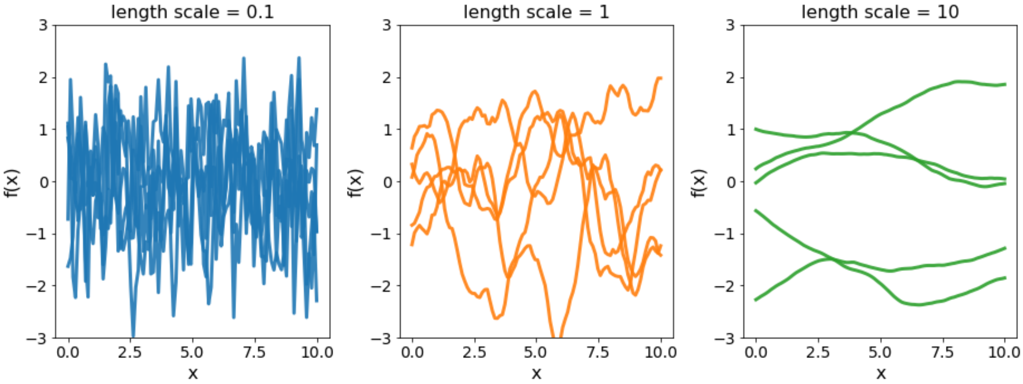


Figure 4.1: A demonstration of the effect of the length scale adopted in the kernel function on the appearance of a Gaussian process (GP) prior. Each panel shows several realisations or draws from a GP. In each case the process has zero mean. However, the hyperparameter that governs the scales over which values of f are correlated varies between panels. The left panel shows the shortest correlation length scale, with $\theta = 0.1$, the middle panel shows 1.0, and the right panel 10.0. A shorter length scale corresponds to a function which changes rapidly with small changes to the input parameters.

Fig. 4.1 shows examples of GPs with different correlation lengths between the values at closeby points in the parameter space. The correlation in this example is described by a radial basis function kernel, $K(\theta, x, x') = \exp((x - x')^2/2\theta^2)$. Each panel of Fig. 4.1 shows several examples or realisations of Gaussian processes; in this example we are simply plotting values drawn from a Gaussian distribution with mean zero and different choices for the length scale in the covariance matrix. Each curve is an example of a 'pseudo'-random walk in which there is some correlation between the steps. In each panel, different values are assumed for the hyperparameter θ which controls the typical length scale over which the values of f are correlated, moving from a small value of θ in the left panel, through to intermediate values in the middle panel and large values in the right panel. If the function that we are trying to reproduce using GPs varies slowly with x, then a large correlation length i.e. a large value for θ is needed.

4.2.3 The choice of kernel function for the Gaussian process

The choice of kernel function is important in GP regression as this sets how rapidly the objective function f can vary, and hence the ability of the GP to detect feature relevance i.e. how rapidly the objective function f changes with a given parameter value varying (Rasmussen & Williams, 2006).

Feature relevance means, in practice, that each parameter, x_i, has an associated length-scale parameter, θ_i, rather than a global value for θ for all parameters. Hence, the length-scale of our objective function, f, (i.e. as shown in Fig. 4.1) is allowed to vary across each dimension in parameter space.

The simple illustration of GPs shown in Fig. 4.1 used a radial kernel function (see subsectiion above) to encode the correlation or covariance between points in the parameter space. For the application to GALFORM we instead use the Matern kernel function, for which the covariance between two points x_i and x_j is given by

$$C_\nu(x_i, x_j) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{d(x_i, x_j)}{l} \right)^\nu B_\nu \left(\sqrt{2\nu} \frac{d(x_i, x_j)}{l} \right), \tag{4.5}$$

where B_ν is a modified Bessel function of the second kind, $d(x, x')$ is the Euclidian distance metric between points x and x' in the parameter space, and l controls the length-scale for a component of the vector describing the full parameter space and of the function. The Matern kernel has a number of attractive features, such as high level of differentiability and the ability to model functions with varying degrees of smoothness. In fact, as $\nu \to \infty$, the Matern kernel converges to the radial basis kernel. We use the Matern 5/2 kernel with automatic relevance detection, that is with $\nu = 5/2$, in which case the kernel can be written as the product of a polynomial and an exponential:

$$K(x_i, x_j) = \sigma^2 \left(1 + \sqrt{5}r + 5r^2 \right) \exp\left(-\sqrt{5}r \right), \tag{4.6}$$

where

$$r = \sqrt{\sum_{m=1}^{d} \frac{(x_{im} - x_{jm})^2}{l_m^2}}. \tag{4.7}$$

Since we have a length scale l_m for each dimension in the parameter space, the length-scale corresponds to the relevance of each input parameter. Parameters with a long length-scale will therefore contribute much less to the covariance.

4.2.4 Where to take the next step in the parameter space?

Our aim is to find a set of parameters, x^*, which minimises the distance between the observational data and the predictions of the model in terms of some metric. The output of a model evaluation is a set of predictions for various statistics describing the galaxy population, such as, as we will see in the next section, the values of the K-band luminosity function in a series of luminosity bins. We then calculate a metric or distance measure between the model predictions, $g(x)$, and the corresponding observational data y, using the L1-norm $||g(x)-y||_1$. We use the L1 norm rather than the more common L2 norm (in which the distance between model prediction and data is squared and then summed over the bins the statistic is measured in) to reduce the influence of outliers in setting the value of the metric. The GP then predicts the value of $f(x) = ||g(x) - y||_1$. We aim to find an optimal value $x^* \in X$ such that $f(x^*)$ is a minimum.

The process of selecting a new position in the parameter space at which to make an evaluation of the full model requires us to calculate an acquisition function, a_{EI}. The acquisition function balances two aims: sampling where the model is going to provide a better fit to the data and improving the knowledge of objective function where the error is large.

Different forms can be specified for the acquisition function (Frazier, 2018). The expected improvement is one of these choices. The expected improvement algorithm compares the minimum value of the objective function found so far, f^n_{min}, with the value at any proposed new location, $f(x)$, which is given by the GP estimate of the objective function. Since we have to use the GP estimate of $f(x)$ rather than the actual value (which we do not know until after the evaluation), this is why the algorithm is called the *expected* improvement (hereafter EI).

The EI can be written in terms of a utility function:

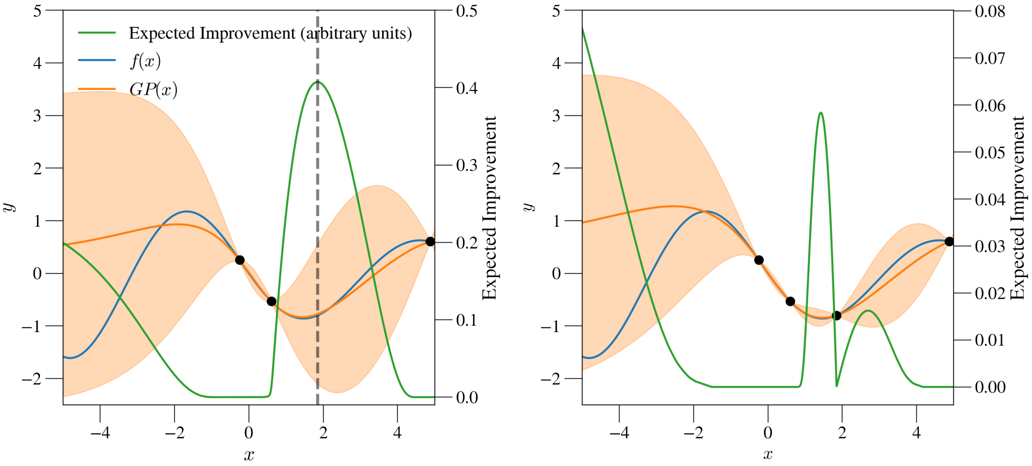
$$u(x) = \max\left(0, f^n_{\rm min} - f(x)\right), \tag{4.8}$$

which is the reduction in our metric at point x in the parameter space, as compared with our minimum evaluation so-far at iteration n, $f^n_{\rm min}$. The utility function is equal to zero unless there is a reduction in the value of f at the new position x, compared to the previous minimum value $f^n_{\rm min}$. The acquisition function in this case then is the expectation value of the utility $u(x)$ at point x. Since the GP provides a probability distribution of the evaluation of the function we are investigating at x, we have to integrate over the possible values and their respective probabilities to calculate an expected improvement at parameter space point x

$$a_{\rm EI}(x) = \mathbb{E}[u(x)] = \int_{-\infty}^{f^n_{\rm min}} (f^n_{\rm min} - f(x)) GP(x) {\rm d}x, \tag{4.9}$$

where we are integrating over the probability distribution given by the GP.

A demonstration of how the expected improvement function works in Bayesian optimization is shown for a 1-dimensional parameter space in Fig. 4.2. Here we show one iteration of the EI Bayesian optimisation algorithm on a toy model. The left panel shows a function in dark blue, which represents the unknown function we are trying to optimize (later on, this function will be a metric based on the error between the predictions of the `GALFORM` model and the chosen calibration datasets, in a much higher dimensional space). The left panel of Fig. 4.2 shows that this function has been evaluated at three locations in the paramter space, shown by the filled circles. We are trying to find the minimum of this function. The orange line shows the GP fit to these three points, and the shaded region shows the 2σ uncertainty on the GP model predictions. The uncertainty shrinks to zero at the points in the parameter space for which the function has been evaluated. The acquisition function, Eqn. 4.9, is shown in green, in arbitrary units since we are only



]

Figure 4.2: An illustration of one iteration of the expected improvement (EI) algorithm. The left panel shows an example function (blue solid line) and a Gaussian process (GP) posterior (orange solid line) fit to 3 evaluations of the function (black solid points). The orange shaded region shows the 3σ confidence interval of the GP. The green curve shows the EI at each point (right axis), which corresponds to the expectation integral of the GP posterior below the minimum evaluation so far (i.e. how much we expect to improve upon the current minimum evaluation at each point x). The right panel shows the updated GP posterior and EI curve after evaluating the function at the point of maximum expected improvement, as shown by the black dashed line in the left panel. At this point, the next evaluation would be chosen at the far left of the right panel.

concerned with its relative amplitude at two different values of x. The black dashed line shows the peak of the acquisition function, and corresponds to the location in the input dimension (i.e. the value of the parameter x) that the algorithm will next sample, as a result of the expected improvement quantity being largest for this value of x. This is the location in parameter space at which we can expect to run the full model evaluation to make the best overall improvement to the GP reproduction of the objective function f. In the right panel, we have sampled the function at this point, and we update our Gaussian process model and acquisition functions. The acquisition function now is largest on the far left of the plot, at the most negative value of x plotted, which is where we would sample the function next.

4.2.5 Dataset selection for parameter calibration

Our aim is to test the ability of the `GALFORM` galaxy formation model to reproduce observations of dusty star-forming galaxies, SMGs, which tend to be high redshift objects, as well as matching the properties of the low-redshift galaxy population. In Chapter 3 we used nine observational datasets measured for local galaxies to calibrate `GALFORM`. To make the analysis simpler to follow, here we restrict ourselves to the $z = 0$ K-band luminosity function, and supplement this with the number counts and redshift distribution of SMGs. We investigate if the best fitting models can reproduce the local 850μm luminosity function and selected other $z = 0$ galaxy observations in § 4.2.3, but emphasize that these datasets are not used in the calibration.

Below we list the observational datasets used to constrain the model parameters:

- For the local K-band LF, we compare to the estimate from Kochanek et al. (2001).
- For the SMG redshift distribution, we compare to the measurements from Dudzevičiūtė et al. (2020). These authors started from a sample of SCUBA sources with 850 micron fluxes of $S_{850} \geq 3.6mJy$, and made follow-up observations with ALMA. The redshift distribution is compiled from the stacked redshift probability distributions inferred using the MAGPHYS model (Da Cunha et al., 2015). The sample covers 1 square degree. The completeness of the sample improves if we use a somewhat brighter flux cut of 4 mJy; this results in a median redshift of $z = 2.8$.
- For the number counts, at the bright end we compare to Stach et al. (2018), which uses the same data as Dudzevičiūtė et al. (2020), and at the faint end we compare to Chen et al. (2013).

To correct the Dudzeviciute et al. (2019) sample for completeness, we re-weight sources according to the results derived in Geach et al. (2017), who gives an estimate

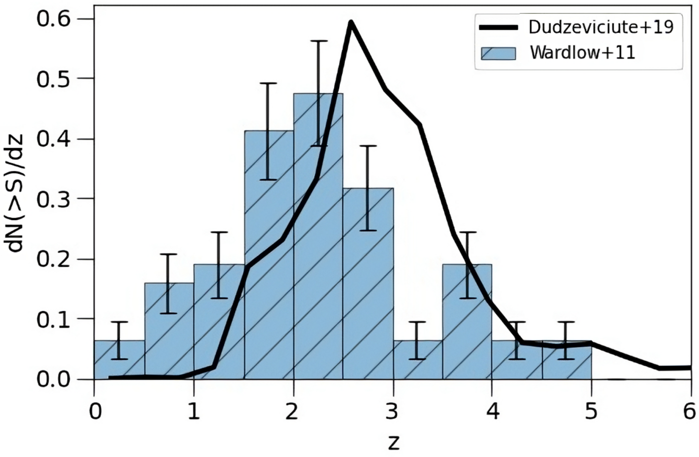


Figure 4.3: Comparison between the redshift distribution for SMGs brighter than 4 mJy inferred by Dudzevičiūtė et al. (2020) (black solid line) and Wardlow et al. (2011) (hatched histogram). Here, we calibrate GALFORM to the redshift distribution estimated by Dudzevičiūtė et al. (2020).

of incompleteness with 850 μm flux. That is, we scale the redshift probability density functions provided in Dudzeviciute et al. (2019) using factors from Geach et al. (2017) to correct for any incompleteness before summing to calculate a total redshift distribution.

It is important to note that the redshift distribution of SMGs that we obtained using the data from Dudzevičiūtė et al. (2020) has a somewhat *higher* median redshift than previous estimates used to calibrate the models. Wardlow et al. (2011) reported a median photometric redshift for their sample of $z = 2.2 {\pm} 0.1$, though this figure increases when considering possible counterparts to unidentified SMGs and applying a brighter flux limit (applying $S_{870} > 4$mJy increases the median redshift to $z = 2.5 \pm 0.5$). The two estimates of the SMG redshift distribution for sources brighter than 4 mJy are shown in Fig. 4.3. The model calibrations performed in Lacey et al. (2016) and Baugh et al. (2019) thus used an SMG redshift distribution with a somewhat lower median redshift than the calibration data employed here. It is therefore interesting to find out if the model with a top-heavy IMF can still reproduce the higher median redshift found by Dudzevičiūtė et al. (2020), and if this has any implications for the value of the recovered slope of the IMF adopted in starbursts.

We will see later that the GALFORM predictions for the redshift distribution of SMGs show a series of spikes. At intermediate and high redshifts, these spikes arise due to the limited number of halo merger trees used in the calculation. Using many more halos would smooth out these spikes at the expense of greatly increasing the computational cost for each full model evaluation. The spike at low redshift is different, being robust to using more halos. The lack of a similar spike at low redshift in the observational estimates could be connected to the choice of fields which avoid local objects to focus on more distant sources. To remove any issue over the low redshift spike, we exclude $z < 0.8$ from the comparison between the model and observed SMG redshift distribution.

4.2.6 Validation of the optimisation approach

Before running the Bayesian optimisation strategy on the full GALFORM model, we first assess whether this method is likely to be successful. We also need to gain some insight into the convergence properties of the optimization process (or parameter calibration) so that we have some guidance as to when to end the search. To do this, we build a simple emulator of GALFORM so that we can get a better understanding of the optimisation routine, without requiring many time-consuming evaluations of the GALFORM model.

To carry out this test, we first build a neural network emulator from 500 runs of the full GALFORM model, with samples drawn from the space defined by the parameter ranges given in Table 4.1 using a Latin hypercube. The trained neural network provides us with a fast alternative to running GALFORM, and allows us to test our optimization strategy before applying it to the full model. This is similar to the approach taken in Chapter 3. One might ask why we cannot simply use this emulator to perform the model calibration instead of using Bayesian optimisation. The observables considered here are more complicated for the neural network to learn than those used to calibrate the model in Chapter 3; in particular the redshift distribution of SMGs is hard to reproduce accurately, without running many more full `GALFORM` evaluations. Hence, if we used the trained neural network to give a set of parameters deemed to be the best fitting ones, these may not lead to the full `GALFORM` model reproducing the calibration data closely. Also, as commented above, because we are predicting statistics that are computed from the model ouptut at many redshifts, each model evaluation has a higher computational overhead than in Chapter 3. Hence, whilst the emulator is too approximate to use in a model calibration, it has similar properties to `GALFORM` and serves to let us test the optimization process.

Here, we use a neural network model from the `TensorFlow` software library (Abadi et al., 2015). From our full `GALFORM` runs, we have sets of parameters $\mathbf{x}$, and an associated output $\mathbf{y}$, which corresponds to the `GALFORM` model predictions for the datasets we are considering (for example, the values of the K-band luminosity function in different magnitude bins). We then calculate the error or distance between the `GALFORM` predictions and the calibration datasets as given by the mean absolute error, MAE:

$$\mathrm{MAE} = \frac{1}{n}\sum_i (y_i - y_{\mathrm{obs},i}), \tag{4.10}$$

where y_{obs} corresponds to the observational calibration dataset which has n bins. Importantly, we scale each dataset to lie within the range [0,1], so that we can combine the errors defined in this way for different datasets in a consistent way into a single distance value or metric. In the case of the K-band LF and the SMG number counts it is the logarithim of these quantities that is rescaled. Later, we will include a weighting scheme so that we can multiply the error for an individual dataset by a fixed value so that this dataset is given more weight during the optimization process.

We train the neural network to predict the error between the `GALFORM` prediction and the observational data, given the parameter values $\mathbf{x}$, so that we have a fast-to-evaluate but approximate emulator of `GALFORM` on which to test our Bayesian optimization strategies. By building an emulator which we can evaluate quickly, we allow ourselves to approximately find the global minimum error in a short amount of time, and then use this result to test our Bayesian optimization strategy. Of course, the emulated model will not correspond exactly to running the full `GALFORM` model, but it will be of similar complexity and so allows us to assess the optimization methods, albeit approximately. We can then use these experiments to apply the optimization strategy to the full `GALFORM` model.

Having trained our neural network emulator to produce an approximate error between the `GALFORM` predictions for the K-band LF, the SMG redshift distribution, and the SMG number counts and their corresponding observational datasets, given a set of input parameters, we first find an approximate global optimum using an exhaustive MCMC search. We use a simple implementation of the Metropolis-Hastings algorithm (e.g. Robert 2015), using 20 chains, each 10 000 steps in length and starting from a different, randomly chosen location in the parameter space, to locate the set of parameters which returns the lowest approximate error for the neural network emulator, as judged by the MAE metric.

Next, we can test our optimization strategies to see if we can find a set of best-fitting parameters that gives a comparable error or MAE to the MCMC approach above. To do this, we initially perform a pseudo-random sampling of the parameter space using Sobol sampling (e.g. see Oleśkiewicz & Baugh 2020), drawing 2 samples per dimension of the parameter space to be searched (in our case 30 initial evaluations for a 15 dimension parameter space). We evaluate the neural network emulator at these points to return approximate errors, and fit the GP to the parameter-error pairs, and begin the expected improvement search. In this exercise, we use a batch-size of four. That means at each step we use EI to locate four points in the parameter space at which to evaluate the neural network emulator. We then update the GP again, and the process continues.

Fig. 4.4 shows the result of applying the EI Bayesian optimisation to the neural network emulator of `GALFORM`. The y-axis gives the error returned by the neural network, with the x-axis showing the total number of evaluations of the emulator carried out up to that point. The vertical dashed line marks the beginning of the EI routine, whereas the first 30 runs are pseudo-random Sobol sampling. The horizontal dashed line corresponds to the minimum error found using the more exhaustive MCMC search of the parameter space with the emulator, which is the target MAE for this exercise. We ran many trials of the optimization, as indicated by the shading, each time with a different random seed. We find that the EI algorithm is able to locate an error close to the approximate global minimum within just 100 to 150 full model evaluations, with some runs converging on the minimum value in as few as 75 evaluations. We see very little improvement beyond 150 evaluations of the emulator.

Although the emulator is just an approximation of the full `GALFORM` model, it still represents a complex 15-dimensional function with properties similar to `GALFORM`, and the encouraging convergence of the EI algorithm allows us to confidently apply the methodology to the full model.

4.2.7 Applying Bayesian optimisation to GALFORM

To apply the Bayesian optimization method to the full `GALFORM` model, we first evaluate the model n times at points chosen in the parameter space using Sobol sampling (see Oleśkiewicz & Baugh 2020), where n is equal to twice the number of parameters we are investigating. In our case, since we are varying 15 model parameters, we draw an initial sample of 30 runs across the parameter space. Once these 30 `GALFORM` runs are in place we begin the optimization process applying the EI algorithm as described above to decide the location in parameter space at which to run new model calculations. Four new samples are drawn from the parameter space at a time (or equivalently we say that there is a batch size of 4), before updating the GP model with these runs and using the new EI to sample

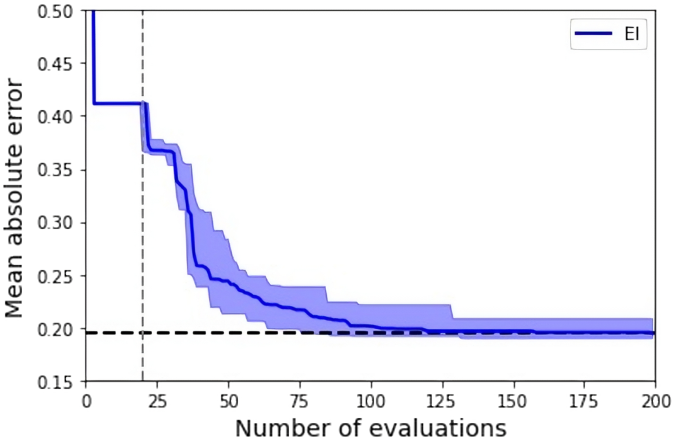


Figure 4.4: Performance of the Expected Improvement (EI) Bayesian Optimisation algorithm on a simple neural network emulator of GALFORM. Solid lines shows the median over 30 separate runs, and the shaded region shows the minimum to maximum range. The dashed horizontal line shows the global minimum found by MCMC.

again. We look to the experiment in the last subsection for guidance as to when to stop the optimization, and allow the runs to progress until at least 150 full model evaluations have been carried out. Once 150 runs have been completed, we decide to stop the optimization if the MAE has not improved significantly comapred to the MAE obtained from the best evaluation over the previous 25 steps.

4.3 Results

Here we present the main results of calibrating GALFORM in a 14 or 15 dimensional parameter space using Bayesian optimisation, applied to different combinations of calibration datasets (the local K-band LF, the number counts of SMGs and the redshift distribution of SMGs brighter than 4mJy). Throughout we consider two model variants, with the main difference being whether we fix the stellar IMF in bursts of star formation to be a solar neighbourhood IMF (a 14 dimensional parameter space, which we refer to as the universal IMF variant) or treat the power law slope of the IMF in bursts as a model parameter (a 15 dimension parameter space, called the dual IMF model). In § 4.3.1, we present the best fitting models for these two variants, treating all the calibration datasets equally (the corresponding parameter values are listed in Table 4.2). In § 4.3.2 we show how the model fits change if more weight is placed on reproducing the local calibration data. § 4.3.3 presents model predictions for observational datasets that were not used in the parameter calibration.

4.3.1 Calibrations

The results of the model calibration are shown in Fig 4.5 for the case in which each of the three datasets is given equal weight in the MAE metric. The associated parameter values are listed in Table 2. (Note these plots also show a special case in which the local K-band luminosity function is given extra weight; this case is discussed in § 4.3.2.) The left panel of Fig. 4.5 shows the `GALFORM` predictions of the SMG redshift distribution, the middle panel shows the low redshift K-band LF, and the right panel the SMG number counts. We also include, for reference, the predictions from the Baugh et al. (2019) model, which are shown by the black dashed lines in each panel. This model has, in general, a lower median redshift than the best fitting models we find here, as it was calibrated to the redshift distribution from Wardlow et al. (2011), before the redshift distribution from Dudzevičiūtė et al. (2020) became available. The Baugh et al. model is a recalibration of the model from Lacey et al. (2016) following its implementation in the P-Millennium N-body simulation. This simulation has updated cosmological parameters and a superior mass resolution compared with the N-body simulation used by Lacey et al. The recalibration carried out by Baugh et al. focussed on the local K-band luminosity function, but not the number counts or redshift distribution of SMGs. Furthermore, the recalibration carried out by Baugh et al. was essentially a perturbation of the Lacey et al.fit, considering a much smaller parameter space of only three parameters, without a framework for an extensive search of the space.

We find that the model with a universal Chabrier IMF and an equal weighting of the calibration datasets is unable to match the local K-band LF, the SMG

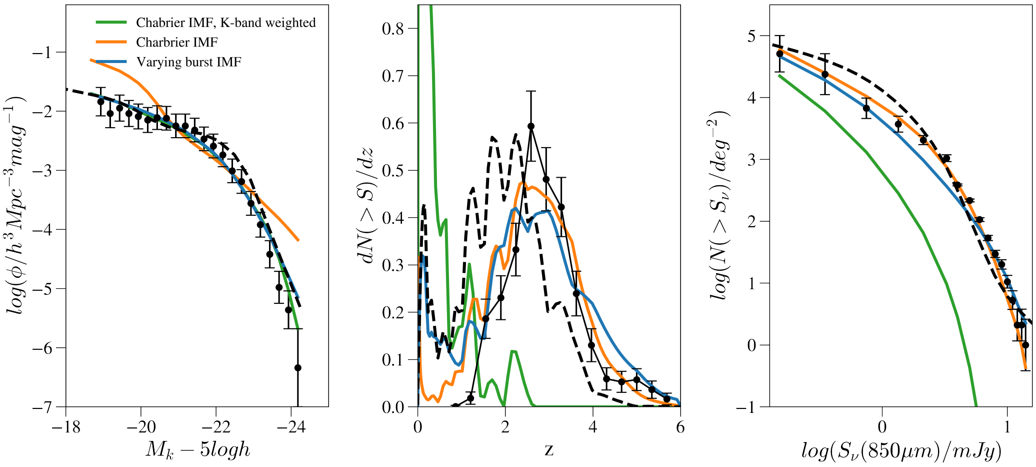


Figure 4.5: A comparison of the model predictions with the three calibration datasets under consideration (the parameters of these models are given in Table 2). Left: The $z = 0$ K-band LF. Center: the normalized SMG redshift distribution. The spikes in the model predictions for the redshift distribution are artifacts due to the number of halos simulated. Right: the SMG number counts. In each case the black points with error bars show the observational data. For the SMG redshift distribution, we calibrate to data from Dudzevičiūtė et al. (2020). For the local K-band LF, we calibrate to data from Kochanek et al. (2001), and for the SMG number counts, we calibrate to data from Stach et al. (2018) at the bright end, and Chen et al. (2013) at the faint end. The orange solid curves show the model which assumes a universal Chabrier IMF in all modes of star formation. The green lines show the predictions from a model that also adopts a universal Chabrier IMF, but which is calibrated to give an improved fit to the low-redshift K-band LF by increasing the weight given to this dataset in the parameter optimisation. The blue lines show a model in which the IMF slope in bursts is allowed to vary according to $\mathrm{d}n/\mathrm{d}lnm \propto m^{-x}$, where x is an adjustable parameter. For reference, the black dashed line shows the `GALFORM` model from Baugh et al. (2019): this model was calibrated using an earlier measurement of the SMG redshift distribution from Wardlow et al. (2011), which has a lower median redshift than the Dudzevičiūtė et al. (2020) data.

redshift distribution, and the SMG number counts *at the same time*. When the calibration data does not include SMG observations, the universal IMF variant is able to produce a good match to the observed K-band LF at $z = 0$ (see Chapter 3). However, when the calibration datasets include SMG observations, this variant returns a poor match to the observed K-band LF, with large excesses at the faint- and bright-ends compared to the observational data. At $M_K - 5\log h = -23$, this model predicts ten times more galaxies than are observed: this difference is many times greater than the uncertainty in the observational estimate. At the faint end, the model over predicts the number of galaxies by at least a factor of three. Interestingly, this model prefers very low values for both the $f_{\rm burst}$ and $F_{\rm stab}$ parameters, as can be seen in Table 4.2 The parameter $f_{\rm burst}$ corresponds to the mass ratio (accreted satellite galaxy over central galaxy) threshold for a burst of star formation to occur following a merger (with the universal IMF model preferring a value of $f_{\rm burst} \approx 0.05$), whereas $F_{\rm stab}$ sets the threshold for a burst of star formation caused by the galactic disk becoming dynamically unstable. The preferred parameter value of $F_{\rm stab} = 0.53$ corresponds to a model in which there are no disk instabilities (in the formulation used from Efstathiou et al. (1982) the ratio on the left hand side of the expression in Eqn. 2.4 is equal to 0.61 for a self-gravitating disk; in the model only disks for which Eqn. 2.4 exceeds this value and the adopted value of $F_{\rm stab}$ are allowed to become unstable and experience bursts). This combination of parameters allows the model galaxies to retain larger reservoirs of gas, which are used up in star bursts following mergers rather than disk instabilities. Though both models prefer a higher value of $\nu_{\rm SF}$, the parameter which controls the rate of quiescent star formation in disks, than is suggested by observations of galactic disks in the local Universe, the value of this parameter for the universal IMF model is significantly higher at $\nu_{\rm SF} = 3.48$ than the value preferred in the other two models listed in Table 4.2. This about a factor of 5 higher than suggested by local measurements (Bigiel et al., 2011). This combination of parameters, with disk instabilities in effect turned off, a high rate of minor-merger driven bursts, and much stronger quiescent star formation rates, allows the model to generate the star formation necessary to match the SMG redshift distribution and counts. However, this behaviour means that the low redshift K-band LF is not matched adequately, since it leads to an excess of large, bright disk galaxies which have not undergone a recent merger, and which now make up the bright-end of the LF.

In the case of the dual IMF variant, the added flexibility of varying the slope of the IMF in starbursts compared to quiescent star formation allows the model to match both the local K-band LF, as well as producing realistic number counts and redshift distribution for SMGs. In this model, we see more typical values for the disk instability parameter ($f_{\mathrm{stab}} = 0.75$), and a smaller number of bursts triggered by minor mergers due to a higher value of $f_{\mathrm{burst}} = 0.17$. The model prefers an IMF slope parameter, $x = 0.7$, which is *more* top-heavy than that assumed in Lacey et al. (2016); Baugh et al. (2019), who adopted $x = 1$ (though those papers were considering a larger number of calibration datasets), but is less flat than assumed in early models (Baugh et al., 2005).

The optimization of the two `GALFORM` variants in shown in Fig. 4.6, in which we plot the MAE metric as a function of the number of full runs of the mode carried out. A solution is found for the universal IMF variant after 90 runs of the full `GALFORM` model, after which there is no change in the value of the MAE for this case on carrying out further model runs. In the case of the variant with a dual IMF, 140 model runs are required to find a best-fitting model. The minimum MAE for the dual IMF variant is almost a factor of two smaller than that for the universal IMF model, confirming that the dual IMF best fit gives a better reproduction of the calibration data.

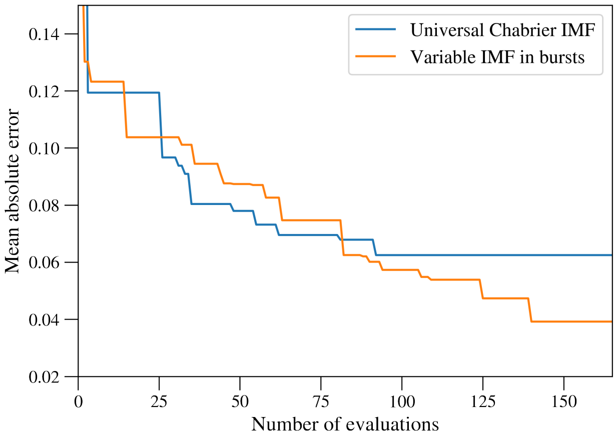


Figure 4.6: The minimum mean absolute error, MAE, of the GALFORM model predictions as a function of the number of full model evaluations carried out, with respect to the three calibration datasets: the $z = 0$ K-band LF, the SMG number counts, and the SMG redshift distribution. The blue line shows the universal IMF tmodel and the orange line shows the variable IMF model, in which the slope of the IMF in bursts is a parameter. The optimization is terminated once 150 model runs are reached and there is no significant improvement in the MAE over the proceding 25 runs.

4.3.2 Enforcing low-redshift agreement

When optimizing the models in the previous subsection the three calibration datasets were weighted equally when combining their respective errors into a single value for the MAE metric. The variant with a dual IMF matches all three calibration datasets reasonably well (as shown by the blue curves in Fig. 4.5). However, the calibration of the simplest variant with a universal Chabrier IMF returned a model which matched the counts and redshift distribution of SMGS, but gave a poor reproduction of the low-redshift K-band LF (orange curves in Fig. 4.5).

It is interesting to know how the predictions for the SMG number counts and redshift distribution are degraded if we enforce reasonable agreement with the low-redshift K-band LF. We can achieve this by re-calibrating the universal Chabrier IMF variant with a triple weighting applied to the contribution of the low-redshift K-band LF to the MAE, and single weightings for each of the SMG calibration datasets.

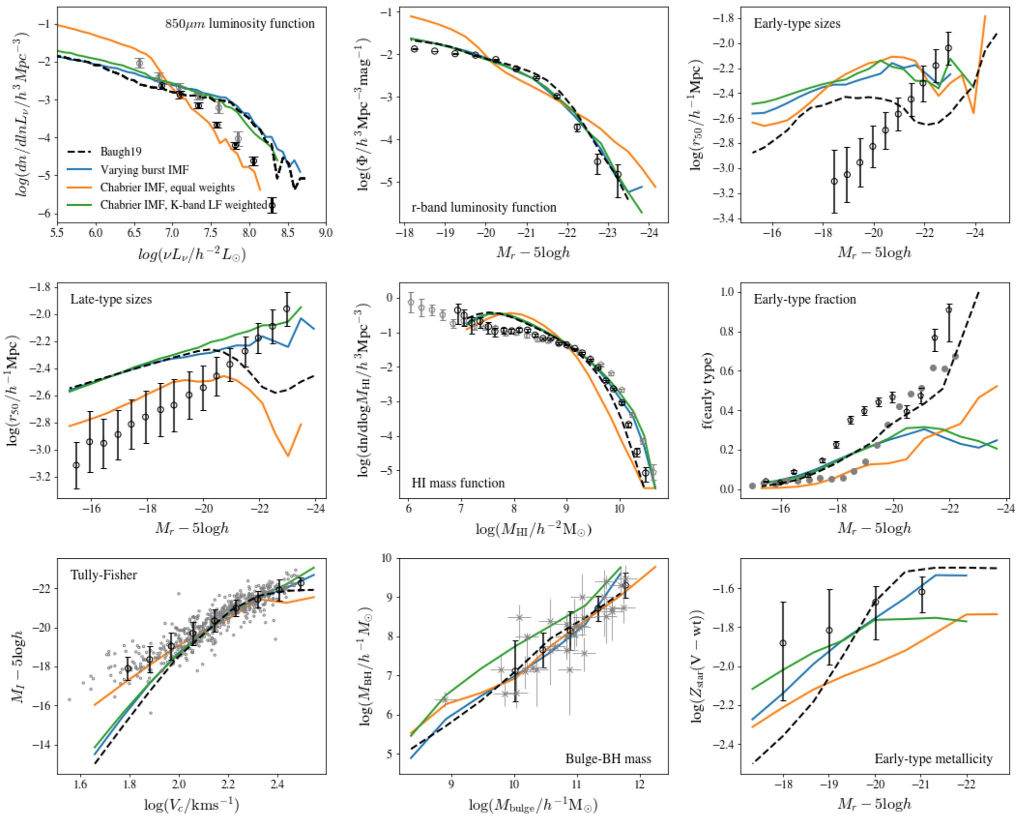


Figure 4.7: Low redshift predictions for the three model calibrations. The `GALFORM` predictions for the model with a variable IMF slope in bursts is shown in blue, the equal-weighted calibration assuming a universal chabrier IMF in orange, and the calibration in which we gave a higher weight to the low-redshift K-band LF, again assuming a universal Chabrier IMF, is shown in green. The black dashed lines shows the predictions for the model calibration performed in Baugh et al. (2019). For the 850 μm LF we compare to data from Vlahakis et al. (2005) (grey circles) and Dunne et al. (2000) (black circles). For the r-band LF we compare to data from Driver et al. (2012). For the early-and late-type sizes, we compare to data from Shen et al. (2003). For the HI mass function, we compare to data from Zwaan et al. (2005) (black circles) and Martin et al. (2010) (grey circles). For the early-type fraction, we compare to data derived from Moffett et al. (2016) (black sysmbols; A. Moffett, private communication), and to data from González et al. (2009) (grey symbols). For the Tully-Fisher relation, we compare to a subsample of Sb-Sd galaxies from the Mathewson et al. (1992) catalogue selected by de Jong & Lacey (2000) (grey points show indivdual galaxies, black points with bars show the binned median and 10-90 percentile range). For the bulge-BH mass relation, we compare to data from Häring & Rix (2004), and for the early-type metallicity, we compare to data from Smith et al. (2009). Note that the models were not calibrated to the datasets plotted in this figure. (Further details on the observational datasets plotted here can be found in Chapter 3.)

Running the optimization routine again with a higher weighting to the low-redshift K-band LF, we find that the model is no longer able to produce a realistic SMG redshift distribution and counts while simultaneously matching the low redshift K-band constraints. We show the results in Fig. 4.5 with the green curves. Despite a much improved match to the K-band LF as expected, the predictions for the number counts are too low at all fluxes, with the offfset ranging from a factor of three at the faint end to more than a factor of a hundred at 3 mJy and brigther; the predicted median redshift is much lower than the observed one. In this weighting case, the best-fitting model parameters are more similar to the model which assumes a different IMF slope in bursts, with $f_{\rm stab} \approx 0.7$.

4.3.3 Further predictions at low redshift

Having calibrated the model to the low-redshift K-band LF, the SMG numbers counts, and the SMG redshift distribution, we now explore the predictions for other low-redshift properties of the galaxy population. In particular, we explore the predictions for the $z = 0$ $850\mu m$ and r-band LFs, the early- and late-type galaxy sizes, the HI mass function, the dependence of the early-type fraction on r-band magnitude, the I-band Tully-Fisher relation, the bulge- black hole (BH) mass relation, and the early-type metallicity.

The model predictions for the above datasets are shown in Fig. 4.7. The blue solid lines show the predictions for the dual IMF GALFORM variant, which allowed a variable IMF slope in bursts, whereas the orange and green lines show the predictions for variants which assumed a universal Chabrier IMF. The orange line represents model predictions with equal weightings applied to the three calibration datasets, whereas the model shown by the green line used a higher weighting to the low-redshift K-band LF during calibration. Interestingly, but perhaps not surprisingly, we see that the dual IMF model and the low-redshift K-band LF weighted models predictions are very similar. As discussed in Chapter 3, the low-redshift K-band is most sensitive to the choice of supernova feedback parameters

Table 4.2: The best-fitting parameters for the three optimisation runs considered. The second column shows the parameter values for the dual IMF variant, which treated the IMF slope in bursts, x, as a parameter, which was jointly optimised along with the other parameters. The third column shows the best-fitting parameters for the variant with a universal Chabrier IMF with equal weighting attributed to each dataset, and the model in the fourth column also assumes a universal Chabrier IMF, but with an increased weighting applied to the low-redshift K-band LF. * indicates that this parameter was held fixed. x gives the slope of the IMF above $1M_{\odot}$.

Model variant:	Dual IMF (x_{IMF} in bursts)	Universal IMF	Universal IMF (extra weight to K LF)
Parameter name			
$F_{\rm stab}$	0.746	0.535	0.77
$\gamma_{\rm SN}$	3.55	1.48	3.06
$\alpha_{\rm cool}$	2.99	3.8	2.90
$\alpha_{\rm reheat}$	1.83	2.14	2.11
$V_{\rm SN,\ disk}$ (km s^{-1})	293	774	383
$V_{\rm SN,\ burst}$ (km s^{-1})	349	399	194
$f_{\rm ellip}$	0.383	0.385	0.212
$f_{\rm burst}$	0.166	0.056	0.261
$\nu_{\rm SF}$	1.94	3.48	2.186
$f_{\rm SMBH}$	0.014	0.037	0.025
$\tau_{*\rm burst,\ min}$ (Gyr)	0.145	0.157	0.176
$f_{\rm cloud}$	0.452	0.404	0.427
$t_{\rm esc}$ (Gyr)	0.006	0.006	0.009
$\beta_{\rm burst}$	1.53	1.76	1.50
x	0.67	1.35*	1.35*

($V_{SN,disk}$,$V_{SN,burst}$), which are similar in the case of these two calibrations. These parameters also dominate the majority of the other predictions, leading to very similar predictions. For example, the Tully-Fisher relation, and late-type sizes (again as shown in Chapter 3). are dominated by the effects of the supernova feedback parameters, at least at fainter magnitudes. We see that the blue and green lines are almost identical for these datasets, where the orange line better matches the faint end of both relationships.

All of the models fail to match the bright end of the early-type fraction brighter than $M_{\rm r} - 5\log h = -20$, with the low-redshift weighted Chabrier model, and the dual IMF model actually predicting a constant or gently decreasing ealry-type fraction at the brightest magnitudes. The models are therefore not producing enough early-type galaxies. This is due to a more complex interplay of parameters, and our method does not give us the tools to investigate this thoroughly. The Baugh et al. calibration, however, is able to match both the early-type fraction, as well as the three calibration datasets adequately while assuming a top-heavy IMF in bursts. No calibration assuming a Chabrier IMF was able to simultaneously match the three calibration datasets.

Here, we have chosen to calibrate our model to the SMG redshift distribution at $z > 1$, as McAlpine et al. (2019) did when investigating the SMG population in the EAGLE simulation of Schaye et al. (2015). The local $850\,\mu m$ LF therefore offers a constraint on this population at low redshift. We find, interestingly, that the equal-weighted Chabrier IMF fit better matches the local $850\,\mu m$ LF, while producing very poor fits to the K- and r-band LFs. On the other hand, the GALFORM variant with a dual IMF over-predict the bright end of the local $850\,\mu m$ LF.

4.4 Discussion

We have assessed if the GALFORM galaxy formation model is able to match observations of SMGs, which are typically high redshift galaxies, and local galaxies, particularly the abundance of bright galaxies, at the same time. In particular, we have tested the the assertion made by Baugh et al. (2005) that properties of the SMG and local galaxy populations can only be matched if the IMF in star bursts is assumed to be top-heavy.

Our analysis contains several new features compared to previous work. We have used Bayesian optimization to carry out an exhaustive search of a parameter space with a large number of dimensions (15 dimensions if the slope of the IMF is allowed to vary in star bursts). This method allows use to search the parameter space without requiring a large number of runs of the full GALFORM model, which is computationally expensive when making predictions for number counts and redshift distributions. We are also able to formally rate how well the models reproduce observational data using a metric. This replaces the old fashioned variation of one parameter at a time and "chi-by-eye" used to reach the conclusion about the need for a top-heavy IMF in bursts in our previous work (Baugh et al., 2005; Lacey et al., 2016). With Bayesian optimisation, we are able to settle once and for all the question of can the model reproduce the SMG and local galaxy populations with a solar nieghbourhood IMF.

We used three observational datasets to calibrate the model parameters, a process also referred to as model optimisation: the $z = 0$ K-band LF, the SMG number counts, and the SMG redshift distribution. We attempted to reproduce these datasets using two variants of the galaxy formation model: (i) a universal IMF model in which a solar neighbourhood IMF was imposed on quesicent star formation in disks and in bursts of star formation and (ii) a dual IMF model, with a solar neighbourhood IMF for star formation in disks and a power law IMF in bursts, the slope of which is treated as a model parameter.

Our calibrations confirmed that within the GALFORM framework, even when varying a large number of relevant parameters not explored in previous work, the model was not able to accurately match the SMG constraints *and* the local K-band LF when assuming a universal Chabrier IMF. This confirms the conclusion of Baugh et al. (2005) and a series of studies since then (e.g. Lacey et al. 2008; Lacey et al. 2016), though this time we have demonstrated this through an exhaustive and automatic search of the parameter space. We have also explored the calibration to the most recent SMG redshift distribution data (Dudzevičiūtė et al., 2020), which has a median redshift that is higher than previous datasets (e.g. Wardlow et al. 2011). When assuming a universal Chabrier IMF and enforcing a good fit to the low-redshift K-band LF, we find a similar result to those obtained with the EAGLE simulation (McAlpine et al., 2019); namely that the best-fitting GALFORM model in this case gives a very large underprediction of the SMG number counts, and produces almost no bright sources at high redshift (i.e. with $z > 1$).

The parameter search used here is based on Bayesian optimisation. The model is first run for a small number of calculations, two per dimension of the space being explored, before the optimization is started. The locations of these initial model evaluations are chosen using a Latin hypercube (see for exampple Oleśkiewicz & Baugh 2020). We use a metric to determine a best-fitting set of model parameters which quantifies the distance between the model prediction and the calibration datasets, after applying a suitable rescaling so that the datasets cover the same dynamic range. Our knowledge of the metric is given by a Gaussian processes (GP), which gives the value of the metric at any point in the parameter space and the uncertainty on this value. The model is optimized by calculating the expected improvement (EI) for proposed new model evaluations in the parameter space. Following four new evaluations, the GP is updated. This process is repeated until there is no significant improvement in the metric used to select the best-fitting set of parameters.

To assess this Bayesian optimization approach, we first made use of a neural network emulator to act as a surrogate for running the full GALFORM model. We trained the neural network emulator to predict the GALFORM output given an initial set of 500 runs of the full model. This allowed us to test the convergence properties of the algorithm on an approximate GALFORM model, without being accurate enough to replace the Bayesian optimization itself. We ran the optimization routine 30 times on this emulated model, varying the starting point in the parameter space and found that the algorithm converged to a very similar minimum error and parameter set in each case. Although this emulator only approximated the GALFORM output, it gave us the confidence to apply the Bayesian optimization routine to the full model with some understanding of the convergence properties of the algorithm within a similarly complex and high-dimensional setting.

We found that the Bayesian optimization methods are able to find good fits to the data within a very small number of evaluations. For example, in Chapter 3, a sample of around 1000 runs was used, though we were calibrating to a larger number of datasets. Earlier works, such as Henriques et al. (2020) used an even larger numbers of runs. We find therefore that this method is significantly faster than other methods explored to calibrate semi-analytic galaxy formation models, converging in fewer than 200 full evaluations of the full model.

On the other hand, in return for this speed-up, we get a diminished sense of the effects of different parameters and their interactions. In previous emulator-based works (Bower et al., 2010; Vernon et al., 2010; Henriques et al., 2009, e.g), an emulator is built from a large number of runs and, as in Chapter 3, is assessed for accuracy across the whole model parameter space. This then allows the models parameters to be comprehensively explored, and easily extended to include extra datasets. Using Bayesian optimization, we only get a limited sense of the effects of the parameters across the whole space, in return for a very fast global optimization routine. In Chapter 3, we were able to build an accurate emulator across a smaller (10-dimensions, rather than the 15 considered here) parameter space. This allowed us to explore the implications of calibrating to a diverse set of datasets, and the tensions between them with only the overhead cost of the initial sample of 1000 runs. We were also able to run full MCMC explorations of the range of parameters which were able to produce acceptable matches to the observational datasets. Here, however, we do not get a full sense of the range of parameters which could match an observational dataset as the routine is simply searching from the global optimum rather than inferring the posterior distribution of the parameters.

We are able to conclude confidently that we were not able to find any set of parameters, assuming a universal Chabrier IMF, which was able to match the low-redshift K-band LF and the SMG redshift distribution and number counts simultaneously, in line with what was understood from ‘by-hand’ searches (Baugh et al., 2005; Lacey et al., 2016), and that within GALFORM, including the flexibility of a top-heavy IMF in bursts allows us to fit these datasets simultaneously.

4.5 Conclusions

We have explored the application of Bayesian optimization to the calibration of the semi-analytic galaxy formation model GALFORM, with the aim of exploring whether we are able to assess, in a statistically rigorous, automatic and exhaustive way, whether the model is able to match the constraints from dusty star-forming galaxies at high redshift(SMGs), as well as the abundance of galaxies in the low-redshift universe.

First, we found that using Bayesian optimisation we are able to quickly fit semi-analytic models to calibration datasets within fewer than 200 full model evaluations, a significant speedup over previous methods, though we do not get a full sense of the parameter space and the range of parameters which could provide acceptable fits.

Second, we found that within the GALFORM framework, for a variant with a universal solar neighbourhood IMF, even when considering varying a large number of parameters compared to previous works, we were unable to match the constraints from SMGs (namely, the redshift distribution and number counts) while also matching simple low-redshift constraints (the K-band luminosity function), at the same time. A variant where the slope of the IMF was allowed to vary as a parameter in starbursts was, however, able to simultaneously match all three calibration datasets. The optimisation routine found preferred values of $x = 0.7$ (where the Chabrier IMF has a slope $x = 1.35$ above $1 M_{\odot}$) for the IMF in starbursts.

In future work, we would explore how to extend the Bayesian Optimisation to recover the parameter ranges for acceptable models, rather than a single best-fitting model, and include more observational datasets in the calibration process, with a view to improving the model constaints and uncovering tensions between datasets.

Chapter 5

Conclusions

In this thesis, we have explored the application of machine learning to tackle the problem of automatically setting the parameters of the semi-analytic galaxy formation model GALFORM. In the past, this task has generally been performed by hand, as in the case of the most recent recalibrations, namely those by Baugh et al. (2019) and Lacey et al. (2016). This “chi-by-eye” methodology, though it has been generally successful and yielded important insights into the model (some of which we confirmed here), can be criticised because it is poorly defined and difficult to reproduce. It is also not clear to what extent such an approach is actually able to search the full parameter space, given the interactions between parameters and the high-dimensionality of the parameter space (see, for example the plots in Bower et al. (2010)).

To remedy this, in Chapter 3 we investigated techniques which would allow us to set the model parameters in an automatic, comprehensible, and reproducible way. Furthermore, we applied a sensitivity analysis to the model predictions for the calibration data to determine which parameters should be varied, as first performed for GALFORM by Oleśkiewicz & Baugh (2020). Much of the literature so far focused on either MCMC techniques (e.g. Lu et al., 2011) or iterative emulation methods (based of the methodology developed by Vernon et al., 2014). Generally, these methods, while successful, require a large number of model evaluations, and often require intervention to refine the parameter search using waves of calculations. First, we turned to techniques from deep learning to investigate whether we could build an emulator of the GALFORM across a comprehensive subset of the full model parameter space, and use this to calibrate the model to a variety of datasets, and hence to explore tensions between datasets. Using less than 1000 evaluations of the full model, we were able to build a demonstrably accurate GALFORM emulator across 9 key outputs and 10 model parameters. We were then able to use this fast-to-evaluate emulator combined with MCMC to explore the parameter space, and how well the model is able to match different combinations of datasets. Here, we focused on low-redshift predictions, and in this regime we were able to uncover new insights into the model parameters as well as re-discover previous parameter choices, but using an automatic calibration framework.

Next, in Chapter 4, we turned our attention to the high-redshift regime, and investigated a technique for calibrating the model parameters based on Bayesian optimization (BO) (Frazier, 2018). This method departed from previous approaches in that we did not build an explicit emulator of the GALFORM model. This method relies instead on building and refining a statistical model of the error term itself, that is some measure of the difference between the observational dataset and the GALFORM predictions. This was done using Gaussian processes. By searching the parameter space only in the most promising regions, we found that this algorithm was able to converge on the minimum error within just 200 model evaluations. This represents a significantly smaller computing time requirement than previous literature, which have generally used in the range of 3,000-20,000 runs (e.g. Henriques et al., 2020; Benson & Bower, 2010).

In return for this reduced number of evaluations, we have to forfeit a wider understanding of the parameter space. In Elliott et al. (2021), for example, we were able to build a statistical emulator of GALFORM across a large range of outputs and parameters. This gave us the flexibility of comprehensively exploring the model and applying sensitivity analysis techniques to directly quantify the importance of the different parameters across the many outputs. With the BO methodology, we are not able to do this, but we are able to fit the model to data with a very small number of full model evaluations. In doing so, we are able to confirm what is possible within the modelling framework, without committing to a full battery of runs to construct a comprehensive emulator. This can be particularly helpful, as in our case, when we are dealing with more expensive runs where we require a large number of output snapshots (for example, to calculate the SMG redshift distribution and number counts).

Having demonstrated the efficacy of the BO algorithm using a simple GALFORM emulator, we re-tested the assumption of a top-heavy IMF in bursts within the GALFORM model. Previous iterations of GALFORM (Baugh et al., 2005; Lacey et al., 2016; Baugh et al., 2019) had found that the top-heavy IMF was necessary to match simple low-redshift constraints such as the K-band luminosity function, as well as the constraints from the SMG population, namely their number counts and redshift distribution. We found that it was not possible to simultaneously match these constraints using a universal Chabrier (ie solar neighbourhood like) IMF, and in fact if we enforced agreement with the local K-band LF, the model vastly under-predicted the number counts and was unable to match the SMG redshift distribution. Including the flexibility of varying the IMF slope in bursts allowed these three constraints to be matched, as was previously understood. However, we were able to demonstrate this through a full parameter search, rather than the less well defined and cumbersome methodology previously employed.

With the methodology we have introduced in this thesis, it is possible to produce genuinely testable predictions. Rather than produce a single “best-fitting" model, the approach set out in Chapter 3 gives a range of acceptable models, all of which give equally good reproductions of the calibration data. The predictions of the model for observations that are not part of the calibration data come with a range of uncertainty, which has been missing from previous studies. This means that the predictions of the model can be ruled out in a formal, statistical way.

Future work would include extending the BO approach to also produce a range of acceptable models. This would require us to carry out a more extensive characterisation of the model parameter space, rather than simply finding the location of a best-fitting set of parameters for the chosen calibration data. The challenge is to do this without requiring a dramatic increase in the number of full model evaluations required. We touched on a possible way of doing this when we tested the BO method using an approximate emulator of `GALFORM` built using a neural network.

Finally, this thesis has contributed more broadly to the development of semi-analytical models. It has provided simple and straightforward methods to assess the ability of galaxy formation models to fit to an array of datasets. Given a new or updated dataset or simulation, it is often a time consuming process to update the model parameters in a satisfactory way. This work has investigated a number of tools that will allow model practitioners to much more quickly and robustly investigate new observational datasets, updated simulations or updated galactic physics. We have also demonstrated that SMG constraints are a real problem within the hierarchical galaxy formation framework, with no set of parameters able to accurately match both low-redshift constraints and the constraints from SMG number counts and redshift distribution without making changes to the assumed IMF. No self-consistent model as of yet is able to match these constraints simultaneously assuming a local IMF, and so this work demonstrates that this is a problem within galaxy formation modelling which requires serious attention.

A Supplementary figures

Here, we provide some additional figures figures to provide further illustration of points discussed in the main text.

Fig. A.1 illustrates the one-at-a-time effect of varying the parameters $f_{\rm stab}$ and $V_{\rm SN,\ burst}$, as a demonstration of their degenerate effects.

Fig. A.2 shows the one-at-a-time effect of varying $\nu_{\rm SF}$ on the K-band LF and late-type galaxy sizes. When fitting both the K-band LF and the late-type galaxy sizes, we see a decrease in the preferred value of $\nu_{\rm SF}$; Fig. A.2 demonstrates that this is because a lower $\nu_{\rm SF}$ counteracts the enhancement in the bright-end of the K-band LF caused by the higher value of $V_{\rm SN,\ disk}$ when including both constraints. We also see that reducing $\nu_{\rm SF}$ marginally improves the fit to the late-type galaxy sizes.

Fig. A.3 shows the accepted parameters of 20 MCMC chains when we fit the K-band LF (red), and when we fit to the K-band LF and the HI mass function (blue). Here, we see that including the HI mass function results in higher values of $V_{\rm SN,\ disk}$ being preferred. $\nu_{\rm SF}$ is also moved to the bottom end of the explored range, and $\alpha_{\rm ret}$ becomes more sharply peaked and takes slightly higher values.

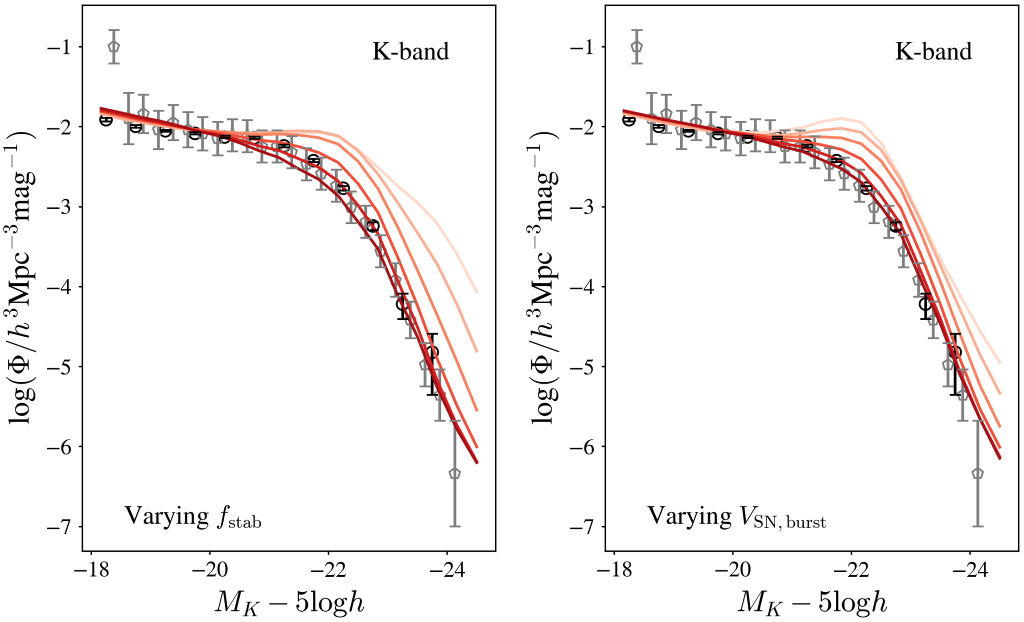


Figure A.1: Emulator predictions for one-at-a-time perturbations of the parameters $f_{\rm stab}$ (*left*) and $V_{\rm SN,\,burst}$ (*right*) around a fit to the K-band luminosity function. We vary the parameters between the full range given in Table 4.1. Darker colours correspond to higher values. The data shown correspond to those described in §3.2.3.

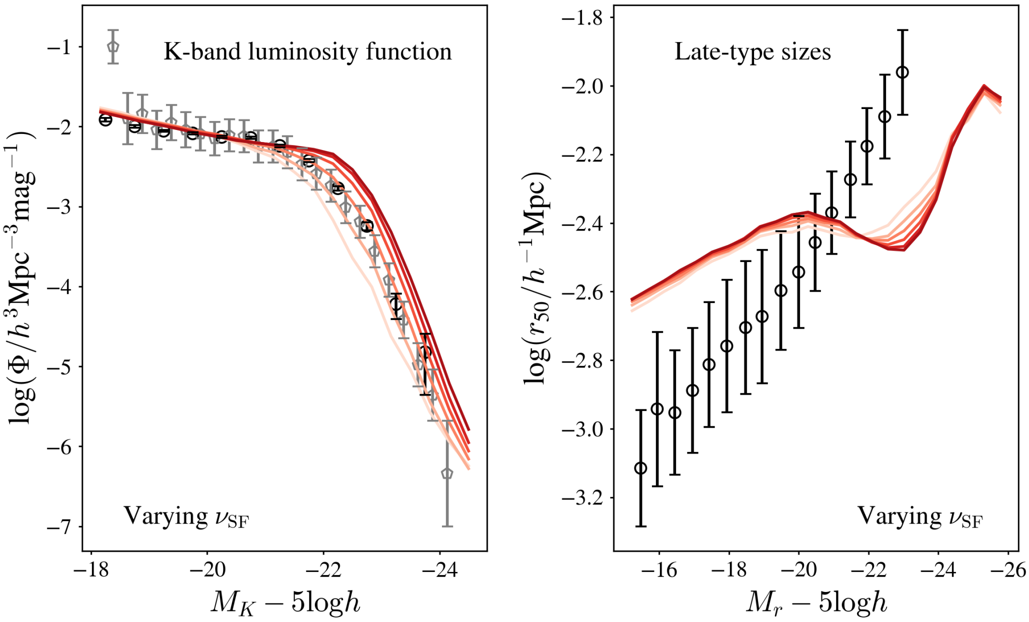


Figure A.2: Emulator predictions for one-at-a-time perturbations of the parameter $\nu_{\rm SF}$ for the K-band luminosity function (*left*) and the late-type galaxy sizes (*right*) around a fit to the K-band luminosity function. We vary the parameters between the full range given in Table 4.1. Darker colours correspond to higher values. The data shown correspond to those described in §3.2.3.

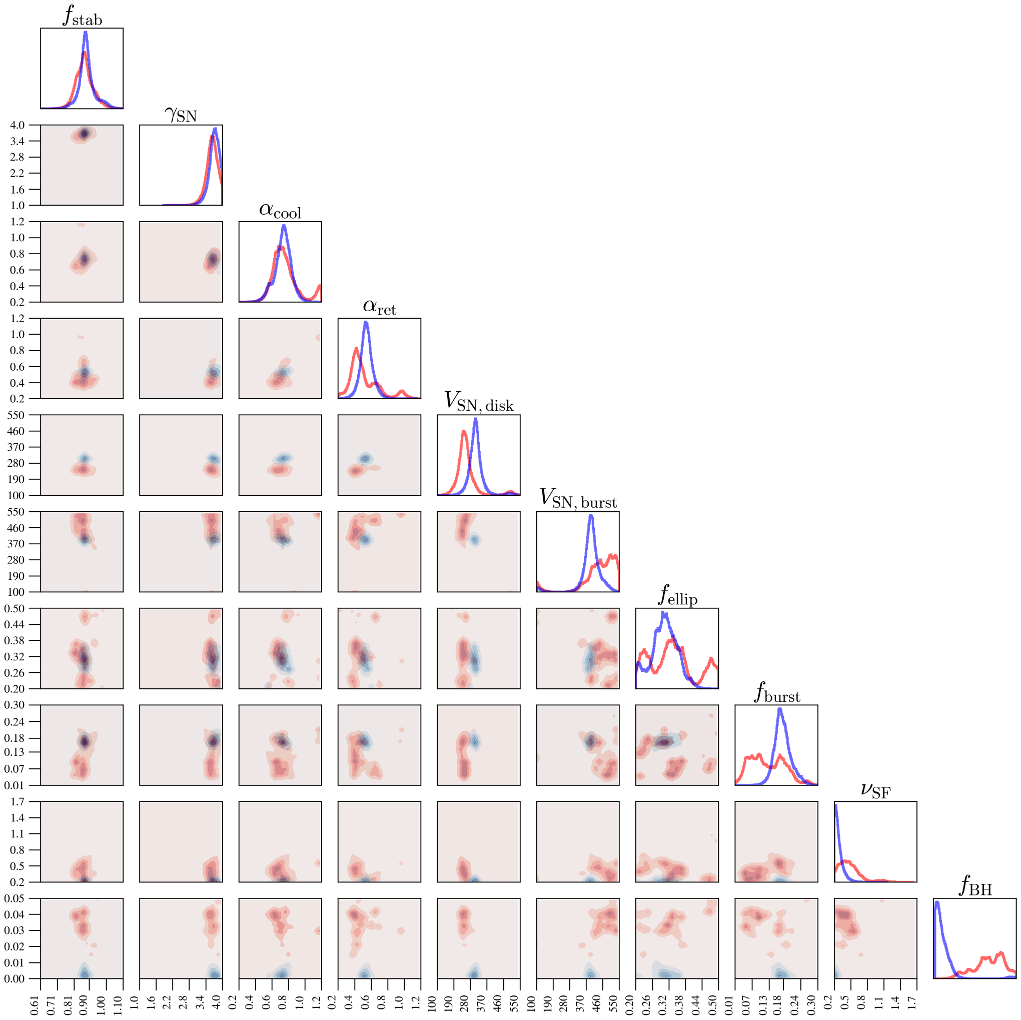


Figure A.3: Accepted samples from 20 MCMC chains for fits to the K-band LF (*red*), and both the K-band LF and the HI mass function (*blue*). The first 50% of samples were discarded to allow for burn-in. The histograms show the distribution of the parameters in 1D projection. The ranges on each axis are the same as those quoted in Table 4.1. The shading gives a sense of the density, 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.

Bibliography

Abadi M., et al., 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, https://www.tensorflow.org/

Aghamousa A., et al., 2016, The DESI Experiment Part I: Science,Targeting, and Survey Design (arXiv:1611.00036)

Alam S., et al., 2017, MNRAS, 470, 2617

Allen S. W., Evrard A. E., Mantz A. B., 2011, ARA&A, 49, 409

Alongi M., Bertelli G., Bressan A., Chiosi C., Fagotto F., Greggio L., Nasi E., 1993, A&AS, 97, 851

Baron D., 2019, Machine Learning in Astronomy: a practical overview (arXiv:1904.07248)

Bastian N., Covey K. R., Meyer M. R., 2010, ARA&A, 48, 339

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

Baugh C. M., Cole S., Frenk C. S., 1996, MNRAS, 283, 1361

Baugh C. M., Lacey C. G., Frenk C. S., Granato G. L., Silva L., Bressan A., Benson A. J., Cole S., 2005, MNRAS, 356, 1191

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

Benson A. J., 2010, Physics Reports, 495, 33

Benson A., 2018, Research Notes of the AAS, 2, 188

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

Benson A., Bower R., Frenk C., Lacey C., Baugh C., Cole S., 2003, The Astrophysical Journal, 599, 38

Bigiel F., et al., 2011, Astrophysical Journal Letters, 730, 1

Bishop C. M., 1997, Neural Networks, pp 1–44

Bishop C. M., 2007, Pattern Recognition and Machine Learning (Information Science and Statistics), 1 edn. Springer

Blain A. W., Chapman S. C., Smail I., Ivison R., 2004, ApJ, 611, 725

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

Bond J. R., Efstathiou G., 1984, ApJ, 285, L45

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

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

Bressan A., Fagotto F., Bertelli G., Chiosi C., 1993, A&AS, 100, 647

Burgarella D., et al., 2013, Astronomy and Astrophysics, 554, 1

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

Casey C. M., Narayanan D., Cooray A., 2014, Phys. Rep., 541, 45

Chabrier G., 2003, Publications of the Astronomical Society of the Pacific, 115, 763

Chapman S. C., Blain A. W., Smail I., Ivison R. J., 2005, The Astrophysical Journal, 622, 772

Chen C. C., Cowie L. L., Barger A. J., Casey C. M., Lee N., Sanders D. B., Wang W. H., Williams J. P., 2013, Astrophysical Journal, 762, 1

Christodoulou D. M., Shlosman I., Tohline J. E., 1995, ApJ, p. 551

Cole S., 1991, ApJ, 367, 45

Cole S., Aragon-Salamanca A., Frenk C. S., Navarro J. F., Zepf S. E., 1994, Monthly Notices of the Royal Astronomical Society, 271, 781

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

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

Coles P., Lucchin F., 1995, Cosmology. The origin and evolution of cosmic structure

Conroy C., 2013, Annual Review of Astronomy and Astrophysics, 51, 393

Conroy C., van Dokkum P. G., 2012, ApJ, 760, 71

Cowley W. I., Lacey C. G., Baugh C. M., Cole S., Frenk C. S., Lagos C. D. P., 2019, Monthly Notices of the Royal Astronomical Society, 487, 3082

Crain R. A., et al., 2015, Monthly Notices of the Royal Astronomical Society, 450, 1937

Cranmer M. D., Xu R., Battaglia P., Ho S., 2019, arXiv e-prints, p. arXiv:1909.05862

Croton D. J., et al., 2006, Monthly Notices of the Royal Astronomical Society, 365, 11

Cucciati O., et al., 2012, Astronomy and Astrophysics, 539

Da Cunha E., et al., 2015, Astrophysical Journal, 806

Davé R., Anglés-Alcázar D., Narayanan D., Li Q., Rafieferantsoa M. H., Appleby S., 2019, MNRAS, 486, 2827

Davis M., Efstathiou G., Frenk C. S., White S. D. M., 1985, ApJ, 292, 371

De Oliveira R. A., Li Y., Villaescusa-Navarro F., Ho S., Spergel D. N., 2020, Fast and Accurate Non-Linear Predictions of Universes with Deep Learning (arXiv:2012.00240)

Draine B., 2003, Annual Review of Astronomy and Astrophysics, 41, 241

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

Dubčáková R., 2011, Genetic Programming and Evolvable Machines, 12, 173

Dudzeviciute U., et al., 2019, 36, 1

Dudzevičiūtė U., et al., 2020, MNRAS, 494, 3828

Dunne L., Eales S., Edmunds M., Ivison R., Alexander P., Clements D. L., 2000, MNRAS, 315, 115

Efstathiou G., Lake G., Negroponte J., 1982, MNRAS, pp 1069–1088

Eisenstein D. J., et al., 2005, ApJ, 633, 560

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

Emmert-Streib F., Yang Z., Feng H., Tripathi S., Dehmer M., 2020, Frontiers in Artificial Intelligence, 3, 1

Fagotto F., Bressan A., Bertelli G., Chiosi C., 1994, A&AS, 104, 365

Farr W. M., Mandel I., 2018, Science, 361, aat6506

Ferrara A., Bianchi S., Cimatti A., Giovanardi C., 1999, The Astrophysical Journal Supplement Series, 123, 437

Forbes J. C., Krumholz M. R., Speagle J. S., 2019, Monthly Notices of the Royal Astronomical Society, 487, 3581

Frazier P. I., 2018, arXiv e-prints, p. arXiv:1807.02811

Geach J. E., et al., 2017, Monthly Notices of the Royal Astronomical Society, 465, 1789

Glorot X., Bengio Y., 2010, Journal of Machine Learning Research, 9, 249

Goldstein M., Wooff D., 2007, Bayes linear statistics: theory and methods. Wiley series in probability and statistics, John Wiley, Chichester, England

González J. E., Lacey C. G., Baugh C. M., Frenk C. S., Benson A. J., 2009, Monthly Notices of the Royal Astronomical Society, 397, 1254

Granato G. L., Lacey C. G., Silva L., Bressan A., Baugh C. M., Cole S., Frenk C. S., 2000, ApJ, 542, 710

Griffin A. J., Lacey C. G., Gonzalez-Perez V., Lagos C. d. P., Baugh C. M., Fanidakis N., 2019, MNRAS, 487, 198

Gunawardhana M. L. P., et al., 2011, MNRAS, 415, 1647

Gunawardhana M. L., et al., 2013, Monthly Notices of the Royal Astronomical Society, 433, 2764

Hamann J., Wons J., 2022, J. Cosmology Astropart. Phys., 2022, 036

Häring N., Rix H.-W., 2004, The Astrophysical Journal, 604, L89

Hayward C. C., Kere D., Jonsson P., Narayanan D., Cox T. J., Hernquist L., 2011, Astrophysical Journal, 743

Hayward C. C., et al., 2021, Monthly Notices of the Royal Astronomical Society, 502, 2922

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

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

Henriques B. M. B., Yates R. M., Fu J., Guo Q., Kauffmann G., Srisawat C., Thomas P. A., White S. D. M., 2020, Monthly Notices of the Royal Astronomical Society, 491, 5795

Herman J., Usher W., 2017, Journal of Open Source Software, 2, 97

Hochreiter S., Schmidhuber J., 1997, Neural Computation, 9, 1735

Holmberg E., 1941, ApJ, 94, 385

Hou J., Lacey C. G., Frenk C. S., 2018, Monthly Notices of the Royal Astronomical Society, 475, 543

Hughes D. H., et al., 1998, Nature, 394, 241

Huško F., Lacey C. G., Baugh C. M., 2021, 20, 1

Jiang L., Helly J. C., Cole S., Frenk C. S., 2014, MNRAS, 440, 2115

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

Kauffmann G., 1996, MNRAS, 281, 487

Kennicutt R. C. J., 1983, ApJ, 272, 54

Kim S.-H., Martin P. G., Hendry P., 1993, in American Astronomical Society Meeting Abstracts #182. p. 08.13

Kingma D. P., Ba J. L., 2015, in 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings. pp 1–15 (arXiv:1412.6980)

Kochanek C. S., et al., 2001, ApJ, 20, 566

Lacey C., Silk J., 1991, ApJ, 381, 14

Lacey C. G., Baugh C. M., Frenk C. S., Silva L., Granato G. L., Bressan A., 2008, MNRAS, 385, 1155

Lacey C. G., et al., 2016, Monthly Notices of the Royal Astronomical Society, 462, 3854

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

Lagos C. D. P., Lacey C. G., Baugh C. M., 2013, Monthly Notices of the Royal Astronomical Society, 436, 1787

Lagos C. d. P., Tobar R. J., Robotham A. S., Obreschkow D., Mitchell P. D., Power C., Elahi P. J., 2018, Monthly Notices of the Royal Astronomical Society, 481, 3573

Lagos C. D. P., et al., 2019, Monthly Notices of the Royal Astronomical Society, 489, 4196

Lange R., et al., 2016, MNRAS, 462, 1470

Larson R. B., 1974, Monthly Notices of the Royal Astronomical Society, 169, 229

Leclercq F., 2018, Phys. Rev. D, 98, 063511

Lejeune T., Cuisinier F., Buser R., 1998, A&AS, 130, 65

Lovell C. C., Geach J. E., Davé R., Narayanan D., Li Q., 2021, Monthly Notices of the Royal Astronomical Society, 502, 772

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

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

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

Maclin R., Opitz D., 2011, arXiv e-prints, p. arXiv:1106.0257

Maraston C., 2005, MNRAS, 362, 799

Martin A. M., Papastergis E., Giovanelli R., Haynes M. P., Springob C. M., Stierwalt S., 2010, Astrophysical Journal, 723, 1359

Martindale H., Thomas P. A., Henriques B. M., Loveday J., 2017, Monthly Notices of the Royal Astronomical Society, 472, 1981

Mathewson D. S., Ford V. L., Buchhorn M., 1992, The Astrophysical Journal Supplement Series, 81, 413

McAlpine S., et al., 2019, Monthly Notices of the Royal Astronomical Society, 488, 2440

Moffett A. J., et al., 2016, Monthly Notices of the Royal Astronomical Society, 457, 1308

Neal R. M., 1994, Journal of the American Statistical Association, 92, 791

Nelson D., et al., 2015, Astronomy and Computing, 13, 12

Ntampaka M., et al., 2019, BAAS, 51, 14

Oesch P. A., et al., 2013, Astrophysical Journal, 773

Oleśkiewicz P., Baugh C. M., 2020, MNRAS, 493, 1827

Patacchiola M., Turner J., Crowley E. J., O'Boyle M., Storkey A., 2020, Bayesian Meta-Learning for the Few-Shot Setting via Deep Kernels (arXiv:1910.05199)

Peebles P. J. E., 1980, The large-scale structure of the universe

Peebles P. J. E., 1982, ApJ, 263, L1

Percival W. J., et al., 2001, MNRAS, 327, 1297

Perlmutter S., et al., 1999, ApJ, 517, 565

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

Pillepich A., et al., 2017, Monthly Notices of the Royal Astronomical Society, 473, 4077–4106

Pillepich A., et al., 2018, Monthly Notices of the Royal Astronomical Society, 473, 4077

Planck Collaboration et al., 2016, A&A, 594, A13

Rasmussen C. E., Williams C. K. I., 2006, Gaussian processes for machine learning.. Adaptive computation and machine learning, MIT Press

Ravanbakhsh S., Oliva J., Fromenteau S., Price L. C., Ho S., Schneider J., Póczos B., 2016, 33rd International Conference on Machine Learning, ICML 2016, 5, 3584

Reddi S. J., Kale S., Kumar S., 2018, in ICLR. pp 1–23

Riess A. G., et al., 1998, AJ, 116, 1009

Robert C. P., 2015, arXiv e-prints, p. arXiv:1504.01896

Robert C. P., 2016, The Metropolis-Hastings algorithm (arXiv:1504.01896)

Robitaille T. P., 2011, A&A, 536, A79

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

Rogers K. K., Peiris H. V., Pontzen A., Bird S., Verde L., Font-Ribera A., 2019, J. Cosmology Astropart. Phys., 2019, 031

Romano D., Matteucci F., Zhang Z. Y., Papadopoulos P. P., Ivison R. J., 2017, MNRAS, 470, 401

Rudy S., Alla A., Brunton S. L., Kutz J. N., 2019, SIAM Journal on Applied Dynamical Systems, 18, 643

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

Safarzadeh M., Lu Y., Hayward C. C., 2017, Monthly Notices of the Royal Astronomical Society, 472, 2462

Saltelli A., 2017, Journal of Chemical Information and Modeling, 53, 1689

Saltelli A., Annoni P., Azzini I., Campolongo F., Ratto M., Tarantola S., 2010, Computer Physics Communications, 181, 259

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

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

Schneider F. R. N., et al., 2018, Science, 359, 69

Shen S., Mo H. J., White S. D., Blanton M. R., Kauffmann G., Voges W., Brinkmann J., Csabai I., 2003, Monthly Notices of the Royal Astronomical Society, 343, 978

Silva L., Granato G. L., Bressan A., Danese L., 1998, ApJ, 509, 103

Simha V., Cole S., 2017, Monthly Notices of the Royal Astronomical Society, 472, 1392

Smail I., Ivison R. J., Blain A. W., 1997, The Astrophysical Journal, 490, L5

Smith R. J., 2020, ARA&A, 58, 577

Smith R. J., Lucey J. R., Hudson M. J., 2009, Monthly Notices of the Royal Astronomical Society, 400, 1690

Smith D. J., Hayward C. C., Jarvis M. J., Simpson C., 2017, Monthly Notices of the Royal Astronomical Society, 471, 2453

Sobral D., Best P. N., Smail I., Mobasher B., Stott J., Nisbet D., 2013, Monthly Notices of the Royal Astronomical Society, 437, 3516

Somerville R. S., Davé R., 2015, Annual Review of Astronomy and Astrophysics, 53, 51

Somerville R. S., Gilmore R. C., Primack J. R., Domínguez A., 2012, MNRAS, 423, 1992

Spergel D. N., et al., 2003, ApJS, 148, 175

Springel V., 2010, MNRAS, 401, 791

Springel V., White S. D. M., Tormen G., Kauffmann G., 2001, MNRAS, 328, 726

Springel V., Frenk C. S., White S. D. M., 2006, Nature, 440, 1137

Stach S. M., et al., 2018, ApJ, 860, 161

Stein M., 1987, Technometrics, 29, 143

Swinbank A. M., Smail I., Chapman S. C., Blain A. W., Ivison R. J., Keel W. C., 2004, The Astrophysical Journal, 617, 64

Takhtaganov T., Lukić Z., Müller J., Morozov D., 2021, ApJ, 906, 74

Tieleman T., Hinton G., 2016, COURSERA: Neural Networks for Machine Learning, 4, 1

Vernon I., Goldstein M., Bower R. G., 2010, Bayesian analysis., 05, 619

Vernon I., Goldstein M., Bower R., 2014, Statistical Science, 29, 81

Vlahakis C., Dunne L., Eales S., 2005, MNRAS, 364, 1253

Vogelsberger M., et al., 2014, MNRAS, 444, 1518

Wardlow J. L., et al., 2011, MNRAS, 415, 1479

Weidner C., Ferreras I., Vazdekis A., La Barbera F., 2013, MNRAS, 435, 2274

Weinberger R., et al., 2016, Monthly Notices of the Royal Astronomical Society, 465, 3291–3308

Westera, P. Lejeune, T. Buser, R. Cuisinier, F. Bruzual, G. 2002, A&A, 381, 524

White S. D. M., Frenk C. S., 1991, ApJ, 379, 52

White S. D., Rees M. J., 1978, Monthly Notices of the Royal Astronomical Society, pp 341–358

Wilson A. G., Hu Z., Salakhutdinov R., Xing E. P., 2016, Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, AISTATS 2016, pp 370–378

Yip J. H. T., et al., 2019, arXiv e-prints, p. arXiv:1910.07813

York D. G., et al., 2000a, AJ, 120, 1579

York D. G., et al., 2000b, The Astronomical Journal, 120, 1579

Zwaan M. A., Meyer M. J., Staveley-Smith L., Webster R. L., 2005, Monthly Notices of the Royal Astronomical Society: Letters, 359, 1

de Jong R. S., Lacey C., 2000, The Astrophysical Journal, 545, 781

van der Velden E., Duffy A. R., Croton D., Mutch S. J., 2021, ApJS, 253, 50