Mid-Course Assessment

Hierarchically Modelling Stars Using Deep Learning and Asteroseismology

By

Alexander J. Lyttle

Student ID 1532473
Supervisor Dr Guy R. Davies
Co-Supervisor Dr Andrea Miglio

Solar and Stellar Physics Group
School of Physics and Astronomy
College of Engineering and Physical Sciences
University of Birmingham

September 2, 2020

ABSTRACT

Your abstract goes here

ACKNOWLEDGMENTS

I acknowledge the people who helped me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Astroseismology

1.1.1 Solar-like oscillators

1.1.2 Detecting oscillation modes

1.1.3 Helium II Ionization Zone Glitch

1.2 Machine Learning in Astrophysics

1.3 Hierarchical Bayesian Models in Astrophysics

1.4 Open-Source Code in Astrophysics

Chapter 2

Peakbagging with PBjam

2.1 How it Works

2.2 Contributing to the Code

2.3 Peakbagging with TESS

Chapter 3

Hierarchically Modelling Many Stars

See the accompanying paper (Appendix A).

Chapter 4

Future Work

4.1 Including the Helium II Glitch

4.2 Increasing the Sample Size

4.3 To Higher Mass Stars and Beyond

Our next step is to include intermediate-mass stars with masses from approx. 1.2 solar masses to 3.0 solar masses.

Appendix A

Accompanying Paper

MNRAS **000**, 1–8 (2020) Preprint 1 September 2020 Compiled using MNRAS LATEX style file v3.0

TBC: Hierarchically modelling *Kepler* dwarfs using machine learning to uncover helium enrichment in the solar neighbourhood

Alexander J. Lyttle,[1,2]★ Tanda Li,[1,2] Guy R. Davies,[1,2] Lindsey M. Carboneau[1,2] and TBC
[1]*School of Physics and Astronomy, University of Birmingham, Birmingham, B15 2TT, UK*
[2]*Stellar Astrophysics Centre (SAC), Department of Physics and Astronomy, Aarhus University, Ny Munkegade 120, DK-8000 Aarhus C, Denmark*

Accepted XXX. Received YYY; in original form ZZZ

ABSTRACT

Key words:
asteroseismology – methods: miscellaneous – methods: statistical – stars: fundamental parameters – stars: low-mass

1 INTRODUCTION

Motivation - precise and accurate stellar fundamentals. Useful for e.g. galactic archaeology and exoplanet research.

Audience - astrophysicist with some knowledge Introduce new method and reference Guy's paper:

- Summarise the typical way in which stellar fundamentals are estimated and their pitfalls (e.g. discrete sampling, and assuming solar calibrated mixing-length parameter and helium enrichment)
  - Problems with grid-based-modelling (e.g. proper sampling)
  - assuming fixed DYDZ and MLT bad; attempts to interpolate, slow and hard to scale
- Why hierarchical models are good with examples of HBMs in astrophysics
- Advantage of HBM is to incorporate population-level distributions
- Why HBMs are difficult with stellar models.
- Introduce the neural network as a way to overcome these issues and give examples of neural networks to approximate models in astrophysics
- Highlight the novel element of this paper - the first application of combining a neural network emulator with a hierarchical model to provide shrinkage of fundamentals uncertainties and simultaneously study a helium enrichment relation
- Use a helium enrichment law prior, and assume a distribution of mixing-length of the population-level, to inform object-level parameters

Why do we care about helium and mixing-length? These parameters have a large (be quantitative) affect on stellar ages. Good stellar ages allow us to better study galactic archaeology (with citations).

Given that we are assuming a helium enrichment prior, give a brief summary of research into the helium enrichment and typical values for $\Delta Y/\Delta Z$. Note that in reality there may not be a linear law, and more may be studied in future work (or using a GP like in Guy's paper?). Why do we care about an enrichment law? Why is it physically justified?

Given that we are assuming a mixing-length distribution, mention this is mainly a nuisance parameter which we will marginalize over, since this differs depending on model physics. However, later justify a normal spread by referring to work (e.g. Magic) which shows little variation in the area of the HRD we are studying.

Outline the structure of the work. We are demonstrating the method on an asteroseismic sample of dwarfs and subgiants from Serenelli 2017. We first introduce the data and why we choose to use spectroscopy and asteroseismology. We then introduce the method, from the grid of stellar models

Why asteroseismology and why this particular set of Kepler-field dwarfs? Acknowledge selection bias but explain that with TESS providing an all-sky sample of solar-like oscillators this method can be extended to a much larger sample size.

Note: here is an example of a paper which would benefit from a value of the intrinsic spread in helium enrichment: (Zinn et al. 2019) 'Until such a time as the intrinsic scatter in helium enrichment can be determined, which... hinders a comparison between the theoretical metallicity trend and the observed radius agreement... the asteroseismic scaling relation radius does not require a metallicity term...'. In other words, they assume a helium enrichment law but this hinders their ability to study the seismic scaling relation correction.

2 DATA

We began with the sample of 415 stars from the first APOKASC catalogue of dwarfs and subgiants (S17). It is, to date, the most comprehensive sample of asteroseismic dwarfs and subgiant stars observed by the *Kepler* mission. We adopted the global asteroseismic parameters – the large frequency separation, $\Delta\nu$, and the

★ E-mail: ajl573@student.bham.ac.uk

frequency of maximum power, $\nu_{\rm max}$ – determined by S17, and references therein. We then cross matched the sample with *Gaia* Data Release 2 for high-precision parallaxes and the Apache Point Observatory Galaxy Evolution Experiment (APOGEE) catalogue to obtain spectroscopic metallicities and effective temperatures. Using the cross matched catalogue, we calculated luminosities for the full sample with Two-Micron All Sky Survey (2MASS) photometry and selected a subsample of stars with similar metallicities and masses. Our method is described in detail in the remainder of this section.

We cross matched the *Kepler* input catalogue (KIC) for the sample with the *Gaia* DR2 catalogue taking the nearest neighbours within a 4" radius [CITE GAIA]. All but two of the sample of 415 stars were available. We then adopted the *Gaia* parallaxes, assuming a zero-point offset of 0.05 mas in the sense that the *Gaia* parallaxes are underestimated. We chose this value in line with recent studies on the *Gaia* zero-point parallax offset in the *Kepler* field [CITATIONS].

We adopted spectroscopic metallicities, [M/H], and effective temperatures, $T_{\rm eff}$determined by the APOGEE stellar parameters and chemical abundances pipeline (ASPCAP) from the second data release of the fourth phase of the Sloan Digital Sky Survey (SDSS) otherwise known as Data Release 14 (DR14). We cross matched the APOGEE DR14 catalogue with our *Kepler-Gaia* DR2 cross match yielding spectroscopic parameters for all 413 stars in the sample.

In order to remove more evolved stars, we made a cut in surface gravity, g. We used asteroseismic $\nu_{\rm max}$ with ASPCAP $T_{\rm eff}$ and rearranged the asteroseismic scaling relation to get,

$$\log g \approx \log g_{\odot} + \log\left(\frac{\nu_{\rm max}}{\nu_{\rm max,\odot}}\right) - \frac{1}{2}\log\left(\frac{T_{\rm eff}}{T_{\rm eff,\odot}}\right), \quad (1)$$

where solar reference values of $\nu_{\rm max,\odot} = 3090 \pm 30\,\mu$Hz (Huber et al. 2011) $\log g_{\odot} = 4.44$ dex and $T_{\rm eff,\odot} = 5777$ K were used to determine the log surface gravity, $\log g$.

We determined luminosities for the sample using the direct method of ISOCLASSIFY [CITE HUBER]. We calculated absolute K_S-band magnitudes using K_s-band photometry from the 2MASS, distances from the zero-point-offset-corrected parallaxes from *Gaia* DR2 and extinctions determined from the 3D galactic reddening maps of Green et al. (2019) [CITE]. We then determined absolute bolometric magnitudes by interpolating the MIST bolometric correction tables using ASPCAP [M/H]and $T_{\rm eff}$, asteroseismic $\log g$ and absolute magnitude as inputs. An uncertainty of 0.02 mag was assumed for both the extinctions and bolometric corrections, in line with typical uncertainties from randomly sampling the input data within their errors. The resulting distances, absolute magnitudes and luminosities with their respective uncertainties are given in Table X.

We selected a subset from the above sample which we determined to lie within the bounds of the model grid described in Section 3.1 using mass estimates from S17. We determined such "on-grid" stars where their estimated mass and metallicity were within one standard deviation of the grid boundary, from 0.8 to 1.2 $M_{\odot}$ in mass and from −0.5 to 0.5 dex in metallicity. We also cut targets in the sample with an asteroseimic $\log g$ less than 3.8 dex to remove more evolved stars. The cut in mass was motivated by our choice of model physics described in Section 3.1. Stars with $M \gtrsim 1.15\,M_{\odot}$ are understood to have a convective, hydrogen-burning core, with some dependence on the choice of stellar physics [CITE Appourchaux](). Modelling stars with a convective core requires the consideration of extra mixing due to the overshooting of convective cells at the core boundary [CITE OVERSHOOT PAPERS], which is beyond the scope of this work.

The final sample comprised 81 stars, after removing stars with null observables. The data for 10 stars from the sample is shown in Table 1 and the full table may downloaded LINK. The Hertzsprung-Rusell diagram in Figure **??** shows the sample plot above a selection of stellar evolutionary tracks from the grid described in Section 3.1. A second plot shows the sample in context with a selection of *Kepler* solar neighbourhood stars. Consider the range of parallaxes plot when defininig the solar neighbourhood (less that 1 kpc).

3 METHODS

Outline the model and its requirements with justification before introducing the following subsections.

3.1 Grid of stellar models

We built up a stellar model grid to train the NN model. The gird includes four independent model inputs: stellar mass (M), initial helium fraction ($Y_{\rm init}$), initial metallicity ([Fe/H]), and the mixing-length parameter ($\alpha_{\rm MLT}$). Ranges and grid steps of the four model inputs are summarised in Table 2. We computed each stellar evolutionary track from the Hayashi line and to the base of red-giant branch where $\log g = 3.6$ dex. We also computed 4,000 evolutionary tracks with random input values in the parameter space for validating the results.

3.1.1 *Stellar models and input physics*

We used Modules for Experiments in Stellar Astrophysics (MESA, version 12115) to establish a grid of stellar models. MESA is an open-source stellar evolution package which is undergoing active development. Descriptions of input physics and numerical methods can be found in Paxton et al. (2011, 2013, 2015). We adopted the solar chemical mixture $[(Z/X)_{\odot} = 0.0181]$ provided by Asplund et al. (2009). The initial chemical composition was calculated by:

$$\log(Z_{\rm init}/X_{\rm init}) = \log(Z/X)_{\odot} + [{\rm Fe/H}]. \quad (2)$$

We used the MESA $\rho - T$ tables based on the 2005 update of OPAL EOS tables (Rogers & Nayfonov 2002) and OPAL opacity supplemented by low-temperature opacity (Ferguson et al. 2005). The MESA 'simple' photosphere were used as the set of boundary conditions for modelling the atmosphere. The mixing-length theory of convection was implemented, where $\alpha_{\rm MLT} = \ell_{\rm MLT}/H_p$ is the mixing-length parameter. We also applied the MESA predictive mixing scheme (Paxton et al. 2018, 2019) in the model computation.

The evolution time step was mainly controlled by the set-up tolerances on changes in surface effective temperature and luminosity. We saved one structural model at every time step at main sequence and every two steps after central hydrogen exhaustion. For each evolutionary track, we obtained ∼100 at the main-sequence stage and 500 − 700 at evolved stages.

3.1.2 *Oscillation models and seismic* $\Delta\nu$

Theoretical stellar oscillations were calculated with the GYRE code (version 5.1), which was developed by Townsend & Teitler (2013). And we computed radial modes (for $\ell = 0$) by solving the adiabatic stellar pulsation equations with the structural models generated by

Table 1.

Name	$T_{\rm eff}$ (K)	$\sigma_{T_{\rm eff}}$ (K)	L ($L_\odot$)	σ_L ($L_\odot$)	$\Delta\nu$ (μHz)	$\sigma_{\Delta\nu}$ (μHz)	[M/H] (dex)	$\sigma_{\rm [M/H]}$ (dex)	log g (dex)	$\sigma_{\log g}$ (dex)
KIC10079226	5928.84	124.84	1.57	0.05	116.04	0.73	0.16	0.07	4.36	0.01
KIC10215584	5666.92	119.33	1.64	0.06	115.16	2.83	0.04	0.07	4.27	0.09
KIC10319352	5456.17	106.65	1.85	0.06	78.75	1.73	0.27	0.06	3.96	0.13
KIC10322381	6146.79	148.58	2.44	0.08	86.64	6.57	-0.32	0.08	4.19	0.04
KIC10417911	5628.26	109.99	3.41	0.12	56.14	2.10	0.34	0.07	3.94	0.02
KIC10732098	5669.65	119.28	3.02	0.12	62.18	1.92	0.05	0.07	3.96	0.02
KIC10794845	6035.12	140.46	1.64	0.06	116.35	6.70	-0.21	0.08	4.40	0.11
KIC10963065	6039.78	139.10	1.88	0.06	103.21	0.11	-0.16	0.08	4.30	0.01
KIC10971974	5748.00	142.40	1.43	0.05	106.63	3.31	-0.07	0.09	4.32	0.04
KIC11021413	5329.18	102.98	3.16	0.11	48.16	1.29	0.01	0.04	3.84	0.01

Table 2. Stellar model computations for training and test datasets.

Training model set (Grid-based)			
Input Parameter	Range	Increment	$N_{\rm track}$
M [$M_\odot$]	0.80 – 1.20	0.01	
[Fe/H] [dex]	-0.5 – 0.2/0.2 – 0.5	0.1/0.05	15,375
$Y_{\rm init}$	0.24 – 0.32	0.02	
$\alpha_{\rm MLT}$	1.7 – 2.5	0.2	

MESA. We computed a seismic large separation($\Delta\nu$) for each model with theoretical radial modes to avoid the systematic offset of the scaling relation. We derived $\Delta\nu$ with the approach given by White et al. (2011), which is a weighted least-squares fit to the radial frequencies as a function of n.

3.2 Artificial neural network

Artificial neural network (ANN). Introduce, with reference, the principle behind a neural network and the way in which it can be evaluated. Used Tensorflow etc. The follwoing will work much better in a diagram.

- Initial mass, M
- Initial helium fraction, $Y_{\rm init}$
- Initial metals fraction, $Z_{\rm init}$
- Mixing-length-theory parameter, $\alpha_{\rm mlt}$
- Fractional main-sequence lifetime, $f_{\tau_{\rm MS}}$

The training dataset comprised the inputs and outputs of the grid of stellar models described in Section 3.1. We chose the ANN inputs to correspond to the following model fundamentals. . . with the exception of age, which serves best as an output for the following reason. Age is understood to scale with at least mass and fractional main-sequence lifetime (citation). We found that the ANN performed better with a proxy for age as input, and age as an output.

We chose the ANN outputs to correspond to, or allow derivation of, observables,

- Effective temperature, $T_{\rm eff}$
- Radius, R
- Large frequency separation, $\Delta\nu$
- Surface metallicity, $[{\rm M/H}]_{\rm surf}$
- Age, τ

We trained an artificial neural network (ANN) on the data generated by the grid of stellar models to map stellar fundamentals to observables. We split the grid of stellar models into a *train* and *test* dataset for tuning the ANN, as described in Section 3.2.1. We tested a multitude of ANN configurations and training data augmentations, evaluating them with the validation set in Section 3.2.2. We reserved a randomly generated set of stellar models as our final *validation* dataset, to evaluate the approximation ability of the ANN. In Section 3.2.3, we trained the optimal ANN on the combined training and test dataset and evaluated it on the validation dataset. Firstly, however, in this section we briefly describe the theory and motivation behind the ANN.

The ANN is a network of so-called *neurons* which each transform some input vector, $\boldsymbol{x}$ based on trainable *weights*, $\boldsymbol{w}$ and a *bias*, b (see CITATIONS). Deep learning (DL) is the case where neurons are arranged into a series of layers such that any neuron in layer $k-1$ is connected to at least one of the neurons in layer k. For this work, we considered a fully-connected ANN, where each neuron in layer $k-1$ is connected to every neuron in layer k. The weights are represented by the connections between neurons and the bias is a unique scalar associated with each neuron. In general, the k-th layer comprises N_k neurons which combined, output the vector $\boldsymbol{x}^k = \{x_i^k\}_{i=1}^{N_k}$. For example, the i-th neuron in layer k transforms the vector of outputs received from the previous layer, $k-1$, such that its output is,

$$x_i^k = A^k\left(o_i^k\right), \tag{3}$$

$$o_i^k = b_i^k + \sum_{j=1}^{N_{k-1}} w_{i,j}^{k-1} x_j^{k-1}, \tag{4}$$

where A^k is the *activation* function for the k-th layer, $\boldsymbol{w}_i^{k-1}$ are the weights connecting all the neurons in layer $k-1$ to the current neuron, and b_i^k is its bias.

We refer to the overall arrangement of neurons in the ANN as the *architecture*. The architecture is arranged such that an initial *input* layer outputs vector $\boldsymbol{x}^{\rm in}$ to the first of M so-called *hidden* layers. We considered ANNs with the same number of neurons, N, in each hidden layer. The inputs are propagated through the hidden layers, $k = 1, 2, \ldots, M$, using Equation 3 with some chosen activation function. The output of the final hidden layer, $\boldsymbol{x}^M$ is passed to the *output* layer which yields the vector $\boldsymbol{x}^{\rm out}$, corresponding to the data we wish to predict. In the case of ANN regression, the activation

function for the output layer is linear, i.e. $A^{\rm out}(o_i^{\rm out}) = o_i^{\rm out}$. One forward pass through the neural network makes predictions given some set of inputs,

$$\boldsymbol{x}^{\rm out} = \eta_{\rm net}(\boldsymbol{x}^{\rm in}). \quad (5)$$

To fit the ANN, we used a set of training data, $\boldsymbol{X} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1) \ldots (\boldsymbol{x}_{N_{\rm train}}, \boldsymbol{y}_{N_{\rm train}})\}$ comprising $N_{\rm train}$ input-output pairs. We split the training data into random batches, $\boldsymbol{X}_{\rm batch}$ because this has been shown to improve model convergence (CITE) and the computational efficiency. An error function, $E(\boldsymbol{X}_{\rm batch})$, also known as the *loss*, quantifies the difference between the predictions and the training data. The weights are updated for each batch using an algorithm called the *optimizer*. We also considered an addition to the loss called *regularisation* which helps reduce over-fitting (CITE). We initialised each ANN with a random set of weights and biases and minimized the loss over a given number of *epochs*. One epoch is one iteration through the training dataset, $\boldsymbol{X}$. We tracked the loss for each ANN using an independent test dataset to choose the most effective architecture, batch size, regularisation, loss function and optimizer in Section 3.2.2.

3.2.1 *Train-test-validation split*

Firstly, consider re-sampling the grid data weighted by position on a Kiel diagram (logg and teff) so that we have good coverage of observables. This will help the network train better.

Then explain normalization of the data.

Describe the train-test split as a way of optimizing the neural network architecture without being biased towards the training data. I.e. we chose parameters based on what minimises the difference between train and test loss.We reserved 20 per cent of the on-grid dataset for validation. This allowed us to asses our choice of of ANN hyperparameters without being biased towards the specific set of training data.

Some histograms showing the training and test data may be useful.

3.2.2 *Optimization*

Some description of how we went about choosing the network architecture and what we eventually settled on. A diagram showing the network architecture with explanation.

THIS IS A ROUGH DRAFT

A generic form of the ANN architecture is shown in Figure 7. We varied the number of neurons, N per hidden layer and the total number of hidden layers, M and the choice of activation function. We also considered the addition of regularisation of the weights, which act to add random scatter to the weights to help dislodge weights stuck in local minima. The impact of these parameters were assessed by looking at the behaviour of the train and test loss with respect to the number of epochs and wall time. The closer the test loss was to the training loss, the better.

The choice of batch size (i.e. the number of forward passes before updating the weights) was also found to impact the training. A smaller batch size helped the ANN converge faster and was less computationally expensive. The final batch size was chosen as a trade-off between GPU efficiency and convergence speed.

The final choice of architecture was $N = 128$ neurons for each of $M = 6$ hidden layers. We found that the exponential linear unit (ELU) was the most appropriate choice of activation function over the more popular rectified linear unit (ReLU), although ReLU would converge faster. ELU suited the smooth nature of the stellar evolutionary tracks better, whereas ReLU contains a gradient discontinuity which caused poorer sampling later-on during modelling.

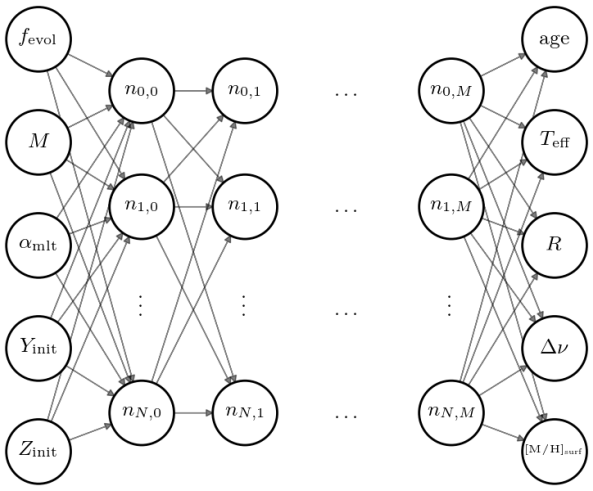

Figure 1. A diagram depicting the artificial neural network architecture. [MAYBE EXPLAIN IN TEXT] Each node represents a neuron and each line represents the weight, w, which connects the neurons. The direction of the arrows represents a forward-propagation through the network. The inputs (left-most layer) and outputs (right-most layer) are provided by the training set. The hidden neurons, labelled $n_{i,j}$ where $i = 0, 1, \ldots, N$ and $j = 0, 1, \ldots, M$, take some input x and transform it $wx + b$ where b is a bias. The output is then transformed given some activation function.

The neural network was then trained on the full train-test dataset for a total of 24 hours before being evaluated using an independent validation dataset described in Section **??**. The final training loss was a mean absolute error of $XX.XX$.

3.2.3 *Testing*

Here will be a summary of the neural network accuracy for the train, test and validation datasets.

3.3 Hierarchical bayesian model

We devised three Bayesian models, each with varying levels of parameter sharing between stars in the sample. Initially, we tested the models and demonstrated shrinkage of statistical uncertainties in the stellar fundamental parameters by analysing a random sample of 100 stars modelled using MESA. Then, we applied the models to the sample of stars collated in Section **??** and compared the results with that of S17.

The first model is equivalent to modelling each star individually and features no parameter sharing; as such, we refer to it as the no-pooled (NP) model. We then introduce two hierarchical Bayesian models (HBMs) which employ population-level parameters to describe their distribution in the sample. Both models partially-pool helium via a linear enrichment law. In other words, the initial helium fraction for each star is drawn from a normal distribution, with a mean described by the enrichment law and standard deviation to describe the deviation of helium from said law. One model also partially-pools the mixing-length theory parameter, $\alpha_{\rm mlt}$ in a similar way, whereas the other maximally-pools $\alpha_{\rm mlt}$ such that is assumes

the same value for the entire sample. We refer to the former as the max-pooled (MP) model and the latter as the partial-pooled (PP) model. All three models are described in the following subsections.

We sampled the posterior for each model using the No U-Turn Sampler (NUTS) of PyMC4 [CITE]. Initially, we modelled each star individually in order to identify stars outside the grid range and other sampling problems. We flagged stars with median modelled values outside the grid range by more than the median 16th or 84th percentile in the sample.

3.3.1 *No-pooled model*

Firstly, we constructed a model comprising independent parameters $\boldsymbol{\theta}_i = \{f_{\rm evol,}i, M_i, \alpha_{\rm mlt,}i, Y_i, Z_i\}$ for a given star, i. Using Bayes' theorem, the *posterior* probability density function (PDF) of the model parameters given a set of observed data, $\boldsymbol{y}_i$ is,

$$p(\boldsymbol{\theta}_i|\boldsymbol{y}_i) \propto p(\boldsymbol{\theta}_i)p(\boldsymbol{y}_i|\boldsymbol{\theta}_i), \tag{6}$$

where $p(\boldsymbol{\theta}_i)$ is the *prior* PDF of the model parameters and $p(\boldsymbol{y}_i|\boldsymbol{\theta}_i)$ is the *likelihood* of observing the data given the model.

We chose weakly-informative, bounded priors for the independent parameters, restricting them to their respective ranges in the ANN training data. Although the neural network is able to make predictions outside the training data range, these have not been tested and may be unreliable. Therefore, we used a beta distribution with $\alpha = \beta = 1.2$ as the prior PDF on the independent parameters, transformed such that the probability is null outside the chosen range,

$$p(\boldsymbol{\theta}_i) = \prod_{k=1}^{N_\theta} \left[\theta_{k,\rm min} + (\theta_{k,\rm max} - \theta_{k,\rm min})\mathcal{B}(\theta_{k,i}|1.2, 1.2)\right], \tag{7}$$

where the beta distribution is defined as,

$$\mathcal{B}(x|\alpha,\beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\int_0^1 u^{\alpha-1}(1-u)^{\beta-1}\mathrm{d}u}. \tag{8}$$

The beta distribution was preferred over a bounded uniform distribution because our sampler evaluates the gradient of the posterior and hence sensitive to discontinuities.

We made predictions for each star using the trained ANN, $\{\log(\rm age)_i, T_{\rm eff,}i, R_i, \Delta\nu_i, [\rm M/H]_{\rm surf,}i\} = \eta_{\rm net}(\boldsymbol{\theta}_i)$, from which we derived the luminosity, L_i using the Stefan-Boltzmann law. Any of the model parameters may be passed as an observable. Hereafter, we denote the set of model observables as $\boldsymbol{\mu}_{y,i} = \boldsymbol{\mu}_y(\boldsymbol{\theta}_i)$. Thus, we write the likelihood we observe any $\boldsymbol{y}_i$ with uncertainty, $\boldsymbol{\sigma}_y$ given the model as,

$$p(\boldsymbol{y}_i|\boldsymbol{\theta}_i) = \prod_{k=1}^{N_{\rm obs}} \frac{1}{\sigma_{y,k,i}\sqrt{2\pi}} \exp\left[-\frac{(y_{k,i} - \mu_{y,k}(\boldsymbol{\theta}_i))^2}{2\sigma^2_{y,k,i}}\right], \tag{9}$$

where $N_{\rm obs}$ is the number of observed variables. We chose to use observed $T_{\rm eff}$, L, $\Delta\nu$ and [M/H] collated for our sample as described in Section **??**.

Using the above model, we sampled from the posterior for each individual star separately and then together as a population of $N_{\rm stars}$ stars, $p(\boldsymbol{\theta}|\boldsymbol{y}) = \prod_{i=1}^{N_{\rm stars}} p(\boldsymbol{\theta}_i|\boldsymbol{y}_i)$. Separate modelling allowed us to identify poorly sampled posteriors, whether the model indicated a fit outside the given input range, or other sampling issues. Once a refined sample was chosen, modelling the sample all together was done as a natural application of the ANN through the use of batching. The ANN inputs were modelled as independent distributions, then the random variables were batched together and passed through the ANN to produce predictions for each star.

3.3.2 *Partial-pooled model*

Sharing, or pooling parameters between stars in a population can improve the uncertainties on stellar fundamentals by encoding our prior knowledge of their distribution in a population. We constructed a hierarchical model [CITE Gelman?], which builds upon the NP model by introducing population-level *hyperparameters*. Specifically, we chose to describe initial helium and $\alpha_{\rm mlt}$ by partially-pooling them.

We constructed the PP model such that each of the initial helium, $\boldsymbol{Y}_{\rm init}$ and mixing-length theory parameter, $\boldsymbol{\alpha}_{\rm mlt}$ are drawn from a common distribution characterised by the set of hyperparameters, $\boldsymbol{\phi} = \{\Delta Y/\Delta Z, Y_P, \sigma_Y, \mu_\alpha, \sigma_\alpha\}$. Thus, Bayes' theorem becomes,

$$p(\boldsymbol{\theta}, \boldsymbol{\phi}|\boldsymbol{y}) \propto p(\boldsymbol{\phi})p(\boldsymbol{Y}_{\rm init}, \boldsymbol{\alpha}_{\rm mlt}|\boldsymbol{\phi})p(\boldsymbol{\psi})p(\boldsymbol{y}|\boldsymbol{\theta}), \tag{10}$$

where $\boldsymbol{\theta}$ is the same as in the NP model, each object-level parameter, $\boldsymbol{\theta}_j = \{\theta_{j,i}\}_{i=1}^{N_{\rm stars}}$ and $\boldsymbol{\psi} = \{\boldsymbol{f}_{\rm evol}, \boldsymbol{M}, \boldsymbol{Z}\}$ are the subset of object-level parameters not governed by the hyperparameters.

We assumed the initial helium and the mixing-length parameter are drawn from a normal distribution characterised by a population mean and standard deviation,

$$p(\boldsymbol{Y}_{\rm init}, \boldsymbol{\alpha}_{\rm mlt}|\boldsymbol{\phi}) = p(\boldsymbol{Y}_{\rm init}|\boldsymbol{\mu}_Y, \sigma_Y)p(\boldsymbol{\alpha}_{\rm mlt}|\mu_\alpha, \sigma_\alpha). \tag{11}$$

Regarding the first term of this equation, the mean initial helium follows a linear enrichment law with respect to the initial fraction of heavy-elements for a given star,

$$\boldsymbol{\mu}_Y = Y_P + \frac{\Delta Y}{\Delta Z}\boldsymbol{Z}_{\rm init}, \tag{12}$$

where Y_P is the primordial helium abundance fraction and $\Delta Y/\Delta Z$ is the so-called enrichment ratio. Therefore, we may write the prior PDF of initial helium given its population-level hyperparameters as,

$$p(\boldsymbol{Y}_{\rm init}|\boldsymbol{Z}_{\rm init}, \Delta Y/\Delta Z, Y_P, \sigma_Y) = \prod_{i=1}^{N_{\rm stars}} \mathcal{N}(Y_{{\rm init},i}|\mu_{Y,i}, \sigma_Y) \tag{13}$$

We justified this assumption based on theoretical and empirical evidence for a linear enrichment law (), but taking into account an intrinsic spread, σ_Y about this law due to random variations in chemical abundance throughout the interstellar medium.

Similarly, for the second term of Equation 11, we chose to partially-pool the mixing-length parameter. We assume that convection in stars of a similar mass, evolutionary stage and area of the HR diagram may be approximated using a similar value of $\alpha_{\rm mlt}$, but the accuracy of the mixing-length theory may vary from star-to-star. There is theoretical evidence for such a variation with [M/H], $T_{\rm eff}$ and $\log g$ in 3D hydrodynamical stellar models (Magic et al. 2015; Viani et al. 2018). However, investigating such dependencies are beyond this scope of this paper. Given the small range of our sample, any such variation will be absorbed by the spread parameter, σ_α. Therefore, we decided to describe the prior on $\boldsymbol{\alpha}_{\rm mlt}$ as,

$$p(\boldsymbol{\alpha}_{\rm mlt}|\mu_\alpha, \sigma_\alpha) = \prod_{i=1}^{N_{\rm stars}} \mathcal{N}(\alpha_{{\rm mlt},i}|\mu_\alpha, \sigma_\alpha) \tag{14}$$

We gave all of the hyperparameters weakly informative priors, with the exception of Y_P for which we adopt a recent measurement of the primordial helium abundance the mean [CITE PLANK] with a standard deviation representative of the range of values in the literature [CITE]. We assumed priors on the hyperparameters as

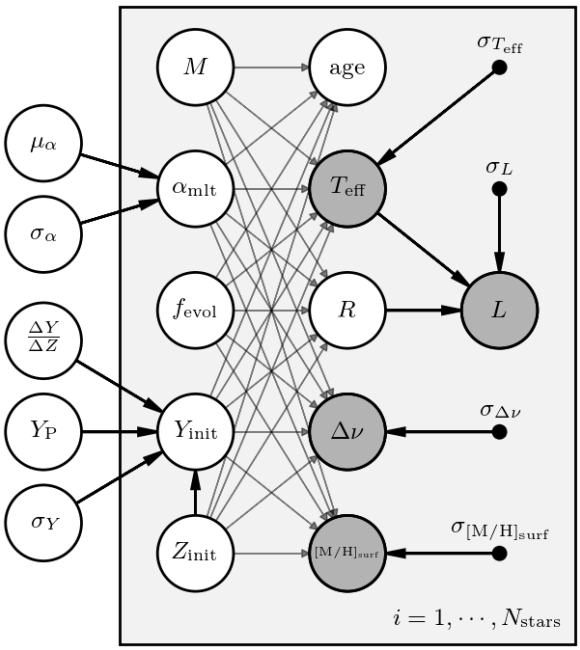

Figure 2.

follows,

$$\begin{aligned}\Delta Y/\Delta Z &\sim 4.0\,\mathcal{B}(1.2, 1.2),\\ Y_P &\sim \mathcal{N}(0.247, 0.1),\\ \sigma_Y &\sim \mathcal{LN}(0.01, 1.0),\\ \mu_\alpha &\sim 1.5 + \mathcal{B}(1.2, 1.2),\\ \sigma_\alpha &\sim \mathcal{LN}(0.1, 1.0),\end{aligned}$$

where $x \sim \mathcal{LN}(m, \sigma)$ represents a random variable drawn from the log-normal distribution,

$$\mathcal{LN}(x|m,\sigma) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left[-\frac{\ln(x/m)^2}{2\sigma^2}\right]. \quad (15)$$

3.3.3 *Max-pooled model*

We built another hierarchical model similar to the PP model except that $\alpha_{\rm mlt}$ is max-pooled. In other words, we assumed that the mixing length must be the same value for every star in the sample, but still allowed it to freely vary. Thus the hyperparameters are now, $\phi = \{\Delta Y/\Delta Z, Y_P, \sigma_Y, \alpha_{\rm mlt}\}$. The posterior distribution of the model takes the same form as in Equation 10 except that now,

$$p(\boldsymbol{\alpha}_{\rm mlt}|\alpha_{\rm mlt}) = \prod_{i=1}^{N_{\rm stars}} f(\alpha_{{\rm mlt},i}|\alpha_{\rm mlt}) \quad (16)$$

where $f(x|\alpha)$ is defined as,

$$f(x|\alpha) = \begin{cases} +\infty, & x = \alpha \\ 0, & x \neq \alpha \end{cases} \quad (17)$$

4 RESULTS

We obtained model stellar fundamental parameters for each of the NP, PP and MP models by taking the median and 68 percent credible region from the marginalised posterior samples. For the test stars, we compared the results with the true values for each of the models. We found good agreement with the true values and showed that the uncertainties on mass, age and radius decrease with increasing sample size in the PP model. We also found that the NP method over-predicts the uncertainties on the fundamental parameters due to poorly sampling $Y_{\rm init}$ and $\alpha_{\rm mlt}$. Fitting the helium enrichment law and spread in mixing-length to the NP model results recovered the true hyperparameters with higher precision than the PP and MP models. However, fitting this way limited the precision of the stellar parameters, wheras the hierarchical models reduced the uncertainties by roughly a factor of $\sqrt{N_{\rm stars}}$. The results for the test stars are shown in more detail in Appendix **??**.

With confidence that the models were able to obtain accurate stellar parameters, in accordance with our choice of stellar models, we present results for the sample of 81 *Kepler* dwarfs for each of our statistical models. Tables of results for each model are available for download at TODO.

We obtained results for the hyperparameters in each of the models and present them in Table 3. For NP, we fit the same hyperparameters from the PP model to the results from the NP model, using the same prior distributions.

The PP and MP results without the Sun are self-consistent, but the biggest difference between the two methods is evident when the Sun is added. The MP model will typically settle for a shared value of $\alpha_{\rm mlt}$ which favours the star with the best observational constraints. In our case, the solar model yields $\alpha_{\rm mlt} = 2.11^{+0.01}_{-0.01}$ which is far from $\alpha_{\rm mlt}$ of $1.73^{+0.08}_{-0.07}$ obtained without the Sun. However, when $\alpha_{\rm mlt}$ is partially-pooled, the σ_α parameter copes with this by increasing to include the Sun in the distribution. The PP models are more robust.

We noticed that the uncertainties on the NP hyperparameters were smaller than those of the hierarchical models. However, these are likely underestimated because the NP model poorly constrains $\alpha_{\rm mlt}$ and $Y_{\rm init}$ such that the bounds of the prior have the effect of reducing the standard deviation of their marginalized posteriors. On the contrary, the hierarchical models SOMETHING. . .

We found that including the sun in our PP and MP models systematically shifted the median ages of the sample by about 0.5 Gyr and 1.0 Gyr respectively. This is a direct result of the solar model favouring a higher $\alpha_{\rm mlt}$ and lower $Y_{\rm init}$ than the rest of the sample. Partially pooling the sun with the rest of the sample copes with this better by accounting for a population spread in the parameters.

5 DISCUSSION

6 CONCLUSIONS

ACKNOWLEDGEMENTS

REFERENCES

Asplund M., Grevesse N., Sauval A. J., Scott P., 2009, ARA&A, 47, 481
Ferguson J. W., Alexander D. R., Allard F., Barman T., Bodnarik J. G., Hauschildt P. H., Heffner-Wong A., Tamanai A., 2005, The Astrophysical Journal, 623, 585
Huber D., et al., 2011, ApJ, 743, 143
Magic Z., Weiss A., Asplund M., 2015, A&A, 573, A89
Paxton B., Bildsten L., Dotter A., Herwig F., Lesaffre P., Timmes F., 2011, ApJS, 192, 3

Table 3. Hyperparameter results for each model in descending order of the helium enrichment ratio, $\Delta Y/\Delta Z$.

Model	$\Delta Y/\Delta Z$	Y_P	σ_Y	μ_α	σ_α	$\alpha_{\rm mlt}$
NP	$1.69^{+0.21}_{-0.21}$	$0.247^{+0.001}_{-0.001}$	$0.0074^{+0.0026}_{-0.0022}$	$1.95^{+0.04}_{-0.04}$	$0.06^{+0.03}_{-0.02}$	—
MP	$1.60^{+0.45}_{-0.42}$	$0.247^{+0.001}_{-0.001}$	$0.0051^{+0.0044}_{-0.0027}$	—	—	$1.73^{+0.08}_{-0.07}$
PP	$1.60^{+0.45}_{-0.42}$	$0.247^{+0.001}_{-0.001}$	$0.0051^{+0.0045}_{-0.0027}$	$1.74^{+0.08}_{-0.07}$	$0.06^{+0.05}_{-0.03}$	—
PPS	$1.05^{+0.28}_{-0.25}$	$0.247^{+0.001}_{-0.001}$	$0.0045^{+0.0038}_{-0.0023}$	$1.90^{+0.09}_{-0.09}$	$0.13^{+0.06}_{-0.05}$	—
MPS	$0.76^{+0.24}_{-0.27}$	$0.247^{+0.001}_{-0.001}$	$0.0049^{+0.0039}_{-0.0025}$	—	—	$2.09^{+0.03}_{-0.03}$

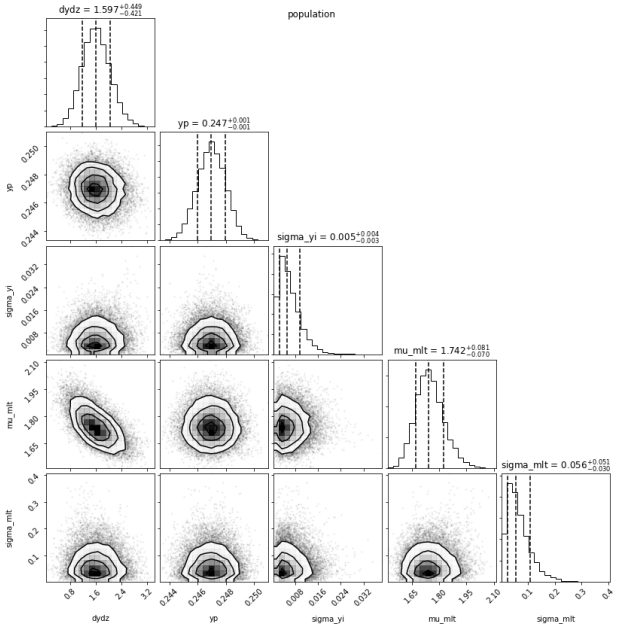

Figure 3. Population hyperparameters for model without the Sun.

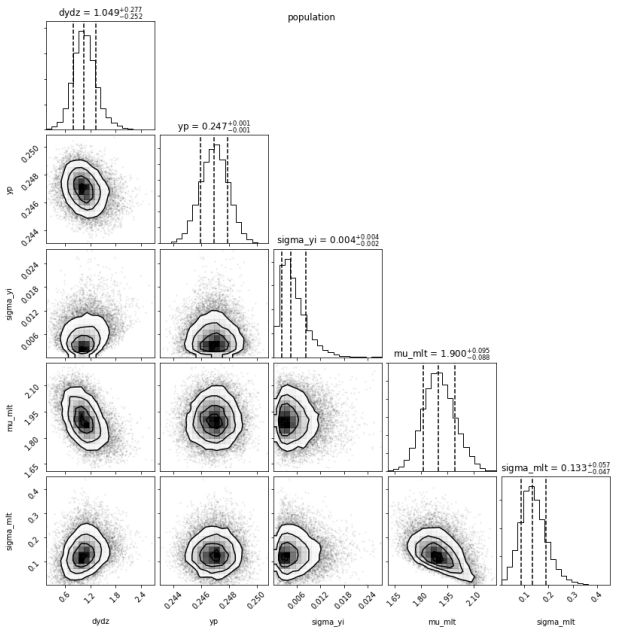

Figure 4. Population hyperparameters for model with the Sun.

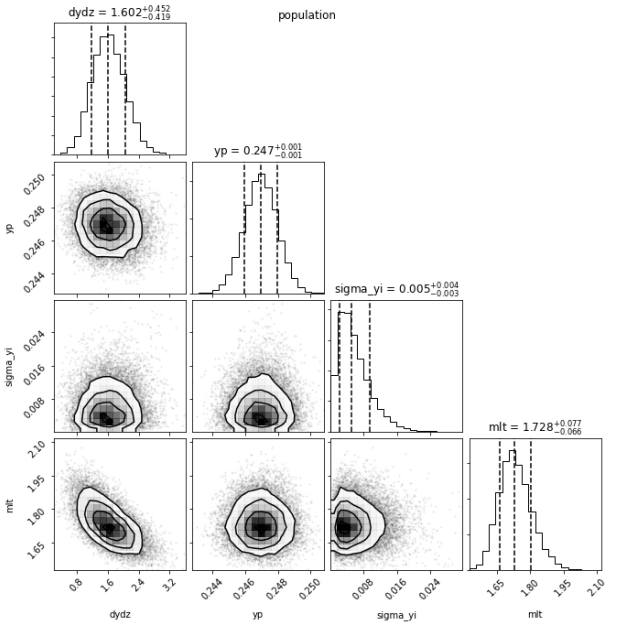

Figure 5. Population hyperparameters for model without the Sun.

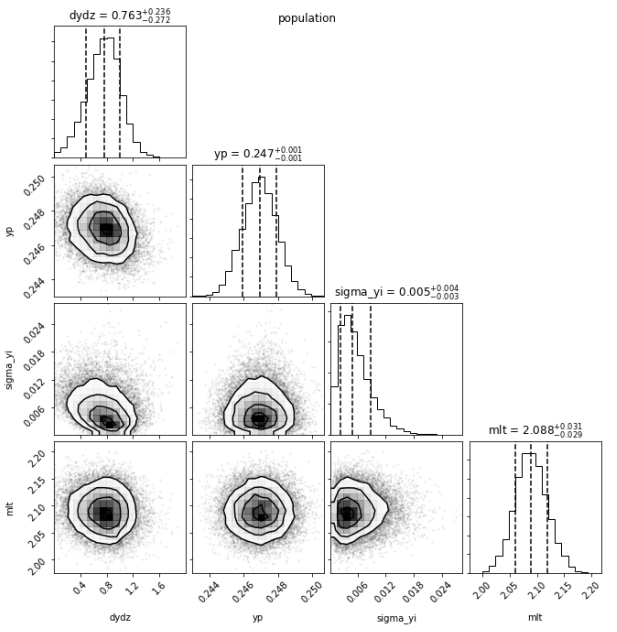

Figure 6. Population hyperparameters for model with the Sun.

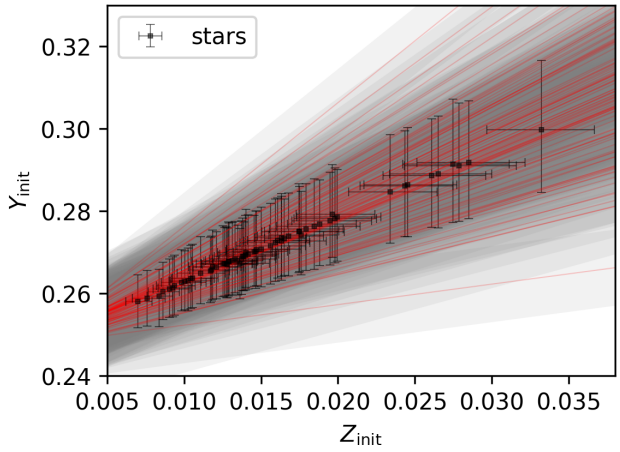

Figure 7. Without sun

Paxton B., et al., 2013, The Astrophysical Journal Supplement Series, 208, 4
Paxton B., et al., 2015, The Astrophysical Journal Supplement Series, 220, 15
Paxton B., et al., 2018, The Astrophysical Journal Supplement Series, 234, 34
Paxton B., et al., 2019, The Astrophysical Journal Supplement Series, 243, 10
Rogers F. J., Nayfonov A., 2002, The Astrophysical Journal, 576, 1064
Serenelli A., et al., 2017, ApJS, 233, 23
Townsend R. H. D., Teitler S. A., 2013, Monthly Notices of the Royal Astronomical Society, 435, 3406
Viani L. S., Basu S., Ong J. M. J., Bonaca A., Chaplin W. J., 2018, The Astrophysical Journal, 858, 28
White T. R., Bedding T. R., Stello D., Christensen-Dalsgaard J., Huber D., Kjeldsen H., 2011, The Astrophysical Journal, 743, 161
Zinn J. C., Pinsonneault M. H., Huber D., Stello D., Stassun K., Serenelli A., 2019, ApJ, 885, 166

APPENDIX A: TESTING THE METHOD

A1 Synthetic population from the neural network

A2 Synthetic population from MESA

A3 Synthetic stars from other works

This paper has been typeset from a TEX/LATEX file prepared by the author.