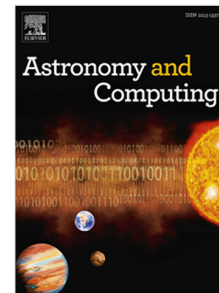# Journal Pre-proof

Unveiling the power of uncertainty: A journey into Bayesian Neural Networks for stellar dating

Víctor Tamames Rodero, Andrés Moya Bedón, Luis Manuel Sarro Baro, Roberto Javier López-Sastre

Please cite this article as: V.T. Rodero, A.M. Bedón, L.M.S. Baro et al., Unveiling the power of uncertainty: A journey into Bayesian Neural Networks for stellar dating. *Astronomy and Computing* (2025), doi: https://doi.org/10.1016/j.ascom.2025.100957.

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Unveiling the Power of Uncertainty: A Journey into Bayesian Neural Networks for Stellar dating

Víctor Tamames Rodero[a], Andrés Moya Bedón[a], Luis Manuel Sarro Baro[b], Roberto Javier López-Sastre[c]

[a]*Departament d'Astronomia i Astrofísica, Universitat de València, C/ Dr. Moliner, 50, Burjassot, 46100, Valencia, Spain*
[b]*Departamento de Inteligencia Artificial, ETSI Informática, UNED, Juan del Rosal, E-16, 28040, Madrid, Spain*
[c]*GRAM Research Group, Department of Signal Theory and Communications, University of Alcalá, Alcalá de Henares, 28805, Madrid, Spain*

Abstract

Context: Astronomy and astrophysics demand rigorous handling of uncertainties to ensure the credibility of outcomes. The growing integration of artificial intelligence offers a novel avenue to address this necessity. This convergence presents an opportunity to create advanced models capable of quantifying diverse sources of uncertainty and automating complex data relationship exploration.

What: We introduce a hierarchical Bayesian architecture whose probabilistic relationships are modelled by neural networks, designed to forecast stellar attributes such as mass, radius, and age (our main target). This architecture handles both observational uncertainties stemming from measurements and epistemic uncertainties inherent in the predictive model itself. As a result, our system generates distributions that encapsulate the potential range of values for our predictions, providing a comprehensive understanding of their variability and robustness.

Methods: Our focus is on dating main sequence stars using a technique known as Chemical Clocks, which serves as both our primary astronomical challenge and a model prototype. In this work, we use hierarchical architectures to account for correlations between stellar parameters and optimize information extraction from our dataset. We also employ Bayesian neural networks for their versatility and flexibility in capturing complex data relationships.

Results: By integrating our machine learning algorithm into a Bayesian

framework, we have successfully propagated errors consistently and managed uncertainty treatment effectively, resulting in predictions characterized by broader uncertainty margins. This approach facilitates more conservative estimates in stellar dating. Our architecture achieves age predictions with a mean absolute error of less than 1 Ga for the stars in the test dataset.

Keywords: Bayesian Neural Networks, Hierarchical modelling, Stellar dating, Uncertainty, Deep Learning, Chemical Clocks

1. Introduction

Nowadays, in many different fields, we see how Artificial Intelligence (AI) approaches emerge; new machine learning (ML), deep learning (DL) algorithms, and other statistical models come to the stage to help us automatize and extract as much information from our datasets as possible. These new tools are changing our workflow and the way we treat our data, promising to be a potential solution for many tasks in the near future.

In the dynamic field of AI, a major challenge has been the ability to clearly express the uncertainties associated with their predictions. Notwithstanding these limitations, significant efforts have been made to enhance and elucidate the functioning of these algorithms, with the objective of developing architectures that can address uncertainties.

Our objective is to apply this capability to stellar dating, with the aim of improving the accuracy and confidence of stellar age estimates. Employing varied methods, from traditional isochrone fitting (e.g. Valls-Gabaud, 2014; Rodrigues et al., 2017) to more modern techniques like gyrochronology (e.g. Barnes, 2003; Angus et al., 2015; Mathur et al., 2023a), asteroseismology (e.g. Moya et al., 2010; Silva Aguirre et al., 2015a,b; Mathur et al., 2023b) and chemical abundance analysis (e.g. da Silva et al., 2012; Nissen, 2015, 2016; Nissen et al., 2020; Moya et al., 2022), accurate stellar dating in astrophysics demands rigorous uncertainty treatment for reliable results. When estimating stellar ages one is confronted with multifaceted challenges in accurately estimating the parameter. Inherent uncertainties within theoretical models describing stellar evolution pose a significant hurdle, as these models rely on assumptions about intricate physical processes, introducing potential inaccuracies. Limitations in observational data quality, characterized by incomplete or sparse data, particularly for distant or faint stars, hinder precise age determinations. Moreover, degeneracies in stellar properties used for obtaining

ages, along with calibration intricacies, can introduce biases, complicating age estimates. Furthermore, age dispersions observed within stellar populations challenge the uniformity of stellar ages within clusters, impeding a comprehensive understanding of stellar formation and evolution. Addressing these challenges necessitates refined models, improved observational capabilities, meticulous calibration techniques, and innovative, interdisciplinary approaches to refine our grasp of stellar ages and cosmic chronologies.

Modern advancements in computational models and observational techniques have substantially enhanced contemporary stellar dating methodologies. Notably, the advent of missions such as Corot (France, ESA), Kepler, and TESS (NASA) has revolutionized age estimations through the application of asteroseismology (e.g. Moya et al., 2010; Mathur, 2013; Silva Aguirre et al., 2015a,b; Soderblom, 2015b; Morel et al., 2021; Mathur et al., 2023b; Goupil et al., 2024). These missions have played a transformative role in refining our understanding of stellar ages. Moreover, the anticipated contributions of missions like Plato 2.0 (ESA) hold substantial promise for delving deeper into stellar age determinations, offering the potential for significant advancements in our comprehension of cosmic chronologies.

Overall, in the field of stellar dating techniques, it's essential to acknowledge that no single method universally applies to all types of stars, except for isochrones model fitting. Depending on observations, this approach holds the potential to yield age predictions with very tight uncertainties. However, this could represent a limitation in broader applicability, as it might be deemed unrealistic or overly constrained in certain contexts. Work has been done in this direction, using Bayesian statistical approaches to manage and better propagate uncertainties for more realistic age estimates (e.g. Guédé et al., 2012; Serenelli et al., 2013; Brandt and Huang, 2015; Angus et al., 2015; von Hippel et al., 2016; Jeffery et al., 2016; Almeida-Fernandes and Rocha-Pinto, 2018; Lin et al., 2018; Kiman et al., 2020, 2021, 2022; Moya et al., 2022; Mathur et al., 2023a,b; Lu et al., 2024), some of them use hierarchical architectures (e.g. Angus et al., 2015; Olivares Romero, 2017; Moya et al., 2022), and other ones using ML techniques (e.g. Basu, 2020; Bu et al., 2020; Moya et al., 2021; Van-Lane et al., 2023).

Currently, using clusters for stellar dating (e.g. Viscasillas Vázquez et al., 2022) and asteroseismology emerge as promising alternatives, each with its distinct strengths and constraints. Calculating the age of stars through cluster characterization, while effective in certain contexts, often yields quantized databases, constraining their utility in broader astronomical applications. On

3

the other hand, asteroseismology, renowned for its precision and accuracy, has predominantly been applied to giants and a select few hundred main sequence (MS) stars of specific spectral types. Despite their advancements, these methodologies encounter challenges in achieving comprehensive coverage across stellar populations.

In this paper, we present a novel approach designed for dating main sequence stars by leveraging machine learning and statistical methods. Our model aims to provide a more accurate and conservative approach to stellar dating by quantifying uncertainties in the predictions. By utilizing machine learning, we can automatically capture intricate relationships within the data, thus potentially enhancing the quality of our estimates. Our contributions are rooted in embedding this approach within a Bayesian framework, known for its robust and well-established methodologies. By employing a hierarchical architecture, our approach not only quantifies uncertainties but also maximizes the extraction of information from available datasets, leading to more reliable and comprehensive stellar age determinations.

1.1. Scope & case

In this study, we adopt a hierarchical Bayesian framework integrated with machine learning algorithms, specifically multilayer perceptrons, to model the relationships embedded within our dataset and be able to derive robust distributions of potential stellar ages. In this work we decided to use the chemical clocks (CCs) technique for calculating stellar ages, which exploits the unaltered surface chemical compositions of stars as proxies for their formation epochs. MS stars are particularly well-suited for this approach because their photospheric abundances remain largely unchanged throughout their lifetimes (e.g. Delgado Mena et al. 2019b, from now on DM19, Viscasillas Vázquez et al. 2022; Moya et al. 2022). Once established at formation, the chemical composition of an MS star effectively serves as a relic of the interstellar medium (ISM) or formation cloud at that time. Aside from minor effects such as atomic diffusion, the surface layers of these stars do not experience significant alterations, preserving the original chemical signature. This characteristic stability is central to the chemical clocks method, as it enables the inference of stellar ages directly from the observed abundance patterns. The foundation of the CCs approach lies in the framework of Galactic chemical evolution. The

4

ISM is progressively enriched over time by a variety of nucleosynthetic sources that operate on different timescales. By empirically calibrating the abundance ratios of elements produced on these disparate timescales (e.g., [Y/Mg] or [Ba/Al]), chemical clocks effectively timestamp the epoch at which a star's birth environment was chemically characterized. Thus, a star's immutable chemical signature encodes vital information about the temporal evolution of the formation cloud. It is important to note that while the majority of heavy elements in MS stars remain stable, certain species do undergo evolutionary changes. However, the reliability of the chemical clock method hinges on the systematic variation of abundance ratios, primarily governed by the nucleosynthetic history of the ISM rather than by intrinsic stellar evolution. This distinction is critical, as it ensures that the observed abundance ratios serve as faithful indicators of stellar age. The applicability of the chemical clock is one of its main advantages since it does not depend on the evolution and structure of the stars, but the method varies across different stellar types. The technique is most effective for low- and intermediate-mass stars, which possess convective envelopes that do not facilitate extensive mixing during the MS phase. In contrast, high-mass stars, characterized by radiative envelopes and rapid evolution may experience convective dredge-up events that transport nuclear-processed material to the surface, thereby altering their chemical abundances and compromising the chemical clock signal. In these cases, either alternative chemical tracers or additional corrective calibrations are necessary to obtain reliable age estimates. CCs offer a valuable complement to other established age-dating methodologies, including gyrochronology, isochrone fitting, and asteroseismology. Gyrochronology, which correlates stellar rotation periods with age, is most effective for cool dwarfs but relies on precise rotational data. Isochrone fitting, based on luminosity and effective temperature, can suffer from degeneracies in certain regions of the Hertzsprung–Russell diagram (e.g. regions where isochrones are tightly packed). Although asteroseismology provides highly precise age determinations, it demands observational resources that may not be available for large samples. In contrast, chemical clocks can be readily applied to extensive spectroscopic surveys, thereby facilitating robust statistical

analyses of stellar populations and extending age estimates to stars that are challenging to date using other techniques.

2. Tools & data sample

2.1. Tools

Inside the vast field of AI, neural networks (NNs) stand out from the rest of the algorithms thanks to their potential versatility to adapt to data, tasks, or domains. However, the limitations of such networks became evident over time and it wasn't until the last decades when we witnessed a remarkable transformation in NNs, driven by advancements in Gradient-based learning (Lecun et al., 1998), the availability of powerful hardware such as graphics processing units (GPUs), and large datasets. This has led to the development of deep neural networks, including other architectures like deep convolutional or recurrent neural networks (e.g. Krizhevsky et al., 2012; Silver et al., 2017).

These models have improved a lot in the last few years, but they often require large amounts of labelled, high-quality data for training, which is not always available. In addition, NNs lack interpretability, making it challenging to understand the reasoning behind their predictions, especially when they make errors or biased decisions (e.g. Dong et al., 2017; Ancona et al., 2017; Räuker et al., 2022). Furthermore, traditional deep learning models were not initially designed for quantifying uncertainty and generally focus on providing point estimates as predictions. Although, there are algorithms like Bayesian Neural Networks (BNNs) (Tishby et al., 1989), Monte Carlo Dropout (Gal and Ghahramani, 2015b), Deep Ensembles (Krizhevsky et al., 2012) or Masksembles (Durasov et al., 2020) that can produce approximations of uncertainties. In this context, probability theory has become a cornerstone for modeling uncertainty in machine learning, seamlessly integrating probabilistic modeling with the predictive capabilities of ML algorithms. In our work, we leverage BNNs, a powerful framework for effectively incorporating and managing uncertainty in AI. This approach represents a significant step toward building reliable and interpretable systems.

BNNs offer a powerful approach for incorporating uncertainty into predictions by modeling their weights and biases as probability distributions rather than fixed values. Unlike traditional neural networks that yield point estimates, BNNs quantify uncertainty in both model parameters and outputs, which is essential for applications demanding reliable decision-making and interoperability.

In particular, by modeling uncertainty, BNNs mitigate overfitting: they capture epistemic uncertainty and avoid overconfident predictions through implicit regularization. This capability is especially important in our application, where understanding the confidence of predictions is critical.

Bayesian statistics underpins this methodology by providing a formal framework to represent uncertainty in model parameters and observations through probability distributions. This framework not only enables the calculation of credible intervals and distributions for parameter estimates, contrasting with the frequentist approach, which assigns probabilities to data rather than hypotheses, but also allows the integration of prior knowledge. Priors in the Bayesian framework discourage extreme weight values, while the posterior distribution naturally favors simpler models when data is scarce. Moreover, by marginalizing over multiple weight configurations instead of relying on a single optimized set, BNNs reduce sensitivity to noise in the training data. As new data is gathered, Bayesian models update the probability of hypotheses, a feature that is fundamental for our stellar characterization model, where handling uncertainties at every step is critical.

In this context, hierarchical models emerge as a particularly valuable tool. These architectures have the ability to capture and account for variations at multiple levels, encompassing both within-group and between-group variations simultaneously. This capability is especially beneficial when dealing with correlated data or where different levels may exhibit distinct sources of variability, providing a more nuanced and accurate analysis. Hierarchical models are often applied within a Bayesian framework because they facilitate the integration of prior knowledge and enable the quantification of uncertainty in parameter estimates. By including prior knowledge and data likelihood, the posterior distribution offers a comprehensive depiction of parameter information, providing a holistic understanding of the model's nuances post-data observation (e.g. Gelman and Hill, 2006; Gelman et al., 2015). They excel in handling situations with limited data, providing stability and informativeness in parameter estimates by borrowing information from other levels of the hierarchy (e.g. Davidian, 2003; Gelman et al., 2004; Gelman and Hill, 2006). Considering these advantages and inspired by the promising results presented in Moya et al. (2022), we have adapted and extended its Hierarchical Bayesian Model (HBM) to suit the specific requirements of our

current experimentation. Technically, in this work, we built upon the HBM architecture proposed in Moya et al. (2022), which models probabilistic relationships using multi-linear regressions. However, we enhanced the model by replacing the multi-linear components with a machine learning module, specifically neural networks (NNs).

In our hierarchical model, we use Markov Chain Monte Carlo (MCMC) methods for parameter inference, leveraging their ability to estimate posterior distributions in Bayesian frameworks. MCMC methods are a class of computational algorithms that rely on repeated random sampling to approximate these distributions, making them particularly suited to complex, high-dimensional models where traditional methods fall short. The need for such techniques has driven advances in probabilistic programming, with languages like Stan, PyMC, and Pyro making MCMC more accessible and applicable to sophisticated models.

However, MCMC methods are also computationally demanding, which can limit their scalability, especially in large models or time-sensitive applications. Addressing these limitations is an active area of research, with current work in BNNs and deep learning–based approaches focused on making these methods more efficient and scalable. These innovations are helping to broaden the practical use of Bayesian inference for uncertainty quantification in machine learning.

2.2. Training sample

The training dataset utilized in our study is thoroughly documented in (Delgado Mena et al., 2019a). It comprises observations of 1059 stars from the HARPS-GTO planetary search program. These stars were selected from a volume-limited area within roughly 70 parsecs of the Sun, with few stars located at greater distances.

Details about derived effective temperature (T_{eff}), metallicity ([Fe/H]), surface gravity (log g), and chemical abundances ([X/Fe], where X represents the different chemical elements) are presented in DM19. Stellar ages **(from now on t)** were computed utilizing the PARAM v1.3 tool, employing PARSEC isochrones (Bressan et al., 2012) in conjunction with a Bayesian estimation method (da Silva et al., 2006). However, it is pertinent to highlight that not all age determinations among these 1059 stars are deemed reliable. DM19 established reliable age estimates as those presenting an age uncertainty smaller than 1.5 billion years (Ga). Stars with age uncertainties

8

below 0.2 Ga were omitted because we regard them as unrealistic for a standard isochrone fitting process, having the potential to introduce bias into the final model (Moya et al., 2022). **Adopting a uniform threshold was a practical decision to avoid introducing biases that might arise from a non-uniform error selection**. Consequently, this resulted in a refined dataset comprising 328 stars extracted from the original set of 1059 stars for our research. Further elucidation regarding the primary characteristics of this subset can be found in DM19.

In summary, our investigation was centered on categorizing 244 thin-disk stars, 14 high-α metal-rich stars, 68 thick-disk stars, and 2 halo stars, following the classification methodology delineated by Adibekyan et al. (2011, 2012). The effective temperature ranged from 5010 to 6788 K, with 95% of the values falling between 5271 and 6416 K. The surface gravity spanned from 3.73 to 4.71 dex, with 95% of the measurements ranging between 3.93 and 4.58 dex. Additionally, the metallicity exhibited variation from -1.15 to 0.55 dex, with 95% of values within the range of -0.81 to 0.33 dex. Each parameter measurement was accompanied by its corresponding observational error. We chose to use the chemical abundances of Mg, Al, Si, Zn, Ti, Sr, and Y for these stars.

2.3. Testing sample

A distinct testing set comprising stars not included in the training was employed to assess the accuracy of the HBM predictions, produced using the training dataset detailed in Section 2.2. The specific composition of this dataset, as referenced in Moya et al. (2022), is twenty-three stars designated with 'reliable' age estimates. They are delineated below:

- Twenty stars whose age determination was derived through asteroseismology. Specific characteristics of eighteen Kepler Input Catalog (KIC) stars, as well as 16 Cygni A and 16 Cygni B, were acquired from Nissen et al. (2017) and Morel et al. (2021). Their chemical abundances were acquired via specialized observations. For data sourced from Nissen et al. (2017), $log(g)$ uncertainties were not explicitly provided; thus, a standard and conservative uncertainty of 0.05 dex was imposed. The ages utilized in this context were primarily sourced from Aguirre et al. (2017).

- Two stars belong to the M67 open cluster (this is one of the best-characterized clusters). Details about these stars were sourced from

9

Liu et al. (2016), which are identified as solar twins. The age of the cluster was obtained from Yadav et al. (2008).

- The Sun, whose general characteristics were taken from Prša et al. (2016). **To minimize biases and facilitate direct comparisons with previous works, we adopt the same abundance reference as in prior studies Sousa et al. (2007, 2008)**.

We consider them more reliable than other selections because they were obtained through asteroseismology and cluster characterization, which nowadays represent two of the most accurate methods for dating MS stars. These stars exhibit well-determined effective temperature, surface gravity, metallicity, age, and chemical abundances of Mg, Al, Si, Zn, Ti, Sr, and Y relative to iron, along with their associated uncertainties. It is noteworthy that not all stars have available data for all chemical abundances, contributing to the realistic nature of our testing dataset. In the case of the Sun, it was treated as a standard field star within the testing dataset. For doing so we intentionally adjusted its uncertainties to align with those of the remaining stars.

3. Inference technique: HBM

As it was previously introduced in Section 2.1, in this work we decided to inherit the HBM architecture used in Moya et al. (2022), whose probabilistic relationships are modeled by multi-linear regressions (from now on HBM-MLR). Thanks to this decision we are able to stratify information for hyperprior specification, account for observational uncertainties, and propagate the errors through the model. In this work, we replaced the multi-dimensional linear models, with a ML module, specifically, NNs (from now on HBM-NNs). This new approach eliminates the need for manual exploration of internal data relationships, as previously suggested in Delgado Mena et al. (2019b). The schematic of the architecture is depicted in Figure 1.

3.1. Bayesian Neural Networks

Drawing upon insights from Dotter et al. (2017) and Gavel et al. (2021), we acknowledge the proposition that all potential predictor variables may carry pertinent physical information. In pursuit of optimal data treatment, several strategic decisions were undertaken. Firstly, our methodology aligns with the principles advocated by Gelman et al. (2004), wherein all available
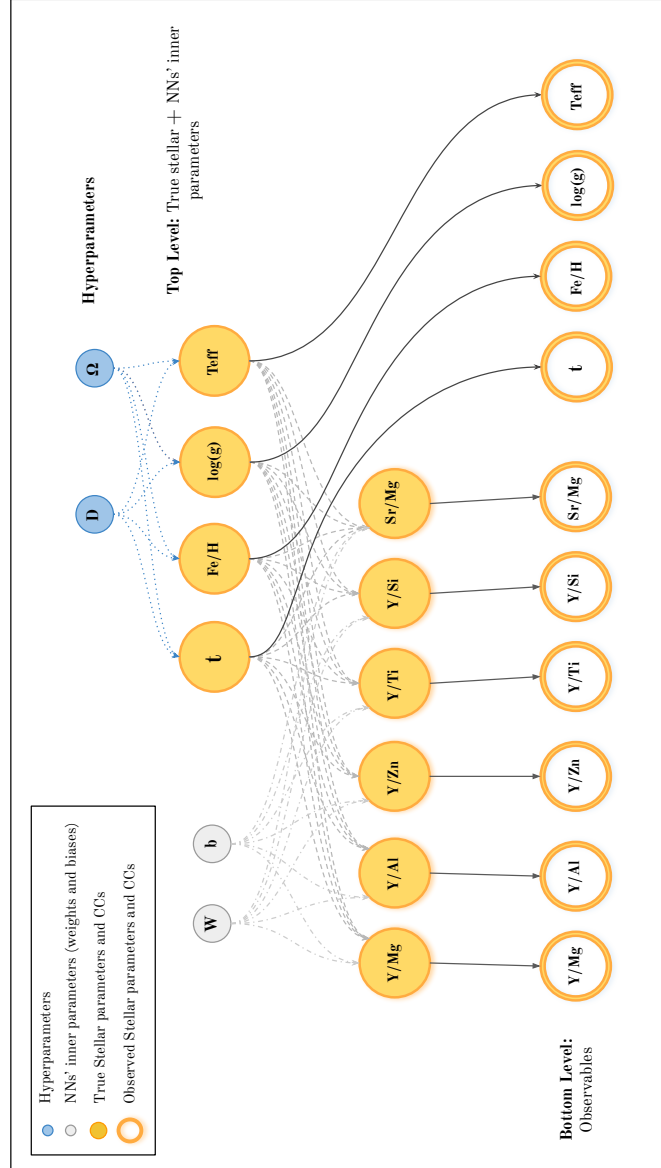
10

Figure 1: Structure of the hierarchical Bayesian model used in this work, comprising two levels: the *Top* and *Bottom*. Hyperparameters Ω (correlation matrix) and D (scaling matrix), which are used to define the covariance matrices for the stellar parameters represent the model's hyperpriors. Additionally, $\boldsymbol{W}$ and $\boldsymbol{b}$ are the weights matrices and biases vectors, respectively, of the neural networks. *Filled circles* are the true stellar parameters and CCs, and *empty circles* represent the observed ones.

11

predictors are incorporated into our BNN model, with priors for the neural network weights and biases centered at zero. This decision serves as a regularization mechanism, ensuring that the posterior distributions of neural network coefficients diverge significantly from zero solely when supported by empirical evidence (Gelman et al., 2004). Additionally, an arrangement was imposed on the bias vectors of hidden layers (HLs) in the BNNs. The reordering of a bias vector b with components b_k is done in ascending order ($b_k < b_{k+1}$ for $k \in \{1, \ldots, k-1\}$). This deliberate ordering aims to break weight-space symmetries, thereby enhancing general identifiability of BNNs. The organized bias vectors contribute to the network's discernment of relevant patterns and dependencies, reducing the risk of underfitting or oversimplification. This, in turn, augments the network's learning capacity, promoting accurate predictions and improving generalization (Pourzanjani et al., 2017). Furthermore, the chosen activation function for the BNNs' HLs, namely the classical sigmoid, is applied within nodes after the linear combination of the inputs with the weights and biases, effectively constraining the domain of internal neural network parameters. The output of the hidden units is defined as

$$h_i = f\left(\sum_{j=1}^{n} w_{ji}x_j + b_i\right), \tag{1}$$

where i indexes neurons in the hidden layer from 1 to m (the number of neurons in the hidden layer). Index j indexes the input features. It ranges from 1 to n, where n is the total number of input features. x_j denotes the j-th input feature and w_{ji} stands for the weight connecting the j-th input feature to the i-th neuron in the hidden layer. Each input feature x_j is weighted by w_{ji} before being summed up and passed through the activation function f (sigmoid). b_i represents the bias term for the i-th neuron in the HL. The output value of the neural network, y, is calculated as

$$y = \sum_{i=1}^{m} w_i h_i + b. \tag{2}$$

In the primary case of a single CC per BNN (Figure 2), in the output layer the weight w_{ji} is represented as a vector, while the bias b_i is represented as a scalar. Our model would then be defined by the inner operations of the neural networks, in this case, the feed-forward multilayer perceptron.
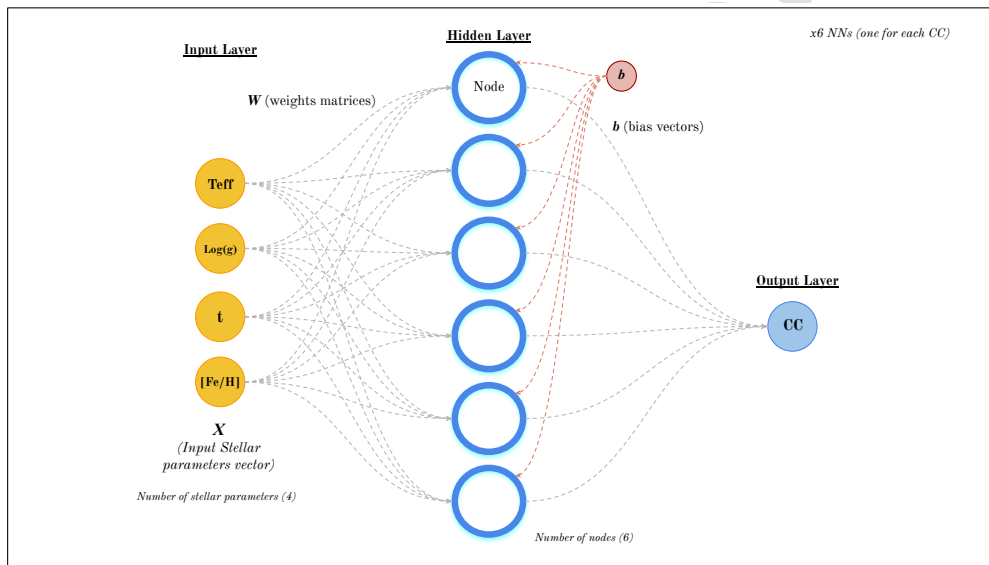
12

Figure 2: An example of a single-output NN architecture with one hidden layer, matching the best BNN schematics used for single-output configurations during experimentation. **W** and **b** represent the weights matrices and biases vectors, illustrated through the dashed gray and red lines, respectively, depicting the information flow within the NN. Additionally, *Orange filled circles* are the stellar parameters (X) as input, and *Blue filled circle* is the single CC output.

13

The dimensions of the weights and biases in NN architectures depend on the number of inputs, nodes, and HLs. To maintain clarity and simplify nomenclature, we define $\boldsymbol{W}$ and $\boldsymbol{b}$ as the sets encompassing all weight matrices and bias vectors, respectively, used in any given experimental configuration, including all 6 NNs (one for each CC). Specific details will be provided as necessary for single NN with multi-output experiments.

3.1.1. Posterior sampling techniques

Bayesian inference updates priors using observed data to compute the posterior distribution, encapsulating the model's uncertainty (epistemic). Specifically, within BNNs, approximation techniques such as MCMC or Variational Inference (VI) (e.g. Gelman et al., 2004; Gelman and Hill, 2006; Liu and Wang, 2016; Gal and Ghahramani, 2015a) are employed due to the intractable nature of the posterior distribution over neural network parameters (e.g. Wilson and Izmailov, 2020; Arbel et al., 2023; Wiese et al., 2023; Papamarkou et al., 2021). In our case, we used PyStan (Gelman et al., 2015), which is the Python version of the probabilistic work-frame Stan (Carpenter et al., 2017). Among the different sampling methods offered by this language, we used the NUTS version of the Hamiltonian Monte Carlo (HMC) (e.g. Gelman et al., 2004; Gelman and Hill, 2006; Hoffman and Gelman, 2011) sampler to obtain the desired samples from the posterior distribution. This approach better fits our needs in the sense that even if the computing time is generally longer than typical VI ones, we sample directly from the posterior distribution and not from an approximation of it.

Our Bayesian model was build using the 2.19.1.1 PyStan version, which just support CPU. Experiments were launched on Intel® i9-13905H and Intel® Xeon® E5-2695 v4 processors, both ranging a clock speed between 2.6-3.3 GHz.

3.2. Hierarchical Architecture

As depicted in Figure 1, our model architecture is defined by two layers or groups. The first layer in which we can find the true stellar parameters Θ (log g, T_{eff}, t, and [Fe/H]) and the inner parameters of the NNs used for predictions (Figure 2), and the final layer, where the observables are. The latter are characterized as random variables, modelled as a normal distribution centred around the actual values. Their variability is defined by the uncertainties inherent in the measurements detailed in the datasets section.

14

In our context, *observed* parameters refer to the direct measurements from different techniques such as spectroscopy or photometry, which inherently include uncertainties. *True* parameters, on the other hand, correspond to the latent stellar properties inferred through our hierarchical Bayesian model, which accounts for observational errors and propagates uncertainties.

Measurement uncertainties of the stellar parameters were introduced in the model through multivariate Gaussian distributions centered at the mean value of the observations and constrained by the uncertainties themselves through a covariance matrix (for deeper insight please refer to Moya et al., 2022). This way, we were able to take into account the two sources of uncertainty: random noise due to the measurement process; and the epistemic uncertainty that affects the parameters of our model.

Our model is hierarchical in the sense that the prior on the stellar parameters is learned from the data. Figure 1 shows the two information levels of our hierarchical model, where the hyperparameters Ω and D represent the correlation and scaling matrices needed to define the covariance matrix of the distribution of stellar parameters. W and b as the sets of weights matrices and biases vectors. *Filled circles* are the true stellar parameters and CCs, and *empty circles* represent the observed ones. In this representation, the connection between the stellar parameters and CCs is through multilayer perceptrons (dashed grey lines). In our case, we used 6 NNs for most experiments, one for each CC (Figure 2). The main reason for this decision was to avoid multimodality in the NNs parameter space (later explained in Section 4).

Following Moya et al. (2022) and Delgado Mena et al. (2019b) we decided to work with 7 chemical abundances; 5 α-elements (Al, Mg, Si, Ti, Zn) and 2 s-process ones (Y and Sr) to build our CCs, leading to 10 potential combinations between the two groups. Among these, 5 ratios were formed with Y ([Si/Y], [Mg/Y], [Ti/Y], [Zn/Y], and [Al/Y]), while one ratio was formed with Sr ([Mg/Sr]). This decision was motivated by the fact that Y is generally easier to obtain and often yields more precise values compared to *Sr*. Besides that, the remaining combinations were not independent of the ones used.

Once the hierarchical Bayesian model is defined we can distinguish two computational stages: 1) the training phase (this stage **runs the bayesian inference using MCMC algorithms and** uses only the training set defined in section 2), where we use labelled data and derive posterior distri-

15

butions for the NNs coefficients ($\boldsymbol{W}$ and $\boldsymbol{b}$), the stellar parameters and the hyperparameters D and Ω; and 2) the prediction stage, wherein the model is used to estimate ages for stars not present in the training set.

During the training stage, we sample 4x328 parameters (the 4 stellar variables of each of the 328 stars present in the training set), the hyperparameters and the NNs parameters, which in our case are going to be $6 \cdot N_{HLs} \cdot N_{Nodes}$ (where we are assuming an equal number of nodes per HL and in all 6 NNs used). Even though this was the main NN configuration during our work, different architectures were tested, and their proper definitions will be introduced in the corresponding sections, 4 and Appendix A. We distinguish $\hat{\theta}$ for the *observed* stellar parameters and θ for the *true* or the ones from the dataset. The same distinction applies to the CC vectors (*observed* $\hat{c}_i$, *true* c_i).

The distributions of probabilities of the stellar parameters inferred during the sampling of the training phase are used in the NNs (along with the $\boldsymbol{W}$ and $\boldsymbol{b}$ distributions) to produce the true CCs values. We assume that stars present a known and independent covariance matrix from each other and that there is not a direct correlation between stellar parameters and CCs. This approach lets us construct individual likelihoods for both groups of parameters (CCs and stellar attributes) in a simpler way by multiplying the probabilities of each observation. Then we can define the joint likelihood function of CCs and stellar parameters by multiplying the individual likelihoods as follows,

$$\mathcal{L} = \prod_{i=1}^{328} p(\hat{c}_i|\theta_i, \boldsymbol{W}, \boldsymbol{b}) \cdot p(\hat{\theta}_i|\theta_i), \tag{3}$$

where $\boldsymbol{W}$ and $\boldsymbol{b}$ represent the set of all matrices and vectors used in all 6 NNs used during the experiment, respectively. Applying Bayes' rule we can finally express the posterior distributions of parameters as

$$p(\boldsymbol{W}, \boldsymbol{b}, \Theta|\hat{c}, \hat{\Theta}) \propto \mathcal{L} \cdot \pi(\Theta, D, \Omega) \cdot \pi(\boldsymbol{W}) \cdot \pi(\boldsymbol{b}), \tag{4}$$

where

$$\pi(\Theta, D, \Omega) = p(\Theta|D, \Omega) \cdot \pi(\Omega) \cdot \pi(D). \tag{5}$$

Equation (4) represents the posterior distribution of the parameters $\boldsymbol{W}$, $\boldsymbol{b}$ given the observed values of the chemical abundance ratios $\hat{c}$ and of the

16

stellar parameters $\hat{\Theta}$. It is proportional to the product of the likelihood function $\mathcal{L}$, which captures the probability of the observed data given the parameters, and the prior distributions: $\pi(\Theta, D, \Omega)$, which accounts for the hierarchical structure of the hyperpriors D and Ω needed to model Θ, along with $\pi(\boldsymbol{W})$ and $\pi(\boldsymbol{b})$, representing the priors for $\boldsymbol{W}$ and $\boldsymbol{b}$, respectively. In Equation (5), the joint prior distribution is expressed as the product of the conditional prior, reflecting Θ's dependence on D and Ω, and the hyperprior distributions $\pi(\Omega)$ and $\pi(D)$, which account for uncertainties in the higher-level parameters.

In our approach, we firstly opted for utilizing Gaussian distributions with varying centering for the initialization of the different weight matrices and bias vectors priors. Through iterative experimentation, we observed that aligning all BNNs priors to 0 significantly improved the predictive performance. Besides that, we decided to use diagonal matrices as their covariances matrices to enforce the independence of parameters for convenience and flexibility during experimentation. In this context, the standard deviation σ, which serves as a measure of how the distribution spreads around its mean, is used as a uniform value across all elements of the diagonal covariance matrices applied to both $\boldsymbol{W}$ and $\boldsymbol{b}$. Then, this parameter σ, emerged as a critical factor during the training phase (e.g. Fortuin, 2021; Noci et al., 2021). In our work we didn't account for correlations between BNNs and assumed their independence. We can summarize the initialization of our inner BNNs parameters with

$$\pi(w_{ji}) = \mathcal{N}(w_{ji}|0, \sigma^2), \tag{6}$$

$$\pi(b_i) = \mathcal{N}(b_i|0, \sigma^2). \tag{7}$$

In Equations 6 and 7 w_{ji} and b_i represent any of the elements of $\boldsymbol{W}$ and $\boldsymbol{b}$, respectively.

We employed a non-informative multivariate Gaussian prior to incorporate the correlations between T_{eff}, t, and [Fe/H], whereas we assigned an independent prior for log g. The rationale behind this choice is that this parameter showed a low correlation with the other stellar parameters in the training set. This approach also lets us reduce the dimensionality of the hyperparameters D and Ω by one, improving our computational times.

In our HBM we did not consider the necessity of defining a joint correlation between the two groups of parameters, the CCs, and stellar ones. The

relationship between each CC and the stellar variables is the result of the BNNs.

The multivariate Gaussian distribution used as prior for the stellar parameters was centred on the mean of the observed values. The covariance matrix of T_{eff}, t, and [Fe/H] was disassembled into a scale matrix and a correlation matrix, a detailed exposition of which is available in Gelman and Hill (2006):

$$\Sigma = D \cdot \Omega \cdot D, \tag{8}$$

where D is a diagonal matrix with a scale for each stellar parameter, and Ω is the correlation matrix. Following Moya et al. (2022) we defined a Lewandowski-Kurowicka-Joe (LKJ) prior (Lewandowski et al., 2009) with shape parameter $\eta = 3$ for the correlation matrix, representing the equivalent to a uniform distribution on correlations. Finally, we defined a Cauchy prior on the scales centered at 0 with $\gamma = 2$. Once the posterior distributions of the true stellar parameters, $\boldsymbol{W}$ and $\boldsymbol{b}$ are inferred we can start with the age prediction stage.

3.3. Prediction stage

Let θ' be the set of true stellar parameters excluding the age (log g, T_{eff}, and [Fe/H]), and $\hat{\theta}'$ the vector of their observed values. The posterior samples of Σ obtained during the training phase were used to define the prior stellar parameters. These samples are employed to evaluate the likelihood term $p(\hat{c}, \hat{\theta}'|t, \theta', \boldsymbol{W}, \boldsymbol{b})$ to finally produce the distributions of t for the stars in the test set.

As previously outlined in this paper, our approach is designed to be as independent as possible from manual modelling. This is the primary reason for our investigation into the automation capabilities and data adaptability of ML. Consequently, we decided to employ BNNs, which have the potential to automatically capture intricate data patterns. In our case, we employed normal priors centered at 0 for our BNNs $\boldsymbol{W}$ and $\boldsymbol{b}$ parameters. While these priors depend on some parameter selection and subsequent fine-tuning experimentation, our model could potentially be used for other MS stars or even different stellar populations if we follow the same methodology for parameter adaptation.

In Section 4, we present experiments in which the training phase, implemented via MCMC sampling, generally required between 1.5 and 2 days. However, certain configurations, including

18

the best-performing one, extended the training duration to as much as 3.5 days. In contrast, the prediction time is primarily dictated by the number of samples used. For instance, processing 8,000 samples (organized into four chains of 2,000 samples each) for a single star typically takes about 20 minutes on the CPUs specified in subsection 3.1.1 (this time varies significantly on CPU models).

4. Results and discussion

In this section, we present a comprehensive analysis of the outcomes derived from the HBMs employed in our study. The culmination of extensive computations and model training is encapsulated in a series of succinct and informative tables and graphics presented in the following paragraphs. Herein, we provide an overview of the experimental setup and the results from the BNN architecture designs we used.

4.1. Experimental setup

In the following paragraphs, we introduce the distributions of the datasets (described in Section. 2), examine convergence during the inference stage, and outline the evaluation metrics for age prediction tasks. This approach enables a comprehensive evaluation of our architecture efficiency compared to established methodologies.

- **Distributions:** Figure 3 presents the distributions of stellar parameters from all stars used in this work (a total of 328 for the training set and 23 for the test). These figures exhibit remarkable similarities between the training and test sets, notwithstanding the relatively limited size of the test set.

- **Inference convergence:** We assess the convergence of the inference process using the NUTS-MCMC sampler heuristics (Hoffman and Gelman, 2011). This involves analyzing two key parameters: the Gelman-Rubin statistic ($\hat{R}$) and the effective sample size (N_{eff}) (Neal, 1993; Geyer, 1992). The $\hat{R}$ statistic measures the convergence of multiple chains, with values close to 1 indicating convergence. N_{eff} represents the effective number of independent samples and is used to ensure that a sufficient number of samples are obtained for accurate inference. Then the target values for these heuristics are a high N_{eff}, which indicates
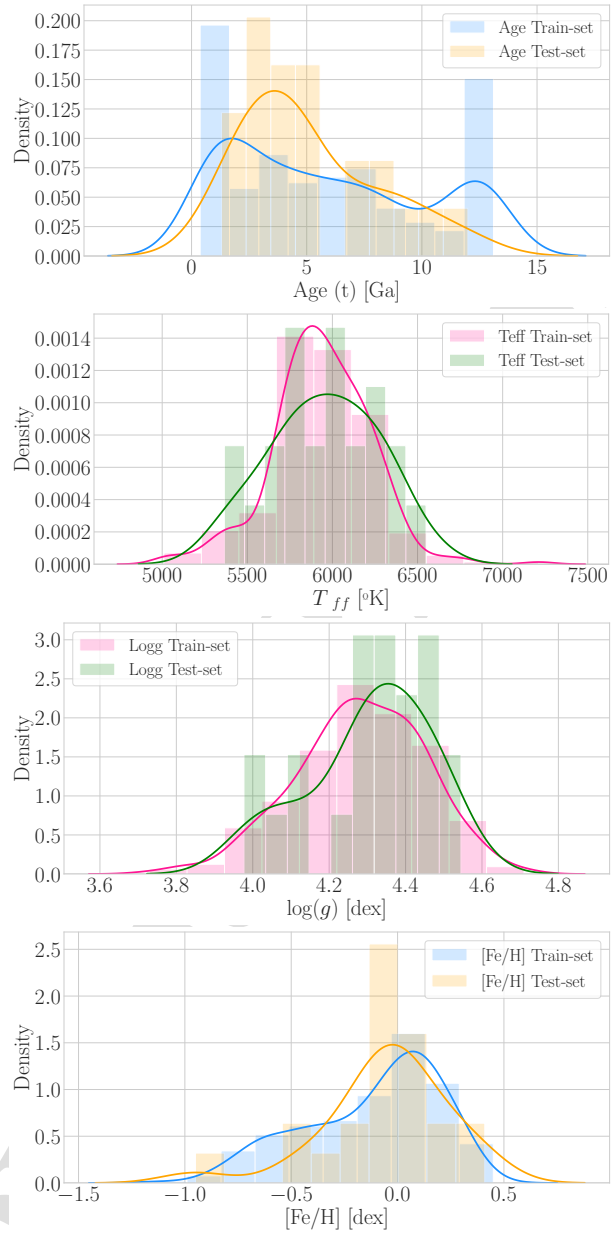
19

Figure 3: Distribution of the stellar parameters from the training and testing sets used in this work. *From Top to Bottom:* Age, effective temperature, surface gravity, and metallicity.

20

low autocorrelation and enhances the informativeness of the samples, and an $\hat{R}$ value close to one, which signifies well-mixed and converged samples, suggesting reliable parameter estimates.

In our analysis, we chose a strict convergence criterion, setting the $\hat{R}$ value threshold at 1.05. Additionally, if the $\hat{R}$ condition is satisfied, we also require that the effective sample size, N_{eff}, be at least 20% of the total number of samples (Gelman and Hill, 2006). Any N_{eff} below this threshold indicates insufficient effective sampling, leading to rejection of the results. However, a small subset of experiments that meet these criteria may still be unreliable. This situation arises because, in the result tables that follow, we present $\hat{R}$ as an average of the individual $\hat{R}$ values for model parameters. While this approach provides a general overview, it can obscure instances where specific $\hat{R}$ values deviate significantly from 1. To ensure quality, each set of results was carefully reviewed, and instances where this issue occurs are noted in the relevant sections.

- **Age prediction metrics:** For evaluating the performance of our model in predicting stellar ages in the test set, we use the mean absolute error (MAE) metric:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} \left| t_i - \hat{t}_i \right|, \tag{9}$$

  where n is the number of stars in the test set, t_i is the target value of the age for star i, and $\hat{t}_i$ is the predicted one. Additionally, we report the results as mean values of the posterior samples along with the standard deviation, representing the uncertainty in our predictions.

- **Baselines:** We compare the results obtained from our model with those reported by Moya et al. (2022) in their multilinear regression (MLR) approach. We provide an overview of their methodology and highlight any similarities or differences between their results and ours. This comparison serves as a benchmark for assessing the effectiveness of our model in relation to existing state-of-the-art approaches.

The experimentation aims to shed light on several key aspects related to BNNs applied to stellar characterization. In order to assess which architectural configurations yield the most accurate and reliable results, we

21

evaluate the performance of different BNN designs, exploring multiple network architectures (other experiments, presented in Appendix A, were run with single BNN configurations). This exploration offers valuable guidance on parameter-tuning strategies to enhance model performance and predictive accuracy.

Furthermore, the experimentation provides an assessment of the potential of BNNs to compete with classical approaches in this domain (e.g. Moya et al., 2022). By comparing the performance of BNNs against traditional methods our study aims to gauge the extent to which BNNs can replicate or improve upon the predictive capabilities of established techniques. This comparative analysis serves to showcase the versatility and effectiveness of BNNs in capturing complex patterns and relationships within the data, potentially surpassing the limitations of conventional approaches.

4.2. Multiple BNNs with single output

Our initial approach involved using a single neural network with six outputs. However, this approach encountered multimodality issues, illustrated in Figure A.8 and discussed in Appendix A. To address these issues and improve the generalization capabilities of the architecture, we adopted a more conservative strategy by using a separate neural network for each CC. This approach simplifies the relationships the networks need to learn and reduces the complexity of the parameter space.

The initial phase of model architecture exploration involved a stepwise progression in complexity, with detailed configurations outlined in Table 1. This exploration commenced by experimenting with a single hidden layer, varying the number of nodes (3, 6, 10, 15, 20, 25). For prior specification of W and b, the investigation began with narrow standard deviations centered at 0, progressively expanding their value until 1, informed by prior experimentation.

The effective sample size and $\hat{R}$ diagnostics are of insufficient quality with 3 nodes but improve significantly starting from 6 nodes onward. Generally in BNNs the parameter space typically grows in complexity as the model's architecture becomes more intricate (e.g. Valentin Jospin et al., 2020). However, using six neurons smooths the initially multimodal distribution of weights and biases observed in the three-node configuration, yielding a shape that more closely resembles a Gaussian distribution. This pattern persists across all our experiments involving architectures with more than three nodes, sug-

22

Nodes	$\pi\,[\sigma]$	MAE Age	$\hat{R}$	N_{eff}
3	0.25	0.99	1.156	3306
3	0.5	0.95	1.166	2042
3	1	0.87	1.838	1615
6	0.25	0.95	1.001	4960
6	0.5	0.85	1.003	2794
6	1	1.14	1.013	1711
10	0.25	0.92	1.000	8537
10	0.5	0.98	1.001	3942
10	1	1.17	1.001	3433
15	0.25	0.90	1.000	7180
20	0.25	0.89	1.000	5892
25	0.25	0.93	1.000	6955

Fixed parameters: 1 hidden layer NN, priors on data and chains centered on 0 (W and b), NUTS adapt delta parameter 0.95, NUTS max tree depth 13, 4 chains with 3000 samples (1000 as algorithm burn-in).

Table 1: Summary of experimental results and parameter configuration. *First column:* number of nodes of the NN architecture. *Second column:* standard deviations used to define the Gaussian priors for inner NN parameters. *Third column:* age mean absolute errors. *Fourth column:* $\hat{R}$ values for convergence diagnostics. *Fifth column:* effective sample sizes.

gesting that this level of complexity is sufficient for the parameters to converge into a monomodal distribution.

Although the exact mechanism behind this behavior is not fully understood, we hypothesize that it could be due to the small age uncertainties of the dataset. Also, the observed multimodality may re-emerge with significantly larger architectures (e.g. Valentin Jospin et al., 2020). In our case, the 6-node configuration appears to represent a threshold of complexity, where the number of nodes is sufficient to effectively map the parameter space. At this point, the nodes seem to specialize, focusing on individual modes to explain the data accurately while avoiding unnecessary complexity (e.g. Cybenko, 1989; Hornik, 1991; Csáji, 2001).

We used a wider prior of $\sigma = 1$ as a weaker prior belief about the values of the weights and biases, allowing the model to explore a broader range of parameter values. On the other hand, a narrower prior, like $\sigma = 0.25$ imposes

Nodes	π [σ]	MAE Age	$\hat{R}$	N_{eff}
3	0.25	1.23	1.070	3979
3	0.5	0.89	1.037	2117
3	1	1.22	1.153	1199
6	0.25	1.12	1.009	5316
6	0.5	0.93	1.011	6220
6	1	1.64	1.005	3445
10	0.25	1.12	1.003	7335

Fixed parameters: 2 hidden layers NN, priors on data and chains centered on 0 (W and b), NUTS adapt delta parameter 0.95, NUTS max tree depth 13, 4 chains with 3000 samples (1000 as algorithm burn-in).

Table 2: Summary of experimental results and parameter configuration. *First column:* number of nodes of the NN architecture. *Second column:* standard deviations used to define the Gaussian priors for inner NN parameters. *Third column:* age mean absolute errors. *Fourth column:* $\hat{R}$ values for convergence diagnostics. *Fifth column:* effective sample sizes.

stronger constraints on the parameters, effectively acting as a stronger form of regularisation (e.g. MacKay, 1992; Bishop, 2006) (having also a direct impact on computational times). Constraining BNNs parameters too tightly through narrow priors may overly restrict the posterior distribution, limiting its ability to explore meaningful regions of the parameter space. This can result in underestimating the true posterior uncertainty and missing important solutions or modes that could otherwise explain the data effectively. Further investigation into the influence of prior width revealed that restrictive standard deviations generally yield improved predictions. Using a value of 0.25 in the standard deviation for the priors yields very good heuristics in 10, 15, 20, and 25 nodes configurations, all of them with MAE ≈ 0.9. Configurations like the 6-nodes one, also give comparable predictions, but not that good heuristics.

Moreover, a prior initialization distant from the posterior can impede chain convergence, resulting in a diminished count of effective samples. Conversely, Table 1 presents the discussed results, predominantly indicating high-quality heuristics, suggesting that a greater number of nodes with precise and restrictive priors generally enhance predictive performance.

Subsequent experimentation delved deeper into the architecture of BNNs.

24

While deeper networks have the potential to capture more complex relationships in the data, they also introduce higher complexity and challenges related to inference, such as increased computational cost and difficulties in sampling from complex posterior distributions (e.g. Blundell et al., 2015). From these investigations, we observed that an increased number of layers does not necessarily yield improved outcomes, as shown in Table 2. Contrary to the insights gained from node variations in earlier experiments, heuristics deteriorated with higher layer counts, reaching a point where results became unreliable, being this one of the scenarios where some of the model $\hat{R}$ parameters significantly deviate from 1. Despite launching additional experiments with varied standard deviations for initialization or larger numbers of nodes, the errors remained non-comparable to those observed in the single hidden layer experiments. Moreover, as discussed in previous sections, our training dataset comprises only 328 stars, presenting an opportunity to leverage simpler models.

The outcomes derived from systematic experimentation revealed instances where certain internal parameters of the BNN posterior distribution deviated significantly from their prior initialization. Consequently, a customized and better-centered initialization method was devised to address this issue, results are summarized in Table 3. In this approach, a variational-based technique, ADVI (Automatic Differentiation Variational Inference) from PyStan, was employed to obtain an initial approximation of the mean values for our target parameters (W and b). In a subsequent and independent stage, these values were input into the MCMC-NUTS sampler as the mean value for the BNN priors, and a range of models was computed by varying the standard deviation width. The results were comparable to non-informative initializations, with some cases exhibiting worse performances. The primary advantage of this approach lay in achieving slightly improved computational times, though this aspect was not a focal parameter of interest in our study.

Building upon the insights gained from the prior experimental iterations, we sought to enhance the optimal results obtained with a single hidden layer comprising 6 nodes and employing restrictive prior initialization (see Table 1). The subsequent batches of experimentation were devised to systematically evaluate the impact of the number of samples and the maximum tree depth (MTD) of the NUTS on model performance.

Doubling the posterior sample size yielded a marginal improvement in the heuristics of the 6-node architecture, with no discernible adverse effects on MAEs. To systematically compare these outcomes, additional experiments

25

Nodes	$\pi\ [\sigma]$ (D, C)	MAE	$\hat{R}$	N_{eff}
3	0.25, 0.25	1.01	1.167	2347
3	0.5, 0.25	0.99	1.153	1464
6	0.25, 0.1	1.01	1.000	5429
6	0.5, 0.1	1.14	1.004	1919
6	0.25, 0.25	0.88	1.002	3726
6	0.5, 0.25	0.92	1.006	2235
6	0.25, 0.5	0.88	1.003	3256
6	0.5, 0.5	0.95	1.006	1856

Fixed parameters: 1 hidden layer NN, priors on data and ADVI centered on 0 (W and b), priors on NUTS chains centered on ADVI posteriors (W and b), NUTS adapt delta parameter 0.95, NUTS max tree depth 13, 4 chains with 3000 samples (1000 as algorithm burn-in).

Table 3: Summary of experimental results and parameter configuration. *First column:* number of nodes of the NN architecture. *Second column:* standard deviations used to define the Gaussian priors for inner NN parameters (D: Data and ADVI initialization, C: Chains). *Third column:* age mean absolute errors. *Fourth column:* $\hat{R}$ values for convergence diagnostics. *Fifth column:* effective sample sizes.

were conducted with a shallower tree depth (for NUTS), results are summarized in Table 4, to analyze the impact of the MTD. With an MTD set to 10, the sampler reached 100% sample saturation, indicating that while achieving favorable heuristics and results, the sampler faced more stringent requirements for realistic parameter space exploration. Altering this parameter to 12 partially resolved the saturation issue (some of the inner parameters of our architecture need deeper MTD), but the results exhibited a decline in quality. This set of experiments suggests that a shallower MTD may be employed in tandem with extended training to alleviate the impact of less accurate exploration. Even though time is not the most significant constraint in our study, this could represent a strong limitation for larger datasets. Conversely, elevating the MTD did not lead to a direct enhancement in MAEs but positively influenced heuristics by increasing the count of effective samples. Higher tree depths improve posterior exploration, particularly for complex distributions, but exponentially increase the computational cost per iteration.

Increasing sample sizes is useful if the correct MTD ensures the sampler can properly explore the posterior. However, once the posterior is well-

Samples	MTD	MAE Age	$\hat{R}$	N_{eff}
3000 (1000)	10	0.92	1.010	827
3000 (1000)	12	1.65	1.011	2334
6000 (2000)	10	1.08	1.014	1130
6000 (2000)	12	1.02	1.001	5474
6000 (2000)	13	0.88	1.005	3487
6000 (2000)	14	1.04	1.004	6741

Fixed parameters: 1 hidden layer with 6 nodes NN, priors on data and chains centered on 0 ($\boldsymbol{W}$ and $\boldsymbol{b}$), NUTS adapt delta parameter 0.95, 4 chains with 3000 samples (1000 as algorithm burn-in), Gaussian priors for inner NN parameters are fixed to $\mu = 0$ and $\sigma = 0.5$.

Table 4: Summary of experimental results and parameter configuration. *First column:* number of nodes of the NN architecture. *Second column:* Max tree depth NUTS algorithm parameter. *Third column:* age mean absolute errors. *Fourth column:* $\hat{R}$ values for convergence diagnostics. *Fifth column:* effective sample sizes.

sampled and diagnostics indicate convergence, additional samples may lead to little improvement or could even worsen the results. Based on this insight, we chose to extend our investigation of the MTD. We focused on configurations with the most promising heuristics from Table 1 and expanded the MTD to 14. Our analysis reveals an overall enhancement in heuristics, particularly beneficial for configurations comprising only 6 nodes. These findings are summarized in Table 5, where it is evident that the majority of MAE values approximate 0.9. This underscores the potential for achieving a balance between architectural complexity, outcomes, and computational efficiency with simpler architectures.

As concluding experiments we systematically investigate the impact of the parameter adapt delta (AD) of the NUTS-HMC algorithm, a critical variable influencing the acceptance ratio of target samples. A higher AD results in a more conservative step size in the sampler, thereby reducing the likelihood of numerical errors during the integration process. Consequently, the sampler takes smaller steps, potentially enhancing the robustness and accuracy of the samples at the expense of increased computational requirements per effective sample. The data depicted in Table 6 closely align with those observed in Table 5. Also, the quality of our heuristics improved due to the adoption of a more restrictive sampling approach, ensuring a balanced acceptance rate

Nodes	$\pi \, [\, \sigma \,]$	MAE Age	$\hat{R}$	N_{eff}
6	0.25	0.90	1.000	5916
6	0.5	0.93	1.003	2940
8	0.25	0.96	1.002	5709
8	0.5	0.91	1.002	3119
10	0.25	0.98	1.001	7260
15	0.25	0.96	1.000	9300
20	0.25	0.91	1.000	6618
25	0.25	0.93	1.000	7575

Fixed parameters: 1 hidden layer NN, priors on data and chains centered on 0 (W and b), NUTS adapt delta parameter 0.95, NUTS max tree depth 14, 4 chains with 3000 samples (1000 as algorithm burn-in).

Table 5: Summary of experimental results and parameter configuration. *First column:* number of nodes of the NN architecture. *Second column:* standard deviations used to define the Gaussian priors for inner NN parameters. *Third column:* age mean absolute errors. *Fourth column:* $\hat{R}$ values for convergence diagnostics. *Fifth column:* effective sample sizes.

that enhances both efficiency and reliability in posterior exploration.

In Table 7, we amalgamated the insights garnered from the current experimentation, incorporating longer sampling durations, deeper MTD exploration, and more restrictive AD settings, particularly targeting configurations comprising 6 nodes with varying prior ranges. This batch yielded some of the most promising heuristics. As a succinct conclusion drawn from our current experimentation, we present the following observations:

- *Architectures and nodes*: Higher layers didn't always improve outcomes due to increased complexity and inference challenges. Single-layer architectures with a higher count of neurons can simplify the parameter space and still capture complex patterns for our problem.

- *Impact of prior width*: Wider priors allowed broader exploration, while narrower priors acted as stronger regularization. Restrictive standard deviations generally improve predictions but seem to increase sensitivity in heuristics.

- *Initialization*: Distant initialization hindered convergence, while pre-

28

Nodes	$\pi\,[\,\sigma\,]$	MAE Age	$\hat{R}$	N_{eff}
6	0.25	0.83	1.000	6411
6	0.5	0.92	1.002	3104
10	0.25	1.05	1.000	7527
15	0.25	1.00	1.000	7931
20	0.25	0.86	1.000	7623

Fixed parameters: 1 hidden layer NN, priors on data and chains centered on 0 ($\boldsymbol{W}$ and $\boldsymbol{b}$), NUTS adapt delta parameter 0.99, NUTS max tree depth 13, 4 chains with 3000 samples (1000 as algorithm burn-in).

Table 6: Summary of experimental results and parameter configuration. *First column:* number of nodes of the NN architecture. *Second column:* standard deviations used to define the Gaussian priors for inner NN parameters. *Third column:* age mean absolute errors. *Fourth column:* $\hat{R}$ values for convergence diagnostics. *Fifth column:* effective sample sizes.

cise priors enhanced performance. A VI-based initialization technique, meaning a better centering for our case, slightly reduced computational times during our experimentation.

- *Sample size and MTD*: Doubling samples slightly improved heuristics, while higher tree depth increased effective samples.

- *Acceptance Denial parameter*: Higher values restricted acceptance, affecting sample acceptance ratio and the robustness of the sampling, at the cost of increased computational times.

4.3. Best architecture and comparison with the state of the art

In this section, we compare the performance of our best model architecture with the results reported in Moya et al. (2022), providing a rationale for our design choices. Based on insights from prior experiments and the outcomes summarized in Table 7, we identified $\sigma = 0.25$ and $\sigma = 0.5$ as potential optimal parameters. To ensure robust statistical validation, we conducted the prediction stage 10 independent times, allowing us to evaluate the consistency and reliability of the results.

In summary, the optimal architecture identified consists of a single HL with six nodes (utilizing six BNNs, each corresponding to a specific CC). The

Nodes	$\pi\,[\,\sigma\,]$	MAE Age	$\hat{R}$	N_{eff}
6	0.1	1.25	1.000	24061
6	0.25	1.05	1.001	8802
6	0.5	0.90	1.002	5873

Fixed parameters: 1 hidden layer NN, priors on data and chains centered on 0 ($\boldsymbol{W}$ and $\boldsymbol{b}$), NUTS adapt delta parameter 0.99, NUTS max tree depth 14, 4 chains with 6000 samples (2000 as algorithm burn-in).

Table 7: Summary of experimental results and parameter configuration. *First column:* number of nodes of the NN architecture. *Second column:* standard deviations used to define the Gaussian priors for inner NN parameters. *Third column:* age mean absolute errors. *Fourth column:* $\hat{R}$ values for convergence diagnostics. *Fifth column:* effective sample sizes.

parameters yielding the best results are presented in Table 8 and illustrated in Figure 5. Even though $\sigma = 0.25$ tended to achieve better results during our experimentation, we selected $\sigma = 0.5$ as the prior initialization width for the winning architecture, as it provides the best balance among computational cost, robustness, simplicity, heuristics, and performance outcomes. This was achieved by using 6000 samples per chain, where 2000 samples were allocated for burn-in and preconditioning of the sampler (with a total of 4 chains for parameter space exploration). The parameters MTD and AD, which are crucial for sampling, were set to 14 and 0.99, respectively. The winning configuration parameters are summarized in Table 9.

This configuration achieved a MAE of 0.92 after ten training runs. The heuristics (N_{eff} and $\hat{R}$) for this architecture are plotted in Figure 5, showing an average $\hat{R}$ value of 1.003 with approximately 6500 effective samples per parameter. The final predictions of this architecture are shown in Figure 4, which compares our results with the state-of-the-art HBM-MLR detailed in Moya et al. (2022) for the same test set. Our results are comparable to theirs, and our predictions align closely with their findings. Notably, for most predictions, we achieved small uncertainties compared to their mean value, with only one of the 23 stars exhibiting a larger uncertainty. This discrepancy could be due to the chemical peculiarities of that star or poor representation in the training set, which may have biased our prediction.

It is important to note that comparable results were achievable with less restrictive parameters and the use of wider neural networks, which also

Nodes	$\pi\,[\,\sigma\,]$	MAE Age	$\hat{R}$	N_{eff}
6	0.25	0.91	1.005	6927
6	0.5	0.92	1.003	6531

Fixed parameters: 1 hidden layer NN, priors on data and chains centered on 0 ($\boldsymbol{W}$ and $\boldsymbol{b}$), NUTS adapt delta parameter 0.99, NUTS max tree depth 14, 4 chains with 6000 samples (2000 as algorithm burn-in).

Table 8: Summary of final experimental results and parameter configuration. Best designs from Table 7 (second and third row) [Average of 10 prediction cicles]. *First column:* number of nodes of the NN architecture. *Second column:* standard deviations used to define the Gaussian priors for inner NN parameters. *Third column:* age mean absolute errors. *Fourth column:* $\hat{R}$ values for convergence diagnostics. *Fifth column:* effective sample sizes.

Parameter	Value
HLs	1
Nodes	6
MTD	14
AD	0.99
μ and σ priors	[0, 0.5]
Samples	6000 (2000)
Chains	4
$\hat{R}$	1.003
MAE Age	0.92

Table 9: Summary of the parameter configuration for the best BNN architecture obtained during the experimentation.

yielded good outcomes. However, we demonstrated that with appropriate fine-tuning, good results could be achieved using simpler architectures.

In this work, we present an approach for stellar dating that achieves results comparable to those reported in previous studies (Moya et al., 2022). The primary contribution of our approach lies in its ability to leverage neural networks to model complex relationships within the dataset without the need for manually designed empirical relations. This flexibility not only simplifies the modeling process but also reduces the risk of oversimplification inherent in manual modeling, which could otherwise lead to suboptimal models and unreliable uncertainty estimates. Furthermore, our model demonstrates versatility and adaptability, making it suitable for application to a wide range of stellar types and astrophysical problems. For instance, it can be employed to deepen our understanding of the evolution of stellar ages as a function of galactocentric radius within the Milky Way (Viscasillas Vázquez et al., 2022). Additionally, the model could be used for advancing our knowledge of the intricate relationships between stellar parameters in the context of magneto-gyrochronology, potentially refining the methods used to estimate stellar ages based on rotational and magnetic properties (e.g. Mathur et al., 2023b). This applicability underscores the model's potential as a powerful tool for addressing fundamental questions in stellar astrophysics.

5. Summary and conclusions

The outcomes presented in this paper affirm the successful formulation of an effective and robust methodology for uncertainty quantification, specifically applied in the context of stellar dating. Optimal results were attained through the adoption of single-layer architectures. While it is feasible to achieve comparable outcomes using an HBM employing a solitary BNN with six outputs (corresponding to each CC), such models often necessitate the utilization of deeper and wider neural networks, thereby having the potential to exhibit pronounced multi-modality. While the incorporation of multi-modal samplers could potentially address this challenge, our choice was to focus on the reliability and versatility of the well-established Stan language, which provides robust sampling techniques while allowing for customization and adaptability.

As illustrated in Figure 6, the posterior distributions of our BNNs consistently exhibit Gaussian-like shapes, suggesting the potential utilization of VI samplers, such as ADVI, in scenarios prioritizing computational speed. While
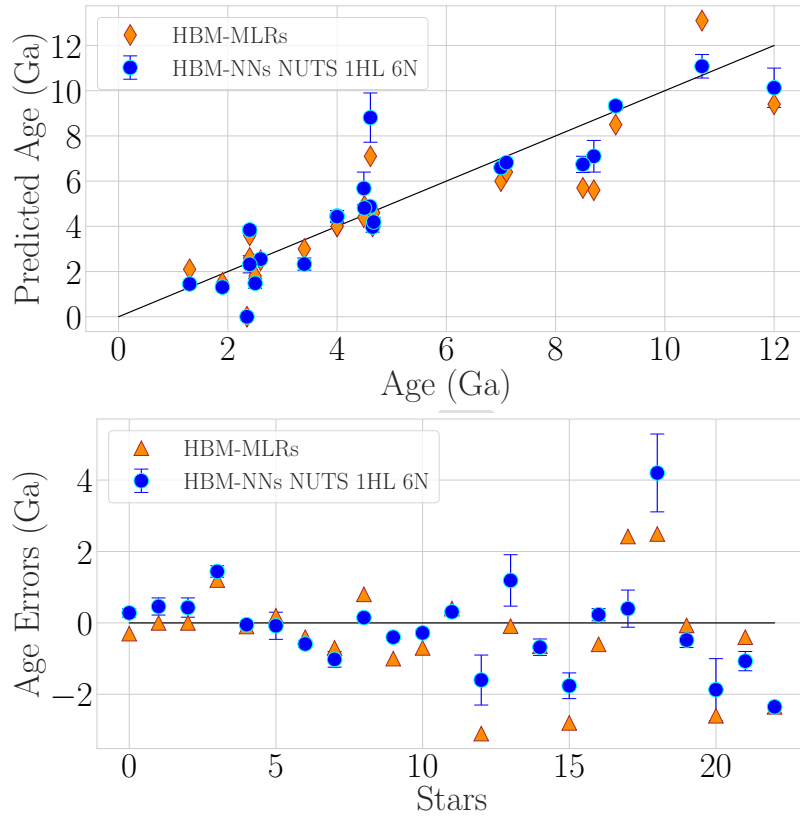
32

Figure 4: Hierarchical Bayesian model result comparison. (*HBM-MLRs:*) architecture whose probability relationships are modeled by multi-linear regressions, from Moya et al. (2022). (*HBM-NNs NUTS 1HL 6N:*) **[Best]** architecture whose probability relationships are modeled by NNs. The architecture consists of 1 Hidden Layer with 6 nodes and single-output (x6 NNs), sampled with MCMC-NUTS algorithm. (*Top*:) Age prediction for the 23 test stars. (*Bottom*:) Age errors for the 23 test stars.
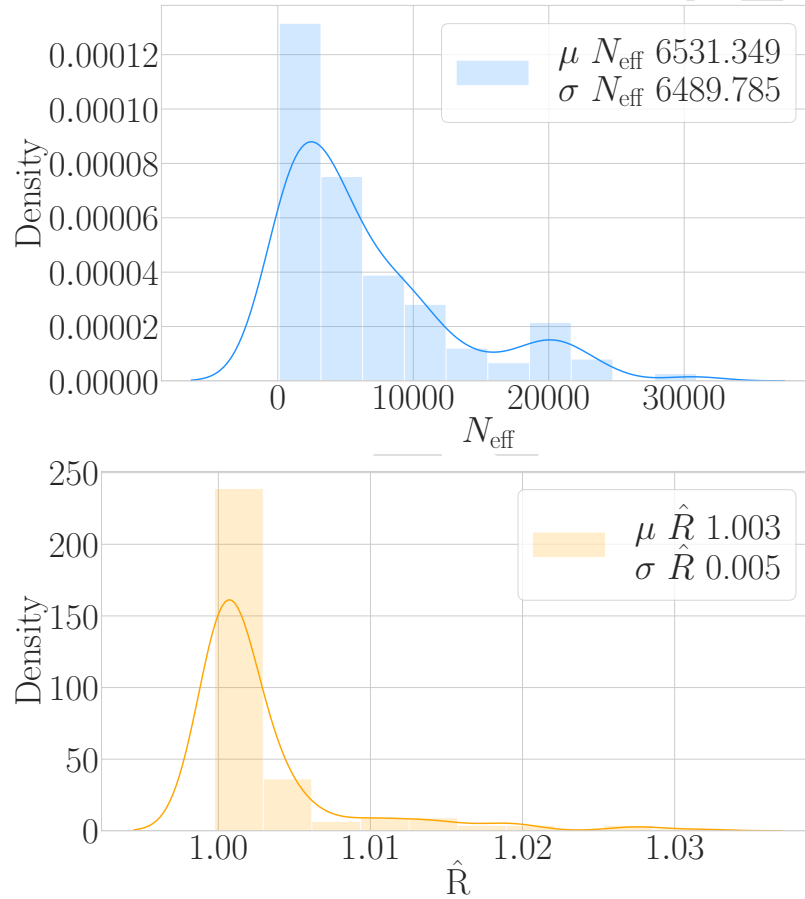
33

Figure 5: From the best experiment configuration of the work, Table 9. *Top:* Distribution of the probabilities of the MCMC-NUTS sampling heuristic of convergence N_{eff} (Number of effective samples). *Bottom:* Distribution of the probabilities of the MCMC-NUTS sampling heuristic of convergence $\hat{R}$ (Gelman-Rubin statistic).
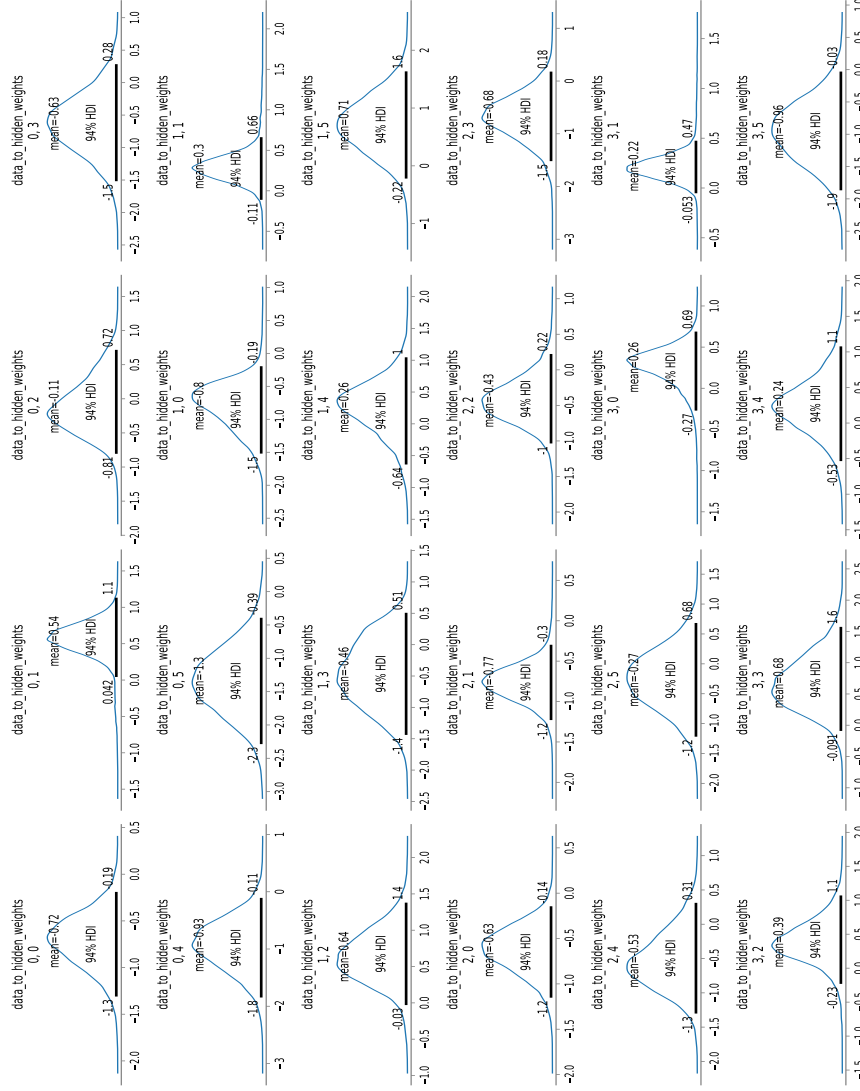
Figure 6: Example of posterior distributions for a weight matrix in one hidden layer, six nodes neural network applied to a model with a single output, single CC (in this case $[Mg/Y]$) prediction, (for better illustration, see figure 2). The figure shows the distributions of each weight (W_{ji}) in the initial weight matrix, which represents the connections between the input data and the hidden layer for the best experiment configuration (second row of Table 8). Each distribution includes its mean and the 94% Highest Density Interval (HDI). The distributions are organized from left to right and top to bottom, corresponding to each of the four stellar parameters and their connections to the six nodes in the hidden layer.

35

the results obtained with VI algorithms are comparable during experimentation (mainly in multi-output NN) and, in some architectures, improvement, we deliberately chose MCMC approaches in this study to prioritize a more conservative approach. This preference is reflected in the selection of an MTD sufficiently high to avoid chain saturation and restrictive ADs (e.g. 0.95 or 0.99). Conversely, the number of samples, set at 3000, proved to be adequate, with a marginal enhancement in heuristics observed with additional samples, particularly advantageous with a lower node count.

Our experimental findings underscore the effectiveness of employing generic initializations for BNN parameters in our context. Assumptions such as Gaussian distributions and centering priors at zero were made during the selection process. These assumptions facilitated the exploration of optimal standard deviations. Generally, initializing priors with small standard deviations (0.25 or less), close to zero, yielded improved heuristics. This effect was more pronounced with a higher node count.

In summary, single hidden layer architectures (with single output) with a node count ranging between 6-25 (optimal heuristics/results ratio at 6 nodes, superior heuristics with higher counts), coupled with stringent restrictions on the prior initialization of weights and biases (standard deviation 0.25-0.5), MTD set at 13-14, and AD at 0.99, led to our most successful outcomes. This configuration enabled the presentation of a statistical-machine learning hybrid model that achieved a MAE of less than 1 Ga in our testing sample. We present a machine learning-based tool for stellar dating that achieves results comparable to prior studies while eliminating the need for manually designed empirical relationships. This approach streamlines modelling, reduces oversimplification risks, and ensures robust predictions with reliable uncertainty estimates. Its versatility also allows us to apply this approach to various datasets, including different stellar types, opening an opportunity for improvement in this, and potentially other fields.

6. Future lines

Future work will focus on leveraging the developed algorithms to investigate the distribution of stellar ages as a function of galactocentric radius. This analysis holds the promise of providing valuable insights into fundamental processes of galactic evolution, such as the Milky Way's formation history and the role of radial migration (e.g. Viscasillas Vázquez et al., 2022). By integrating precise stellar age estimates with spatial distribution data, this

36

extension aims to deliver a comprehensive characterization of age variation across the galaxy.

Furthermore, we plan to explore the integration of gyrochronology and magnetic activity indices for stellar age determination. Building on recent work (Mathur et al., 2023b), which first combined these indices through linear relationships. While this approach demonstrated promise, it may also introduce errors and biases due to its reliance on simplistic models. By employing our more flexible framework, we aim to mitigate these limitations and develop a robust methodology for combining gyrochronology and magnetic activity indices in a unified age estimation model.

This approach will not only refine stellar age determinations but also create a versatile toolkit for stellar dating, encompassing a range of complementary techniques. Such advancements are expected to enhance our understanding of the evolution of stellar populations and the dynamic history of the Milky Way.

On the other hand, future research in BNNs and uncertainty quantification remains focused on addressing the persistent challenge of multimodality (Wiese et al., 2023). Enhancing inference methods to better capture and navigate multimodal distributions is a crucial aspect of this ongoing investigation. The current target aims to develop innovative algorithms that can efficiently handle the complex parameter spaces inherent in neural networks, seeking a balance between computational efficiency and the accurate representation of uncertainty (e.g. Liu and Wang, 2016; Wilson and Izmailov, 2020; Durasov et al., 2020). Furthermore, there is a notable emphasis on integrating domain knowledge and expert insights into the Bayesian framework, providing more informed priors that can contribute to robust and interpretable uncertainty estimates (e.g. Fortuin, 2021). Besides that, extending the application of BNNs to diverse domains is a key frontier.

In addition, establishing standardized benchmarks and improving current evaluation metrics will be crucial for evaluating the performance of BNNs across different applications, to ensure that progress in the field is measurable and reproducible. In summary, the dynamic landscape of BNN research still revolves around tackling multimodality, refining inference methods for better scalability, and broadening the applicability of these models to address complex real-world challenges.

Acknowledgements

References

Adibekyan, V.Z., Santos, N.C., Sousa, S.G., Israelian, G., 2011. A new α-enhanced super-solar metallicity population. Astronomy & Astrophysics 535, L11. doi:10.1051/0004-6361/201118240, arXiv:1111.4936.

Adibekyan, V.Z., Sousa, S.G., Santos, N.C., Delgado Mena, E., González Hernández, J.I., Israelian, G., Mayor, M., Khachatryan, G., 2012. Chemical abundances of 1111 FGK stars from the HARPS GTO planet search program. Galactic stellar populations and planets. Astronomy & Astrophysics 545, A32. doi:10.1051/0004-6361/201219401, arXiv:1207.2388.

Aguirre, V.S., Lund, M.N., Antia, H.M., Ball, W.H., Basu, S., Christensen-Dalsgaard, J., Lebreton, Y., Reese, D.R., Verma, K., Casagrande, L., Justesen, A.B., Mosumgaard, J.R., Chaplin, W.J., Bedding, T.R., Davies, G.R., Handberg, R., Houdek, G., Huber, D., Kjeldsen, H., Latham, D.W., White, T.R., Coelho, H.R., Miglio, A., Rendle, B., 2017. Standing on the shoulders of dwarfs: the kepler asteroseismic legacy sample. ii. radii, masses, and ages. The Astrophysical Journal 835, 173. URL: https://dx.doi.org/10.3847/1538-4357/835/2/173, doi:10.3847/1538-4357/835/2/173.

Almeida-Fernandes, F., Rocha-Pinto, H.J., 2018. A method to estimate stellar ages from kinematical data. Monthly Notices of the Royal Astronomical Society 476, 184–197. doi:10.1093/mnras/sty119, arXiv:1801.04046.

Ancona, M., Ceolini, E., Öztireli, C., Gross, M., 2017. Towards better understanding of gradient-based attribution methods for Deep Neural Networks. arXiv e-prints , arXiv:1711.06104doi:10.48550/arXiv.1711.06104, arXiv:1711.06104.

38

Angus, R., Aigrain, S., Foreman-Mackey, D., McQuillan, A., 2015. Calibrating gyrochronology using kepler asteroseismic targets. Monthly Notices of the Royal Astronomical Society 450, 1787–1798. URL: http://dx.doi.org/10.1093/mnras/stv423, doi:10.1093/mnras/stv423.

Angus, R., Morton, T.D., Brewer, J., Bedell, M., Foreman-Mackey, D., Curtis, J., Hogg, D.W., 2019. An age-dating method that combines stellar evolution models with gyrochronology, in: American Astronomical Society Meeting Abstracts #233, p. 235.09.

Arbel, J., Pitas, K., Vladimirova, M., Fortuin, V., 2023. A primer on bayesian neural networks: Review and debates. arXiv:2309.16314.

Barnes, S.A., 2003. On the Rotational Evolution of Solar- and Late-Type Stars, Its Magnetic Origins, and the Possibility of Stellar Gyrochronology. The Astronomical Journal 586, 464–479. doi:10.1086/367639, arXiv:astro-ph/0303631.

Basu, V., 2020. Prediction of stellar age with the help of extra-trees regressor in machine learning, in: Proceedings of the International Conference on Innovative Computing & Communications (ICICC) 2020, p. . Available at SSRN: https://ssrn.com/abstract=3563397 or http://dx.doi.org/10.2139/ssrn.3563397.

Bishop, C.M., 2006. Pattern Recognition and Machine Learning. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D., 2015. Weight Uncertainty in Neural Networks. arXiv e-prints , arXiv:1505.05424doi:10.48550/arXiv.1505.05424, arXiv:1505.05424.

Brandt, T.D., Huang, C.X., 2015. Bayesian Ages for Early-type Stars from Isochrones Including Rotation, and a Possible Old Age for the Hyades. The Astronomical Journal 807, 58. doi:10.1088/0004-637X/807/1/58, arXiv:1501.04404.

Bressan, A., Marigo, P., Girardi, L., Salasnich, B., Dal Cero, C., Rubele, S., Nanni, A., 2012. PARSEC: stellar tracks and isochrones with the PAdova and TRieste Stellar Evolution Code. Monthly Notices of the Royal Astronomical Society 427, 127–145. doi:10.1111/j.1365-2966.2012.21948.x, arXiv:1208.4498.

Bu, Y., Yerra, B.k., Xie, J., Pan, J., Gang, Z., Wu, Y., 2020. Estimation of stellar ages and masses using gaussian process regression. The Astrophysical Journal Supplement Series 249, 7. doi:10.3847/1538-4365/ab8bcd.

Carpenter, B., Gelman, A., Hoffman, M.D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., Riddell, A., 2017. Stan: A probabilistic programming language. Journal of Statistical Software 76, 1–32. URL: https://doi.org/10.18637/jss.v076.i01, doi:10.18637/jss.v076.i01.

Ciucă, I., Kawata, D., Miglio, A., Davies, G.R., Grand, R.J.J., 2021. Unveiling the distinct formation pathways of the inner and outer discs of the Milky Way with Bayesian Machine Learning. Monthly Notices of the Royal Astronomical Society 503, 2814–2824. doi:10.1093/mnras/stab639, arXiv:2003.03316.

Csáji, B.C., 2001. Approximation with artificial neural networks. Ph.D. thesis. Faculty of Sciences, Eötvös Loránd University, Hungary.

Cybenko, G., 1989. Approximations by superpositions of sigmoidal functions. Mathematics of Control, Signals, and Systems 2, 303–314. doi:10.1007/BF02551274.

da Silva, L., Girardi, L., Pasquini, L., Setiawan, J., von der Lühe, O., de Medeiros, J.R., Hatzes, A., Döllinger, M.P., Weiss, A., 2006. Basic physical parameters of a selected sample of evolved stars. Astronomy & Astrophysics 458, 609–623. doi:10.1051/0004-6361:20065105, arXiv:astro-ph/0608160.

da Silva, R., Porto de Mello, G.F., Milone, A.C., da Silva, L., Ribeiro, L.S., Rocha-Pinto, H.J., 2012. Accurate and homogeneous abundance patterns in solar-type stars of the solar neighbourhood: a chemo-chronological analysis. Astronomy & Astrophysics 542, A84. doi:10.1051/0004-6361/201118751, arXiv:1204.4433.

Davidian, M., 2003. Hierarchical linear models: Applications and data analysis methods (2nd ed.). stephen w. raudenbush and anthony s. bryk. Journal of the American Statistical Association 98, 767–768. doi:10.2307/30045307.

Delgado Mena, E., Moya, A., Adibekyan, V., Tsantaki, M., Gonzalez Hernandez, J.I., Israelian, G., Davies, G.R., Chaplin, W.J., Sousa, S.G., Ferreira, A., Santos, N.C., 2019a. VizieR Online Data Catalog: Masses and ages of 1059 HARPS-GTO stars (Delgado Mena+, 2019). å doi:10.26093/cds/vizier.36240078.

Delgado Mena, E., Moya, A., Adibekyan, V., Tsantaki, M., González Hernández, J.I., Israelian, G., Davies, G.R., Chaplin, W.J., Sousa, S.G., Ferreira, A.C.S., Santos, N.C., 2019b. Abundance to age ratios in the HARPS-GTO sample with Gaia DR2. Chemical clocks for a range of [Fe/H]. Astronomy & Astrophysics 624, A78. doi:10.1051/0004-6361/201834783, arXiv:1902.02127.

Delgado Mena, E., Tsantaki, M., Adibekyan, V.Z., Sousa, S.G., Santos, N.C., González Hernández, J.I., Israelian, G., 2017. Chemical abundances of 1111 FGK stars from the HARPS GTO planet search program. II. Cu, Zn, Sr, Y, Zr, Ba, Ce, Nd, and Eu. Astronomy & Astrophysics 606, A94. doi:10.1051/0004-6361/201730535, arXiv:1705.04349.

Dong, Y., Su, H., Zhu, J., Bao, F., 2017. Towards Interpretable Deep Neural Networks by Leveraging Adversarial Examples. arXiv e-prints , arXiv:1708.05493doi:10.48550/arXiv.1708.05493, arXiv:1708.05493.

Dotter, A., Conroy, C., Cargile, P., Asplund, M., 2017. The influence of atomic diffusion on stellar ages and chemical tagging. The Astrophysical Journal 840, 99. URL: https://dx.doi.org/10.3847/1538-4357/aa6d10, doi:10.3847/1538-4357/aa6d10.

Durasov, N., Bagautdinov, T., Baque, P., Fua, P., 2020. Masksembles for Uncertainty Estimation. arXiv e-prints , arXiv:2012.08334doi:10.48550/arXiv.2012.08334, arXiv:2012.08334.

Fortuin, V., 2021. Priors in Bayesian Deep Learning: A Review. arXiv e-prints , arXiv:2105.06868doi:10.48550/arXiv.2105.06868, arXiv:2105.06868.

Gal, Y., Ghahramani, Z., 2015a. Bayesian Convolutional Neural Networks with Bernoulli Approximate Variational Inference. arXiv e-prints , arXiv:1506.02158doi:10.48550/arXiv.1506.02158, arXiv:1506.02158.

41

Gal, Y., Ghahramani, Z., 2015b. Dropout as a Bayesian Approxima-
tion: Representing Model Uncertainty in Deep Learning. arXiv e-prints ,
arXiv:1506.02142doi:10.48550/arXiv.1506.02142, arXiv:1506.02142.

Gavel, A., Gruyters, P., Heiter, U., Korn, A.J., Nordlander, T., Scheutwinkel,
K.H., Richard, O.A., 2021. Atomic diffusion and mixing in old stars. VII.
Abundances of Mg, Ti, and Fe in M 30. Astronomy & Astrophysics 652,
A75. doi:10.1051/0004-6361/202140770, arXiv:2110.12391.

Gelman, A., Carlin, J.B., Stern, H.S., Rubin, D.B., 2004. Bayesian Data
Analysis. 2nd ed. ed., Chapman and Hall/CRC.

Gelman, A., Hill, J., 2006. Data Analysis Using Regression And Multi-
level/Hierarchical Models. volume 3. doi:10.1017/CBO9780511790942.

Gelman, A., Lee, D., Guo, J., 2015. Stan: A probabilistic program-
ming language for bayesian inference and optimization. Journal of
Educational and Behavioral Statistics 40, 530–543. URL: https://
doi.org/10.3102/1076998615606113, doi:10.3102/1076998615606113,
arXiv:https://doi.org/10.3102/1076998615606113.

Gelman, A., Rubin, D.B., 1992. Inference from Iterative Simulation Using
Multiple Sequences. Statistical Science 7, 457 – 472. URL: https://doi.
org/10.1214/ss/1177011136, doi:10.1214/ss/1177011136.

Geyer, C.J., 1992. Practical Markov Chain Monte Carlo. Statistical Science
7, 473 – 483. URL: https://doi.org/10.1214/ss/1177011137, doi:10.
1214/ss/1177011137.

Goupil, M.J., Catala, C., Samadi, R., Belkacem, K., Ouazzani, R.M., Reese,
D.R., Appourchaux, T., Mathur, S., Cabrera, J., Börner, A., Paproth, C.,
Moedas, N., Verma, K., Lebreton, Y., Deal, M., Ballot, J., Chaplin, W.J.,
Christensen-Dalsgaard, J., Cunha, M., Lanza, A.F., Miglio, A., Morel, T.,
Serenelli, A., Mosser, B., Creevey, O., Moya, A., Garcia, R.A., Nielsen,
M.B., Hatt, E., 2024. Predicted asteroseismic detection yield for solar-like
oscillating stars with PLATO. arXiv e-prints , arXiv:2401.07984doi:10.
48550/arXiv.2401.07984, arXiv:2401.07984.

Guédé, C., Lebreton, Y., Babusiaux, C., Haywood, M., 2012. Age dating
large samples of stars: Ways toward improved accuracy, in: Boissier, S.,

de Laverny, P., Nardetto, N., Samadi, R., Valls-Gabaud, D., Wozniak, H. (Eds.), SF2A-2012: Proceedings of the Annual meeting of the French Society of Astronomy and Astrophysics, pp. 195–198.

Hoffman, M.D., Gelman, A., 2011. The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. arXiv e-prints , arXiv:1111.4246doi:10.48550/arXiv.1111.4246, arXiv:1111.4246.

Hornik, K., 1991. Approximation capabilities of multilayer feedforward networks. Neural Networks 4, 251–257. doi:10.1016/0893-6080(91)90009-T.

Izmailov, P., Vikram, S., Hoffman, M.D., Wilson, A.G., 2021. What Are Bayesian Neural Network Posteriors Really Like? arXiv e-prints , arXiv:2104.14421doi:10.48550/arXiv.2104.14421, arXiv:2104.14421.

Jeffery, E.J., von Hippel, T., van Dyk, D.A., Stenning, D.C., Robinson, E., Stein, N., Jefferys, W.H., 2016. A Bayesian Analysis of the Ages of Four Open Clusters. The Astronomical Journal 828, 79. doi:10.3847/0004-637X/828/2/79, arXiv:1611.00835.

Kiman, Faherty, Cruz, Gagné, Angus, Schmidt, Mann, Bardalez Gagliuffi, Rice, 2021. Age-relations for low-mass stars, in: The 20.5th Cambridge Workshop on Cool Stars, Stellar Systems, and the Sun (CS20.5), p. 264. doi:10.5281/zenodo.4567649.

Kiman, R., Angus, R., Xu, S., Schmidt, S., Gagné, J., Cruz, K., Faherty, J., Rice, E., 2020. Age-Activity relation for M dwarfs using Hα equivalent widths, in: American Astronomical Society Meeting Abstracts #235, p. 274.16.

Kiman, R., Xu, S., Faherty, J.K., Gagné, J., Angus, R., Brandt, T.D., Casewell, S.L., Cruz, K.L., 2022. wdwarfdate: A Python Package to Derive Bayesian Ages of White Dwarfs. The Astronomical Journal 164, 62. doi:10.3847/1538-3881/ac7788, arXiv:2206.05388.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., Hadsell, R., 2017. Overcoming catastrophic forgetting in neural networks. Proceedings of the National

43

Academy of Science 114, 3521–3526. doi:10.1073/pnas.1611835114, arXiv:1612.00796.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet Classification with Deep Convolutional Neural Networks. volume 25. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Kurucz, R., 1993. ATLAS9 Stellar Atmosphere Programs and 2 km/s grid. ATLAS9 Stellar Atmosphere Programs and 2 km/s grid. Kurucz CD-ROM No. 13. Cambridge 13.

Lakshminarayanan, B., Pritzel, A., Blundell, C., 2016. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. arXiv e-prints , arXiv:1612.01474doi:10.48550/arXiv.1612.01474, arXiv:1612.01474.

Lecun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86, 2278–2324. doi:10.1109/5.726791.

Lewandowski, D., Kurowicka, D., Joe, H., 2009. Generating random correlation matrices based on vines and extended onion method. Journal of Multivariate Analysis 100, 1989–2001. URL: https://www.sciencedirect.com/science/article/pii/S0047259X09000876, doi:https://doi.org/10.1016/j.jmva.2009.04.008.

Lin, J., Dotter, A., Ting, Y.S., Asplund, M., 2018. Stellar ages and masses in the solar neighbourhood: Bayesian analysis using spectroscopy and Gaia DR1 parallaxes. Monthly Notices of the Royal Astronomical Society 477, 2966–2975. doi:10.1093/mnras/sty709, arXiv:1803.10875.

Liu, F., Asplund, M., Yong, D., Meléndez, J., Ramírez, I., Karakas, A.I., Carlos, M., Marino, A.F., 2016. The chemical compositions of solar twins in the open cluster M67. Monthly Notices of the Royal Astronomical Society 463, 696–704. URL: https://doi.org/10.1093/mnras/stw2045, doi:10.1093/mnras/stw2045.

Liu, Q., Wang, D., 2016. Stein Variational Gradient Descent: A General Purpose Bayesian Inference Algorithm. arXiv e-prints , arXiv:1608.04471doi:10.48550/arXiv.1608.04471, arXiv:1608.04471.

Lu, Y., Angus, R., Foreman-Mackey, D., Hattori, S., 2024. In This Day and Age: An Empirical Gyrochronology Relation for Partially and Fully Convective Single Field Stars. The Astronomical Journal 167, 159. doi:10.3847/1538-3881/ad28b9, arXiv:2310.14990.

MacKay, D.J., 1992. Bayesian interpolation. Neural computation 4, 415–447.

Mathur, S., 2013. Study of Stellar Magnetic Activity with CoRoT and mbox{Kepler} Data, in: Shibahashi, H., Lynas-Gray, A.E. (Eds.), Progress in Physics of the Sun and Stars: A New Era in Helio- and Asteroseismology, p. 425. doi:10.48550/arXiv.1308.0645, arXiv:1308.0645.

Mathur, S., Claytor, Z.R., Santos, Â.R.G., García, R.A., Amard, L., Bugnet, L., Corsaro, E., Bonanno, A., Breton, S.N., Godoy-Rivera, D., Pinsonneault, M.H., van Saders, J., 2023a. Evolution of rotation and magnetic activity of solar-like stars with age:magneto-(gyro-)chronology, in: PLATO Stellar Science Conference 2023, p. 36. doi:10.5281/zenodo.8140785.

Mathur, S., Claytor, Z.R., Santos, Â.R.G., García, R.A., Amard, L., Bugnet, L., Corsaro, E., Bonanno, A., Breton, S.N., Godoy-Rivera, D., Pinsonneault, M.H., van Saders, J., 2023b. Magnetic Activity Evolution of Solar-like Stars. I. S $_{ph}$-Age Relation Derived from Kepler Observations. The Astronomical Journal 952, 131. doi:10.3847/1538-4357/acd118, arXiv:2306.11657.

McCulloch, W.S., Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics 5, 115–133. URL: https://doi.org/10.1007/BF02478259, doi:10.1007/BF02478259.

Morel, T., Creevey, O.L., Montalbán, J., Miglio, A., Willett, E., 2021. Testing abundance-age relations beyond solar analogues with Kepler LEGACY stars. Astronomy & Astrophysics 646, A78. doi:10.1051/0004-6361/202039212, arXiv:2011.04481.

Moya, A., Amado, P.J., Barrado, D., García Hernández, A., Aberasturi, M., Montesinos, B., Aceituno, F., 2010. Age determination of the HR8799 planetary system using asteroseismology. Monthly Notices of the Royal Astronomical Society 405, L81–L85. doi:10.1111/j.1745-3933.2010.00863.x, arXiv:1003.5796.

Moya, A., Recio-Martínez, J., López-Sastre, R.J., 2021. Ai for dating stars: a benchmarking study for gyrochronology, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1971–1981. doi:10.1109/CVPRW53098.2021.00225.

Moya, A., Sarro, L.M., Delgado-Mena, E., Chaplin, W.J., Adibekyan, V., Blanco-Cuaresma, S., 2022. Stellar dating using chemical clocks and Bayesian inference. Astronomy & Astrophysics 660, A15. doi:10.1051/0004-6361/202141125, arXiv:2201.05228.

Müller, P., Insua, D.R., 1998. Issues in Bayesian Analysis of Neural Network Models. Neural Computation 10, 749–770. URL: https://doi.org/10.1162/089976698300017737, doi:10.1162/089976698300017737.

Neal, R.M., 1993. Probabilistic Inference Using Markov Chain Monte Carlo Methods. Technical Report CRG-TR-93-1. Department of Computer Science, University of Toronto.

Neal, R.M., 1996. Bayesian Learning for Neural Networks. Springer-Verlag, Berlin, Heidelberg.

Neal, R.M., 2012. Bayesian learning for neural networks. volume 118. Springer Science & Business Media.

Nissen, P.E., 2015. High-precision abundances of elements in solar twin stars. Trends with stellar age and elemental condensation temperature. Astronomy & Astrophysics 579, A52. doi:10.1051/0004-6361/201526269, arXiv:1504.07598.

Nissen, P.E., 2016. High-precision abundances of Sc, Mn, Cu, and Ba in solar twins. Trends of element ratios with stellar age. Astronomy & Astrophysics 593, A65. doi:10.1051/0004-6361/201628888, arXiv:1606.08399.

Nissen, P.E., Christensen-Dalsgaard, J., Mosumgaard, J.R., Silva Aguirre, V., Spitoni, E., Verma, K., 2020. High-precision abundances of elements in solar-type stars. Evidence of two distinct sequences in abundance-age relations. Astronomy & Astrophysics 640, A81. doi:10.1051/0004-6361/202038300, arXiv:2006.06013.

Nissen, P.E., Silva Aguirre, V., Christensen-Dalsgaard, J., Collet, R., Grundahl, F., Slumstrup, D., 2017. High-precision abundances of elements

in Kepler LEGACY stars. Verification of trends with stellar age. Astronomy & Astrophysics 608, A112. doi:10.1051/0004-6361/201731845, arXiv:1710.03544.

Noci, L., Roth, K., Bachmann, G., Nowozin, S., Hofmann, T., 2021. Disentangling the Roles of Curation, Data-Augmentation and the Prior in the Cold Posterior Effect. arXiv e-prints , arXiv:2106.06596doi:10.48550/arXiv.2106.06596, arXiv:2106.06596.

Ojha, V.K., Abraham, A., Snášel, V., 2017. Metaheuristic design of feedforward neural networks: A review of two decades of research. Engineering Applications of Artificial Intelligence 60, 97–116. URL: https://www.sciencedirect.com/science/article/pii/S0952197617300234, doi:https://doi.org/10.1016/j.engappai.2017.01.013.

Olivares Romero, J., 2017. Bayesian hierarchical modelling of young stellar clusters. Theses. Université Grenoble Alpes ; Universidad nacional de educación a distancia (Madrid). URL: https://theses.hal.science/tel-01762982.

Papamarkou, T., Hinkle, J., Young, M.T., Womble, D., 2021. Challenges in markov chain monte carlo for bayesian neural networks. arXiv:1910.06539.

Pourzanjani, A.A., Jiang, R.M., Petzold, L.R., 2017. Improving the identifiability of neural networks for bayesian inference. Second workshop on Bayesian Deep Learning (NIPS 2017), Long Beach, CA, USA. URL: https://api.semanticscholar.org/CorpusID:46932278.

Prša, A., Harmanec, P., Torres, G., Mamajek, E., Asplund, M., Capitaine, N., Christensen-Dalsgaard, J., Depagne, É., Haberreiter, M., Hekker, S., Hilton, J., Kopp, G., Kostov, V., Kurtz, D.W., Laskar, J., Mason, B.D., Milone, E.F., Montgomery, M., Richards, M., Schmutz, W., Schou, J., Stewart, S.G., 2016. Nominal Values for Selected Solar and Planetary Quantities: IAU 2015 Resolution B3. The Astronomical Journal 152, 41. doi:10.3847/0004-6256/152/2/41, arXiv:1605.09788.

Räuker, T., Ho, A., Casper, S., Hadfield-Menell, D., 2022. Toward Transparent AI: A Survey on Interpreting the Inner Structures of Deep Neural

47

Networks. arXiv e-prints , arXiv:2207.13243doi:10.48550/arXiv.2207.13243, arXiv:2207.13243.

Rodrigues, T.S., Bossini, D., Miglio, A., Girardi, L., Montalbán, J., Noels, A., Trabucchi, M., Coelho, H.R., Marigo, P., 2017. Determining stellar parameters of asteroseismic targets: going beyond the use of scaling relations. Monthly Notices of the Royal Astronomical Society 467, 1433–1448. doi:10.1093/mnras/stx120, arXiv:1701.04791.

Rosenblatt, F., 1958. The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review 65, 386–408. doi:10.1037/h0042519.

Serenelli, A.M., Bergemann, M., Ruchti, G., Casagrande, L., 2013. Bayesian analysis of ages, masses and distances to cool stars with non-LTE spectroscopic parameters. Monthly Notices of the Royal Astronomical Society 429, 3645–3657. doi:10.1093/mnras/sts648, arXiv:1212.4497.

Silva Aguirre, V., Davies, G.R., Basu, S., Christensen-Dalsgaard, J., Creevey, O., Metcalfe, T.S., Bedding, T.R., Casagrande, L., Handberg, R., Lund, M.N., Nissen, P.E., Chaplin, W.J., Huber, D., Serenelli, A.M., Stello, D., Van Eylen, V., Campante, T.L., Elsworth, Y., Gilliland, R.L., Hekker, S., Karoff, C., Kawaler, S.D., Kjeldsen, H., Lundkvist, M.S., 2015a. Ages and fundamental properties of Kepler exoplanet host stars from asteroseismology. Monthly Notices of the Royal Astronomical Society 452, 2127–2148. doi:10.1093/mnras/stv1388, arXiv:1504.07992.

Silva Aguirre, V., Davies, G.R., Basu, S., Christensen-Dalsgaard, J., Creevey, O., Metcalfe, T.S., Bedding, T.R., Casagrande, L., Handberg, R., Lund, M.N., Nissen, P.E., Chaplin, W.J., Huber, D., Serenelli, A.M., Stello, D., Van Eylen, V., Campante, T.L., Elsworth, Y., Gilliland, R.L., Hekker, S., Karoff, C., Kawaler, S.D., Kjeldsen, H., Lundkvist, M.S., 2015b. Ages and fundamental properties of Kepler exoplanet host stars from asteroseismology. Monthly Notices of the Royal Astronomical Society 452, 2127–2148. doi:10.1093/mnras/stv1388, arXiv:1504.07992.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., Hassabis, D., 2017. Mastering Chess and Shogi by Self-

Play with a General Reinforcement Learning Algorithm. arXiv e-prints , arXiv:1712.01815doi:10.48550/arXiv.1712.01815, arXiv:1712.01815.

Sneden, C., 1973. The University of Texas at Austin. Ph.D. thesis. Ph. D. thesis.

Soderblom, D., 2015a. Astrophysics: Stellar clocks. Nature 517, 557–558. doi:10.1038/517557a.

Soderblom, D.R., 2010. The Ages of Stars. Annual Review of Astronomy and Astrophysics 48, 581–629. doi:10.1146/annurev-astro-081309-130806, arXiv:1003.6074.

Soderblom, D.R., 2015b. Ages of Stars: Methods and Uncertainties, in: Asteroseismology of Stellar Populations in the Milky Way, p. 3. doi:10.1007/978-3-319-10993-0_1, arXiv:1409.2266.

Sousa, S.G., Santos, N.C., Israelian, G., Mayor, M., Monteiro, M.J.P.F.G., 2007. A new code for automatic determination of equivalent widths: Automatic Routine for line Equivalent widths in stellar Spectra (ARES). Astronomy & Astrophysics 469, 783–791. doi:10.1051/0004-6361:20077288, arXiv:astro-ph/0703696.

Sousa, S.G., Santos, N.C., Mayor, M., Udry, S., Casagrande, L., Israelian, G., Pepe, F., Queloz, D., Monteiro, M.J.P.F.G., 2008. Spectroscopic parameters for 451 stars in the HARPS GTO planet search program. Stellar [Fe/H] and the frequency of exo-Neptunes. Astronomy & Astrophysics 487, 373–381. doi:10.1051/0004-6361:200809698, arXiv:0805.4826.

Tishby, N., Levin, E., Solla, S.A., 1989. Consistent inference of probabilities in layered networks: predictions and generalizations. International 1989 Joint Conference on Neural Networks , 403–409 vol.2URL: https://api.semanticscholar.org/CorpusID:15012839.

Valentin Jospin, L., Buntine, W., Boussaid, F., Laga, H., Bennamoun, M., 2020. Hands-on Bayesian Neural Networks – a Tutorial for Deep Learning Users. arXiv e-prints , arXiv:2007.06823doi:10.48550/arXiv.2007.06823, arXiv:2007.06823.

Valls-Gabaud, D., 2014. Bayesian isochrone fitting and stellar ages, in: Lebreton, Y., Valls-Gabaud, D., Charbonnel, C. (Eds.), EAS Publications Series, pp. 225–265. doi:10.1051/eas/1465006, arXiv:1607.03000.

Van-Lane, P., Speagle, J.S., Douglas, S., 2023. A Novel Application of Conditional Normalizing Flows: Stellar Age Inference with Gyrochronology. arXiv e-prints , arXiv:2307.08753doi:10.48550/arXiv.2307.08753, arXiv:2307.08753.

Viscasillas Vázquez, C., Magrini, L., Casali, G., Tautvaišienė, G., Spina, L., Van der Swaelmen, M., Randich, S., Bensby, T., Bragaglia, A., Friel, E., Feltzing, S., Sacco, G.G., Turchi, A., Jiménez-Esteban, F., D'Orazi, V., Delgado-Mena, E., Mikolaitis, Š., Drazdauskas, A., Minkevičiūtė, R., Stonkutė, E., Bagdonas, V., Montes, D., Guiglion, G., Baratella, M., Tabernero, H.M., Gilmore, G., Alfaro, E., Francois, P., Korn, A., Smiljanic, R., Bergemann, M., Franciosini, E., Gonneau, A., Hourihane, A., Worley, C.C., Zaggia, S., 2022. The Gaia-ESO survey: Age-chemical-clock relations spatially resolved in the Galactic disc. Astronomy & Astrophysics 660, A135. doi:10.1051/0004-6361/202142937, arXiv:2202.04863.

von Hippel, T., van Dyk, D.A., Stenning, D.C., Robinson, E., Jeffery, E., Stein, N., Jefferys, W.H., O'Malley, E., 2016. The Power of Principled Bayesian Methods in the Study of Stellar Evolution. arXiv e-prints , arXiv:1605.02810doi:10.48550/arXiv.1605.02810, arXiv:1605.02810.

Werbos, P., 1974. Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Science. Thesis (Ph. D.). Ph.D. thesis. Appl. Math. Harvard University.

Wiese, J.G., Wimmer, L., Papamarkou, T., Bischl, B., Günnemann, S., Rügamer, D., 2023. Towards efficient mcmc sampling in bayesian neural networks by exploiting symmetry. arXiv:2304.02902.

Wilson, A.G., Izmailov, P., 2020. Bayesian Deep Learning and a Probabilistic Perspective of Generalization. arXiv e-prints , arXiv:2002.08791doi:10.48550/arXiv.2002.08791, arXiv:2002.08791.

Xuan, J., Lu, J., Zhang, G., 2019. A survey on bayesian nonparametric learning. ACM Comput. Surv. 52. URL: https://doi.org/10.1145/3291044, doi:10.1145/3291044.

Yadav, R.K.S., Bedin, L.R., Piotto, G., Anderson, J., Cassisi, S., Villanova, S., Platais, I., Pasquini, L., Momany, Y., Sagar, R., 2008. Ground-based CCD astrometry with wide-field imagers. II. A star catalog for M 67: WFI@2.2 m MPG/ESO astrometry, FLAMES@VLT radial velocities. Astronomy & Astrophysics 484, 609–620. doi:10.1051/0004-6361:20079245, arXiv:0803.0004.

Zhao, Y., Saxena, D., Cao, J., 2022. AdaptCL: Adaptive Continual Learning for Tackling Heterogeneity in Sequential Datasets. arXiv e-prints , arXiv:2207.11005doi:10.48550/arXiv.2207.11005, arXiv:2207.11005.

Zhao, Y., Saxena, D., Cao, J., 2022. Revisiting parameter reuse to overcome catastrophic forgetting in neural networks. arXiv e-prints .

51

Appendix A. Appendix Title

Appendix A.1. Single BNN with multiple output

This subsection focuses on the performance and behavior of a single BNN with multiple output dimensions. By leveraging the inherent capacity of BNNs to model uncertainty, we analyze the outcomes derived from our experiments. The results highlight key observations regarding prediction accuracy, uncertainty quantification, and the network's ability to handle multi-output scenarios effectively.
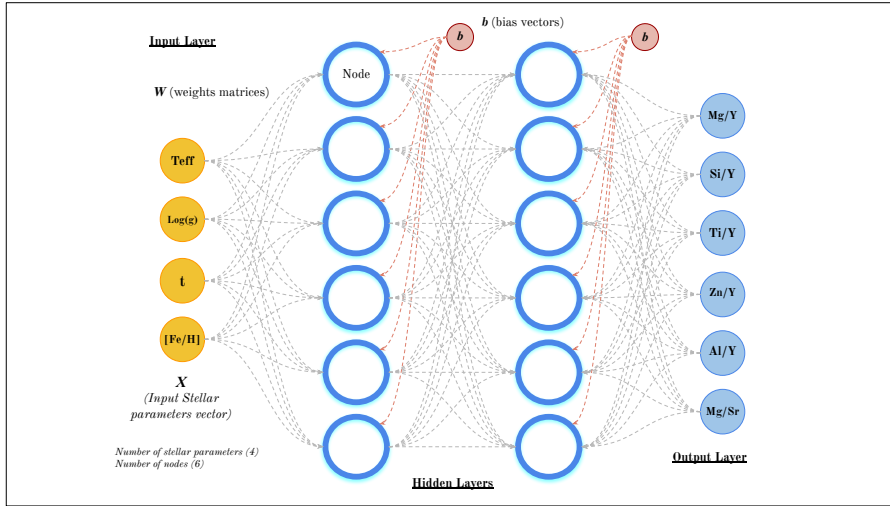

Figure A.7: An example of a multiple-output architecture with two hidden layers, matching the best BNN schematics used for multiple-output configurations during experimentation (single NN for all 6 CCs). W and b represent the weight matrices and bias vectors, *Orange filled circles* are the stellar parameters, (X) as input, and *Blue filled circle* are the CCs output. The *Dashed gray and Red lines* illustrate the information flow within the NN (representing the W matrices and b vectors).

The initial version of our HBM-NNs featured a single neural network that outputs six values, each corresponding to an individual CC, Figure A.7). This picture depicts the best schematic in multiple output configurations for our problem. Parameters W and b represent the weights matrices and bias vectors, respectively. In Figure A.7 the dashed gray and red lines depict the information flow within the NN. Additionally, orange-filled circles are the stellar parameters (X) as input, and the blue-filled circles are the CC

52

outputs. This design choice aimed to streamline the architecture and reduce the number of parameters entering the Bayesian model.

Increasing the number of HLs and nodes in BNNs enhances their capacity to model intricate and complex patterns in data. However, this increased expressiveness comes at the cost of significantly heightened complexity in the parameter space. A higher number of nodes introduces additional symmetries in how the weights can be arranged, as multiple configurations of weights may represent latent features in the data equally well. This symmetry further compounds the challenge, as the posterior distribution may encompass multiple plausible explanations of the data. Consequently, new modes emerge in the parameter space, contributing to multimodality in the posterior distribution.

The proliferation of modes leads to a parameter space characterized by numerous local optima, complicating the inference process. Such landscapes make posterior exploration more difficult for traditional optimization and sampling methods, as they can become trapped in suboptimal regions. This increased complexity and multimodality result in greater uncertainty in predictions, making them more variable and less stable.

The inherently multimodal and high-dimensional nature of the parameter space in BNNs (e.g., Müller and Insua, 1998; Bishop, 2006; Izmailov et al., 2021; Arbel et al., 2023) poses significant challenges for sampling from their posterior distributions. This complexity arises from the interplay of the non-linearities in neural network architectures and the prior assumptions, often resulting in posterior distributions with multiple isolated modes. While MCMC methods are commonly employed for posterior inference, these algorithms can struggle to adequately explore the parameter space. Specifically, MCMC samplers may become trapped in a single mode, failing to traverse energy barriers to explore other regions of the posterior. This limitation can lead to incomplete mixing, slow convergence, and posterior estimates that are biased or unrepresentative of the true distribution. Such inadequacies can compromise the reliability of downstream analyses, producing heuristics or predictions that fail to capture the full uncertainty encoded in the Bayesian framework. Addressing these challenges requires either developing advanced sampling methods tailored to the unique characteristics of BNN posteriors or leveraging variational inference techniques as an alternative.

This issue represented a crucial point during our experimentation. Figures A.8 and 6 show an example of the posterior distributions of weights matrices in BNNs used in this work, each illustrating different parameter

53

spaces influenced by the complexity of the network. Both figures depict the posteriors of each element in the first weight matrix, which represents the connections between the input vector and the first HL nodes. Considering the reasons outlined during this work, we aim to have parameter spaces as monomodal as possible in order to have the information concentrated, improving computational times and uncertainty. Even though our first iteration presented complex parameter spaces with multimodality, due to the deeper architecture (Figure A.7), we first opted for these approaches to avoid a higher number of parameters in our Bayesian model.

Initially, to address this multimodality challenge, we implemented various strategies, principally involving the imposition of narrower and centered priors, with a focus on the primary mode of the distributions for the weights and biases. To achieve this, we trained an identical NN architecture as the one in the Bayesian model multiple times, aiming to attain a configuration of W and b with lower loss. Subsequently, these configurations were introduced as homonyms priors in PyStan.

We determined that the optimal balance between prediction accuracy and computational efficiency was achieved with a setup comprising two hidden layers, each consisting of 6 nodes (Figure A.7) for the MCMC-NUTS sampler and 10 for the VI-ADVI-based one. Leveraging the expressive capacity of NNs, we successfully captured intricate relationships among stellar parameters, yielding results comparable to Moya et al. (2022) (for the same 23 test set stars they achieve a MAE=0.86) while benefitting from automated learning. In Figure A.9 we present a comparison of the results obtained by these two configurations. Both architectures achieve similar results, a MAE=0.88 for the NUTS case and 0.87 for the ADVI one.

While this approach yielded reliable results and heuristics, the constrained nature of the priors has the potential to hinder the sampling of other modes, particularly posing risks when encountering distant or disconnected modes in this or other datasets. Moreover, this restrictive approach has a potential adverse impact on computational times.

To achieve more reproducible and stable results we decided to start new experimentation, this time focusing on monomodal distributions that let us concentrate the information. This was the main reason we built a new architecture based on 6 simpler NNs, one for each CC (Figure 2). Illustrating this, Figure 6 shows a parameter space with very little multimodality. The posterior distribution in this case is much less complex, suggesting a more deterministic relationship between inputs and output. The near-unimodal
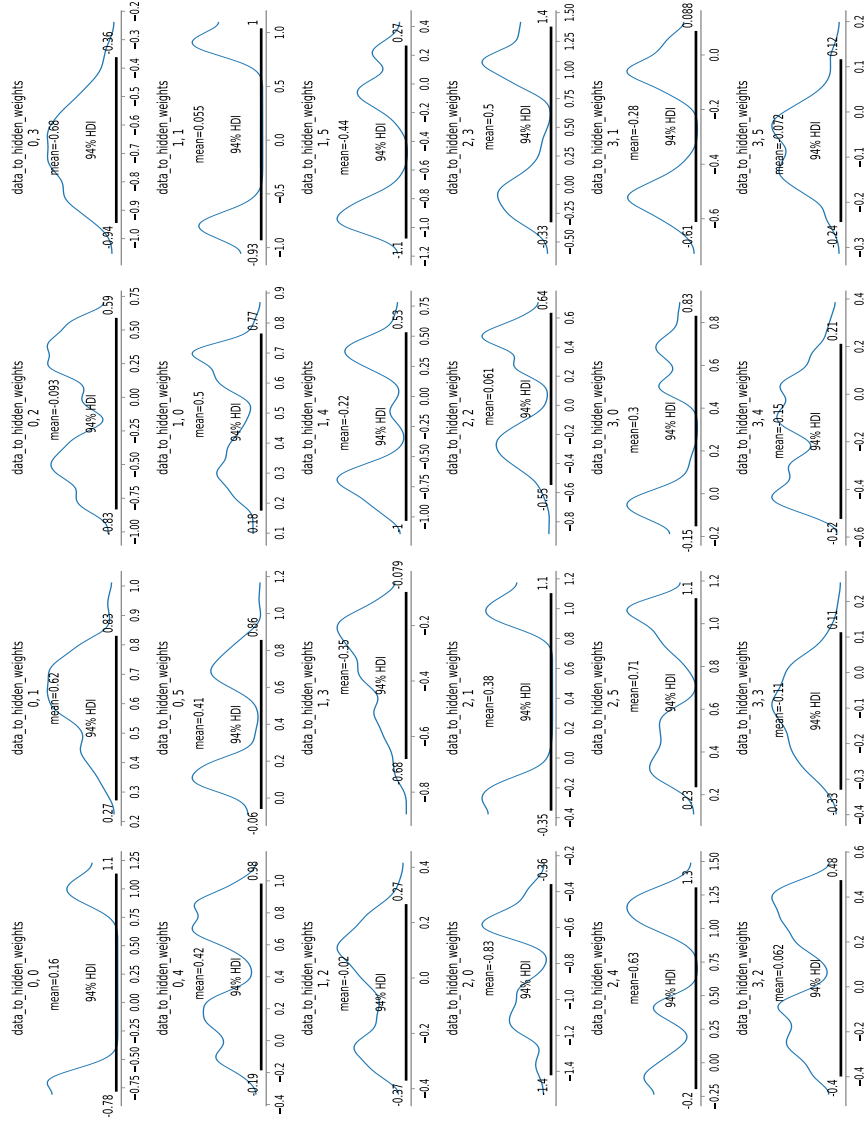
54

Figure A.8: Example of posterior distributions for a weight matrix in a two hidden layer, six nodes per hidden layer neural network applied to a model with six outputs, 6 CCs prediction at once (for better illustration, see figure A.7). The figure shows the distributions of each weight (W_{ji}) in the initial weight matrix, which represents the connections between the input data and the first hidden layer. Each distribution includes its mean and the 94% Highest Density Interval (HDI). The distributions are organized from left to right and top to bottom, corresponding to each of the four stellar parameters and their connections to the six nodes in the first hidden layer.
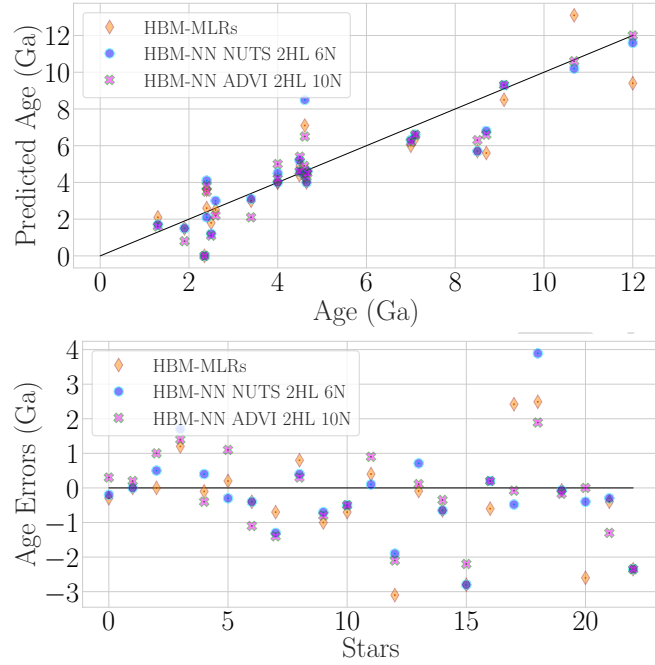
55

Figure A.9: Hierarchical Bayesian model results comparison. *(HBM-MLRs:)* architecture whose probability relationships are modeled by multi-linear regressions, from Moya et al. (2022). (*HBM-NN NUTS 2HL 6N:*) architecture whose probability relationships are modeled by one NN. The architecture consists of 2 Hidden Layers with 6 nodes per layer and multiple outputs (6 CCs), sampled with the MCMC-NUTS algorithm. (*HBM-NN ADVI 2HL 10N:*) architecture whose probability relationships are modeled by one NN. The architecture consists of 2 hidden layers with 10 nodes per layer and multiple outputs (6 CCs), sampled with the VI-ADVI algorithm.(*Top*:) Age prediction for the 23 test stars. (*Bottom*:) Age errors for the 23 test stars.

landscape indicates a concentrated posterior distribution, meaning that there are fewer alternative hypotheses about the best weight configurations. This simplicity reduces the uncertainty in parameter estimation, leading to a potentially more reproducible solution.

ADA (Adapt Delta Parameters): A parameter in the MCMC-NUTS algorithm that adjusts the step size dynamically to improve sampling efficiency.

ADVI (Automatic Differentiation Variational Inference): A computational method for approximating posterior distributions in Bayesian inference using automatic differentiation to optimize variational objectives efficiently.

AI (Artificial Intelligence): A broad field of computer science focused on creating systems capable of performing tasks that typically require human intelligence, such as learning, reasoning, and decision-making.

BNN (Bayesian Neural Network): A type of neural network that incorporates Bayesian principles to estimate uncertainty in predictions by learning probability distributions over model parameters.

CC (Chemical Clock): A method used in astrophysics to estimate stellar ages by analyzing the chemical abundances in a star's spectrum.

DL (Deep Learning): A subset of machine learning that uses neural networks with multiple hidden layers to model and solve complex patterns and relationships in data.

GPU (Graphical Processing Unit): A specialized hardware device originally designed for rendering graphics but now widely used in parallel computing tasks, including training neural networks and deep learning models.

HBM (Hierarchical Bayesian Model): A statistical model that uses Bayesian inference to represent relationships across multiple levels of parameters, often capturing dependencies in complex systems.

HDI (Highest Density Interval): A Bayesian measure used to describe the range within which a parameter is most likely to lie, given the posterior distribution.

HL (Hidden Layer): A layer of neurons in a neural network that processes inputs received from preceding layers, extracting and transforming features to be passed to subsequent layers.

ISM (Interstellar medium): is the matter and radiation that exists in the space between the star systems in a galaxy.

MAE (Mean Absolute Error): A metric used to measure the average magnitude of errors in predictions without considering their direction, often used to evaluate regression models.

ML (Machine Learning): A subset of artificial intelligence focused on developing algorithms that enable systems to learn from data and improve performance without being explicitly programmed.

MLR (Multiple Linear Regression): A statistical technique used to model the relationship between a dependent variable and multiple independent variables through a linear equation.

MS (Main Sequence Star): A star that is in the longest stable phase of its lifecycle, where it generates energy through the fusion of hydrogen into helium in its core.

MTD (Maximum Tree Depth Parameters): A parameter in the MCMC-NUTS algorithm that controls the maximum depth of the tree in Hamiltonian Monte Carlo sampling to avoid excessive computational cost.

MCMC (Markov Chain Monte Carlo): A class of algorithms used for sampling from complex probability distributions, often employed in Bayesian statistics.

NEFF (Number of Effective Samples): A metric in MCMC that quantifies the effective number of independent samples from the posterior distribution, accounting for autocorrelation.

NN (Neural Network): A computational model inspired by the structure and function of biological neural networks, consisting of interconnected layers of nodes (neurons) used to approximate complex functions.

NUTS (No-U-Turn Sampler): A variant of the Hamiltonian Monte Carlo algorithm that dynamically adjusts its trajectory to improve sampling efficiency and avoid retracing steps.

VI (Variational Inference): A method in Bayesian inference that approximates posterior distributions through optimization, offering a computationally efficient alternative to MCMC.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: