E-ISSN: 2583-9667
Indexed Journal
Peer Reviewed Journal
https://multiresearchjournal.theviews.in

Received: 23-04-2023
Accepted: 29-06-2023

INTERNATIONAL JOURNAL OF ADVANCE RESEARCH IN MULTIDISCIPLINARY

Volume 1; Issue 1; 2023; Page No. 942-947

A study on asteroseismology of sun-like stars using artificial neural networks

[1]Chandan Kumar and [2]Dr. Shailesh Kumar Singh

[1]Research Scholar, Department of Physics, Maharaja Agrasen Himalayan Garhwal University, Uttarakhand, India
[2]Professor, Department of Physics, Maharaja Agrasen Himalayan Garhwal University, Uttarakhand, India

Corresponding Author: Chandan Kumar

Abstract

Asteroseismology, the precise determination of star parameters, is essential to many facets of astrophysics. In fact, precise stellar attributes enable us to constrain stellar evolution models more tightly. Moreover, the precision of exoplanet properties is largely dependent on the precision of their host star properties. Finally, defining star populations in the Milky Way and reconstructing its history depend heavily on acquiring precise stellar attributes. It is now widely recognized that the basic star parameters (mass, radius, age, etc.) may be determined with extremely high precision using the low-degree oscillation frequencies in conjunction with the classical observables. An asteroseismic research on 22 of the brightest sun-like stars that Kepler had detected. They were able to deduce the ages to within 2.5% and the masses and radii to an accuracy level of 1.0%. In a comprehensive analysis of the effects of stellar physics uncertainty, conducted a case study on the planet-host star HD52265, which was detected by the CoRoT satellite. They have determined the mass, radius, and age of HD52265 with uncertainties of 7%, 1.5%, and 10%, respectively, after optimizing stellar parameters for a variety of oscillation frequency combinations, large and small frequency separations, frequency ratios, and spectroscopic observables, as well as for various input physics.

Keywords: Asteroseismology, Sun-Like Stars, Artificial Neural Networks

Introduction

It is a fundamentally important problem to trace the stories of brilliant lives. Our knowledge of star evolution can be precisely tested because to the extensive collections of stellar data. In addition to being significant for stellar development, the stellar attributes are also significant for exoplanetary systems, stellar populations, and Galactic Archeology. To properly characterize the linked planets, it is crucial to ascertain the basic properties of the host stars. In order to properly characterize the related planets, used asteroseismology to assess the fundamental features of 66 Kepler planet-candidate host stars. Using inferences of the parameters of the respective host stars for six stellar systems, able to better restrict the attributes of exoplanets. A consistent investigation of 33 Kepler planet-candidate host stars was presented, who also determined the stellar characteristics with great precision. In an ensemble investigation of 500 sun-like stars spotted that the observed mass distribution deviates from what the models of artificial stellar populations in the Milky Way predicted.

There is a complex relationship between the observables (effective temperature, surface metallicity, luminosity, oscillation frequencies, etc.) and the stellar parameters (mass, starting Helium abundance, initial metallicity, mixing-length, and age). Even though the physics of the problem shows that all of these quantities are closely related, the inference problem is complicated by the fact that there is no known analytical direct inverse relation between them and that the constellation of stellar parameters is probably highly nonlinearly related to the observables. Due to the former, efforts are undertaken to find the most pertinent set of models by sampling the model space. Furthermore, the problem exists in a high-dimensional space since the inference process involves consuming tens of inputs and producing many outputs. Finding the underlying nonlinear relationships is a challenging task, particularly when using traditional techniques like regression.

Numerous methods exist for identifying stellar attributes, related errorbars, and best-fitting models in the context of both helio- and asteroseismology. Of the several methods currently in use, most of them entail calculating multiple evolutionary tracks for individual stars and optimizing the stellar parameters for a specific set of observables. The resulting uncertainties on the stellar parameters are not only computationally costly and time-consuming, but they are also hard to quantify, especially for approaches that use

local optimization algorithms. Simpler approaches that rely on searching fixed grids are computationally efficient, but their accuracy is constrained by the grid's resolution and the systematics involved in concentrating on small search regions. In this research outline and evaluate an artificial neural network, a machine-learning technique that may be utilized to quickly and precisely ascertain the basic star characteristics using traditional seismic and spectroscopic measurements.

Research Methodology

Training the network using stellar models

Multiple layers of neurons make up a neural network. A group of parameters known as "hyper-parameters" includes the decisions on the number of layers and the number of neurons per layer. Heuristics are employed to determine the hyper-parameters, which are dependent on the problem's complexity. By optimizing the hyper-parameters on a fixed grid, the network shown and created for the current situation. Before the neural network can be used to solve a physical problem, it must first learn. We apply a learning technique that trains the network using the stellar models to predict the basic stellar properties for a given set of measurements.

Learning algorithm

The network's neurons are biased by nature, and the weights of the connections between neurons in successive levels are related. During training, the network picks up these weights and biases. We represent the weight between the kth neuron in the (l − 1)th layer and the jth neuron in the lth layer by Wl, and the bias of the jth neuron in the lth layer by bl. The activation function, denoted by σ, is contingent upon the type of neuron, such as a rectifying neuron, hyperbolic tangent neuron, or sigmoid neuron. Since rectifying neurons have recently been demonstrated to be a better model of biological neurons and to function on par with or better than sigmoid and hyperbolic tangent neurons, we have chosen them for this research. For a given input vector x (= a1), the weights and biases of the network can be iteratively calculated in terms of its output y (= aL, where L is the total number of layers).

A dataset comprising the input vectors xj and the matching output vectors yj is required for training; ideally, this dataset should cover the whole space of the problem at hand. An example consists of an input vector and its matching output vector. where the regularization parameter is denoted by λ, the length of the output vector by m, and the number of training examples by n. During the training process, the cost function is minimized in relation to the network's weights and biases. The traditional gradient descent techniques to update the weights and biases are highly costly because practical problems usually require a large training dataset. Therefore, stochastic gradient descent is employed, which uses the gradient derived from a randomly chosen subset, referred to as a mini-batch, of the entire dataset to estimate the true gradient of the cost function. The back-propagation algorithm, which is explained below, can be used to determine the gradient of the cost function with respect to the weights and biases.

1. Set the input vector as activation a^1.
2. For each $l = 2, 3, \ldots, L$, compute the vectors $z^l = W^l a^{l-1} + b^l$ and activations $a^l = \sigma(z^l)$.
3. Compute the vector $\delta^L = \nabla_a C \odot \sigma'(z^L)$, where $\nabla_a C$ is the vector $\partial C / \partial a_j^L$, $\odot$ denotes Hadamard product, and $\sigma'(z^L) = \partial a_j^L / \partial z_j^L$.
4. For each $l = L - 1, L - 2, \ldots, 2$, compute vectors $\delta^l = ((W^{l+1})^T \delta^{l+1}) \odot \sigma'(z^l)$.
5. Compute $\partial C / \partial w_{jk}^l = a_k^{l-1} \delta_j^l$ and $\partial C / \partial b_j^l = \delta_j^l$.

Training models

The training models are built using the stellar evolution code MESA. With the exception of excluding the overshoot and using the solar metallicity mixtures, the models are produced using the input physics outlined. With starting parameters evenly and randomly distributed throughout the following ranges, we construct 20,000 evolutionary tracks: M ∈ 0.70 − 1.10M, Yi ∈ 0.20 − 0.40, Zi ∈ 0.003 − 0.040, and αMLT ∈ 1.20 − 2.50. It is anticipated that the evenly and randomly distributed grid points will better sample the parameter space. To cover a range of eras, the star models are periodically retrieved and evolved. When the age surpasses 20 Gyr or the center hydrogen mass fraction drops below 10−4, the evolution is terminated. The 360,000 models we produced, which cover a variety of masses, beginning compositions, mixing lengths, and ages. Because these stars have convective cores and a sixth parameter, the overshoot, must be considered, we did not take into account the mass range of 1.10 −1.60M in this analysis. Furthermore, a careful treatment is necessary since the existing models of atomic diffusion are more subtle in this mass range.

Results and Discussion

We usually have luminosity from parallaxes, low-degree oscillation frequencies (l = 0, 1, and 2) from high precision photometry, and values of the effective temperature and surface metallicity from spectroscopic studies. To ascertain the stellar attributes, we employ an appropriate mix of the aforementioned observables. We use ADIPLS to calculate the low-degree adiabatic oscillation frequencies for each model. The surface term has an impact on the theoretical model frequencies. If we train the network in the current case using the raw model frequencies, the surface term will add systematic mistakes, resulting in inaccurate estimation of actual stars' stellar properties. As a result, we train the network using frequency ratios that are insensitive to the surface effect rather than the individual oscillation frequencies.

The 360,000 models in total are separated into three categories: 240,000 training examples, 60,000 validation examples, and 60,000 test examples. The network parameters, such as weights and biases, are defined and adjusted using training examples; the network hyper-parameters, such as the number of hidden layers, neurons per layer, mini-batch size, learning rate, regularization parameter, etc., are adjusted using validation examples; and the accuracy of the predictive algorithm is assessed using test examples. By deducting the respective mean values from the training data and dividing the residual by the standard deviations, we normalize the network's inputs and outputs. When needed, the normalized values, mean values,

and standard deviations can be used to reconstruct the original inputs and outputs. In theory, normalization is not required. sary for this issue, but in reality, it speeds up training and lowers the likelihood of becoming trapped in local optima.

Training

THEANO, a publicly accessible Python-based machine-learning program, was used for the training and was run on the SEISMO cluster at TIFR. To update the weights and biases using the stochastic gradient decent approach, we compute the gradient of the cost function with a mini-batch size of 50. Finding and using the ideal mini-batch size is necessary since a too-small one causes a significant inaccuracy in the gradient computation, while a big one slows down learning. The approach is continued for multiple epochs until convergence is reached, updating the weights and biases for each mini-batch of the training data (240,000/50 = 4,800 mini-batches ≡ a "epoch"). The training's soundness is verified using the validation data. For example, the mean square errors (MSEs) for three distinct selections of the regularization parameter, λ, between the inferred and true parameters for the training and validation data as a function of the epoch. Even if the MSE for the training data is still declining, the MSE for the validation data first declines until epoch ∼ 1200 and then begins to increase for an underestimated choice, λ = 1.00 × 10−6. When the network overlearns the training data and is unable to generalize the prediction on the unseen validation, this is a sign of over-fitting information. The MSEs for the training and validation data fall with iteration and saturate at the same value for the ideal choice, λ = 1.69 × 10−6. The MSEs are lower with an inflated choice, λ = 4.00 × 10−6, but they saturate at higher values.

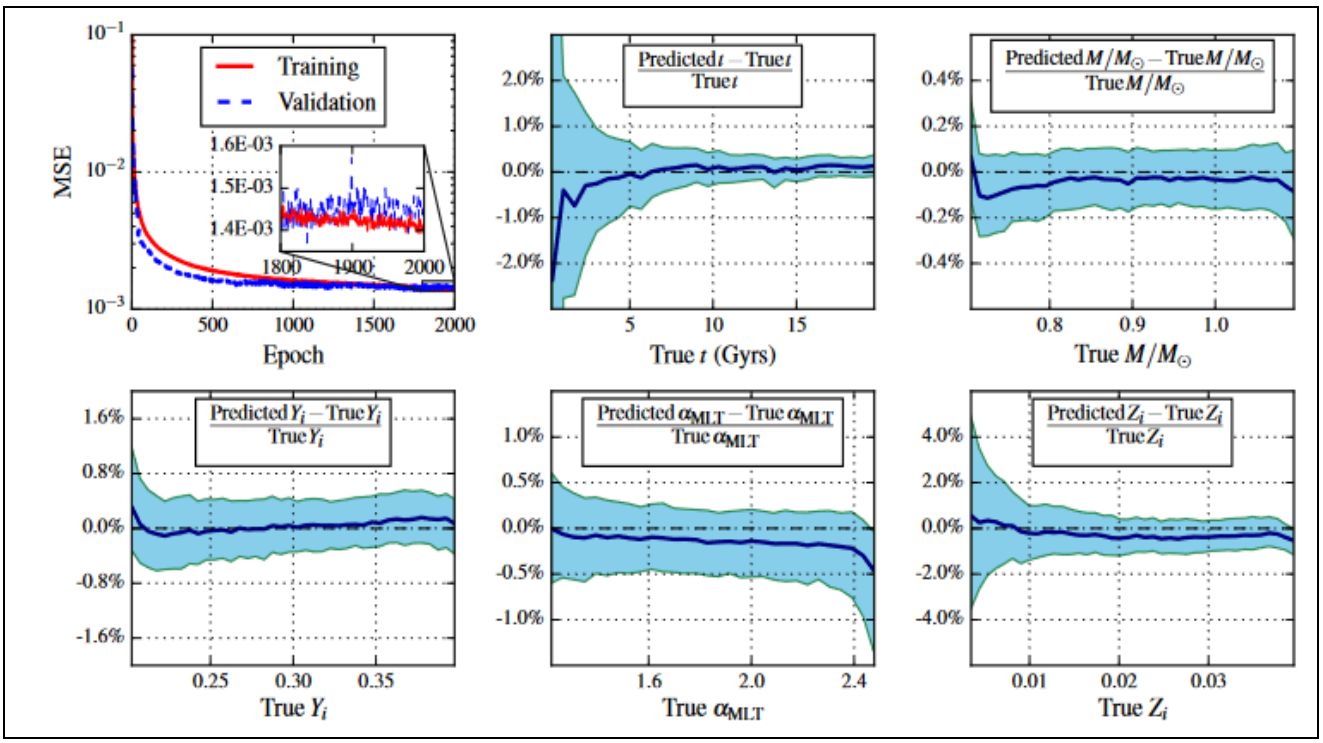

Fig 1: Well-trained network should not over-fit the data, a criterion we meet, as demonstrated by the upper-left panel which shows that the training and validation data are fit to the same degree at the end of the training. The inset shows the mean square errors for the last 200 epochs. The remaining panels show errors incurred in predicting the fundamental parameters post training. The median and 1-σ errors are plotted to illustrate the accuracy of the prediction.

As seen in Figures upper-left panel, the network's weights and biases converge and do not overfit the data. The network's success in determining the different basic parameters for the test data is displayed in the five more panels of Figure. Predicting basic stellar properties for the test data involves methodological flaws, which rely, for example, on the training models' grid resolution. Internal error is the term used to describe this problem. The graphic shows that the internal errors for age, mass, and radius are significantly less than those required by PLATO. We train another network with four more inputs, two r02 at radial orders 14 and 15 and two r01 at radial orders 15 and 16, to the current set in order to verify the impact of the selected number and range of the radial orders. At the lower end, this necessitates two more modes for every degree. Since the MSE just marginally drops from 1.39 × 10−3 to 1.32 × 10−3, we conclude that this network's prediction performance is not substantially different from the prior one. In contrast to Figure-1, a substantially higher predictive power would suggest a smaller scatter in Figure-2 (which is not the case here). The lack of substantially fresh independent information in these extra radial orders could be the cause of this. The network without extra radial commands is represented by the following results.

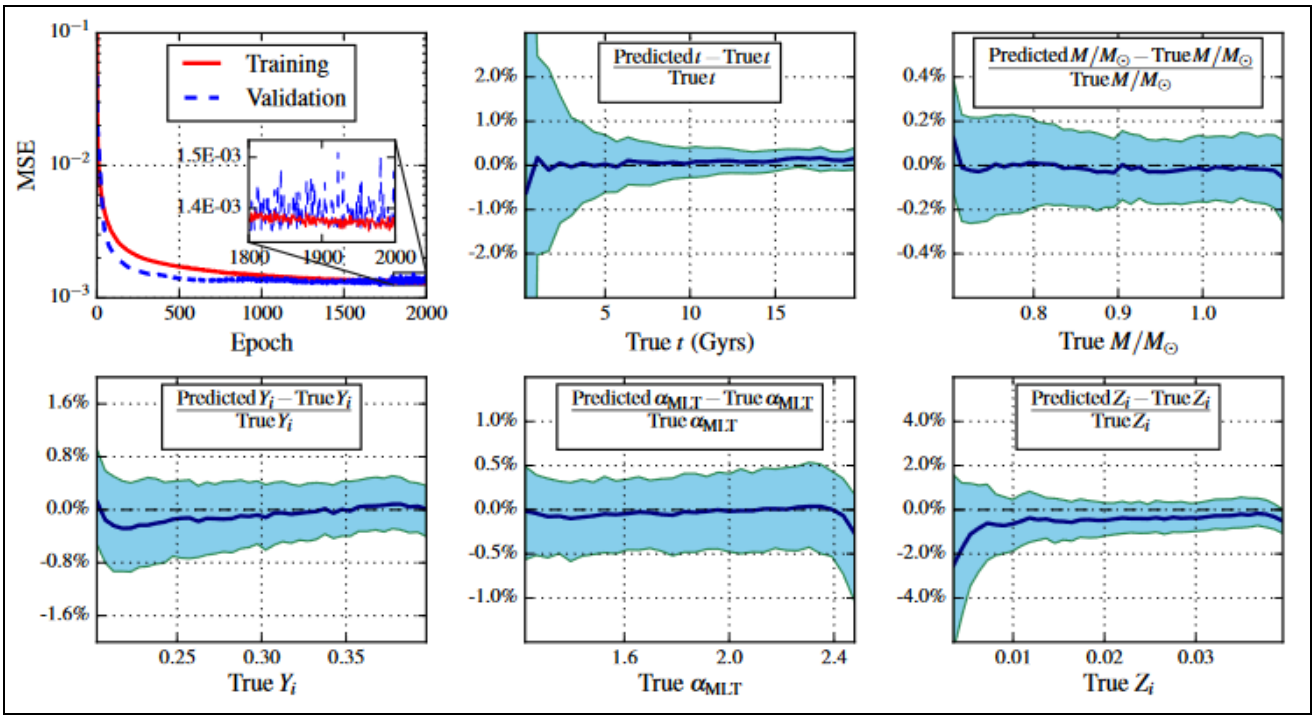

Fig 2: Same as except that the corresponding network has 4 additional inputs, two r_{02} at radial orders 14 and 15 and two r_{01} at radial orders 15 and 16, to the current set.

Test of grid resolution

We randomly select 120,000 of the 360,000 models for testing and validation (60,000 for each) in order to determine whether the aforementioned grid is dense enough for training. Four networks with varying amounts of examples-the first with 20,000, the second with 40,000, the third with 100,000, and the fourth with 240,000-are trained using the remaining 240,000 models. The goal is to compare each of these networks' predicting abilities to the same test dataset, which consists of the 60,000 models we placed aside. With the exception of the final scenario, in which every model is included, randomization guarantees that approximately the same number of models are left out of each evolutionary track.

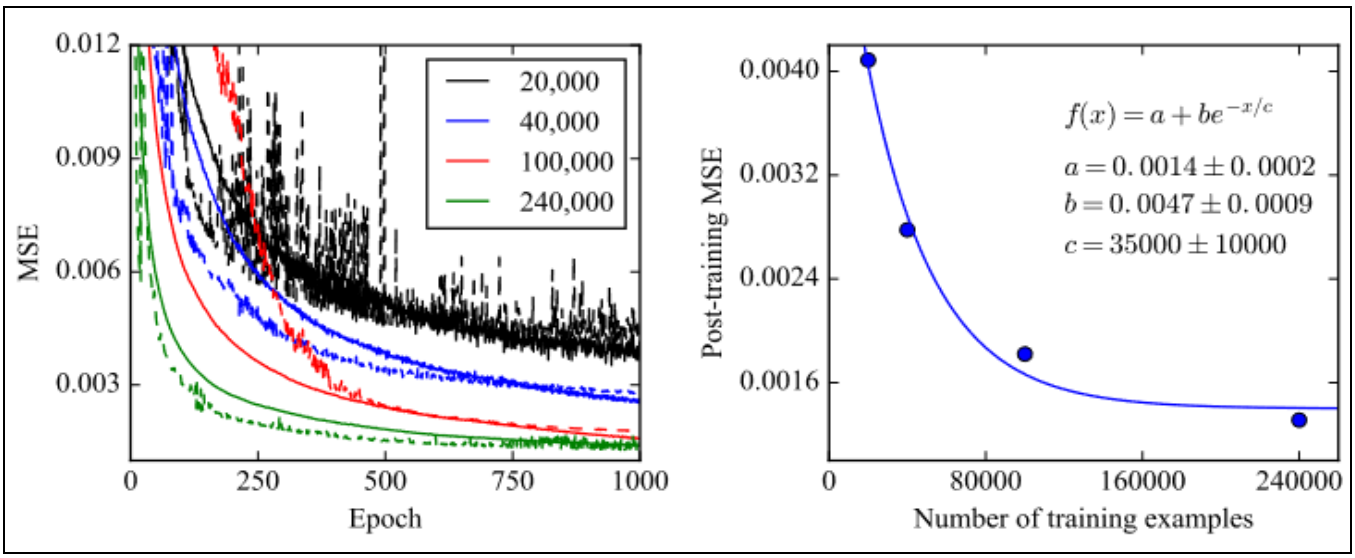

Fig 3: Left panel shows the mean square errors for training (continuous lines) and validation (dashed lines) data as a function of the epoch for all the four training sets. Right panel shows the MSE for the test data at the end of the training as a function of the number of training examples

In Figure-3, the MSEs for training and validation data are displayed in the left panel as a function of epoch for each of the four sets, while the MSE for test data at the conclusion of training is displayed in the right panel as a function of the number of training examples. The more training instances and saturates there are, the lower the MSE is for the test, training, and validation data. As seen in the right panel, we use an exponential function to approximate this fall-off in order to extrapolate it to infinity. This provides an estimate of 0.0014 ± 0.0002 for the MSE at infinity (asymptotic value). When 240,000 examples are used for training, the MSE for test data is 0.0013, which is in line with the

asymptotic value. This indicates that before 240,000 models, the network performance saturates as a function of training examples, and thus using denser grids for training is unlikely to result in a noticeable improvement. Because the neural network only approximates the physics of this problem, it should be noted that the asymptotic MSE does not go to zero. But the theoretical ambiguity brought forth by the network's use (internal mistake) is significantly less than the measurement errors.

Test on various stars

We can now identify the five basic stellar parameters (M, Yi, Zi, αMLT, and t) using the trained network. It is possible to determine the other parameters, like radius, separately. In fact, if we train the network to produce only one parameter at a time, we might be able to obtain the evolutionary parameters independently of one another. Networks that output all five parameters are referred to as NN5, whereas networks with only one output parameter are called NN1.

In the preceding step, we trained the network using stellar models. The physics of actual stars may not match the set of inputs assumed by the stellar models. Furthermore, in the case of a real star, the network's inputs have observational uncertainty. We use the trained networks (NN5 and NN1) for three well-studied real stars and one artificial star to determine the fundamental stellar parameters and compare them with the known values in order to test the robustness of the determination against the uncertainties in the standard stellar physics and the observational uncertainties.

The Sun

We calculate the necessary inputs for the network using spectroscopic measurements and seismic data. To determine the distribution of the output parameters, we use the input's covariance matrix to generate 40,000 realizations of the input, assuming that the input's errors follow a multivariate Gaussian distribution. We calculate the parameter values' median as well as the 16th and 84th percentiles for positive and negative un- certainties about them. The uncertainty is influenced by the internal error, which was measured in the preceding section. We observe that the uncertainties in the measurements usually make it hard to separate the internal error from the error caused by the standard procedures.

Table 1: Predictions for various stars using different methods. NN5 refers to the simultaneous infer- ence of the 5 parameters, whereas NN1 infers each parameter independently. The Metc15 and Rees16 values, respectively.

Star	Method	$M/M_\odot$	$R/R_\odot$	Y_i	Z_i	t(Gyr)	$\alpha_{\rm MLT}$
Sun	NN5	$0.995^{+0.058}_{-0.055}$	...	$0.291^{+0.032}_{-0.035}$	$0.0216^{+0.0043}_{-0.0036}$	$4.46^{+0.20}_{-0.22}$	$1.77^{+0.11}_{-0.10}$
Sun	NN1	$0.992^{+0.066}_{-0.059}$	$1.001^{+0.022}_{-0.022}$	$0.292^{+0.037}_{-0.036}$	$0.0223^{+0.0035}_{-0.0031}$	$4.44^{+0.18}_{-0.17}$	$1.77^{+0.12}_{-0.11}$
Sun	...	1.000	1.000	0.275	0.0200	4.60	...
Sun	Grid	0.981 ± 0.030	0.996 ± 0.015	...	...	4.46 ± 0.20	...
16 Cyg A	NN5	$1.085^{+0.059}_{-0.065}$	...	$0.271^{+0.051}_{-0.045}$	$0.0257^{+0.0050}_{-0.0044}$	$6.65^{+0.58}_{-0.67}$	$1.86^{+0.12}_{-0.10}$
16 Cyg A	NN1	$1.083^{+0.059}_{-0.066}$	$1.229^{+0.025}_{-0.026}$	$0.276^{+0.037}_{-0.040}$	$0.0283^{+0.0042}_{-0.0040}$	$7.17^{+0.75}_{-0.79}$	$1.87^{+0.11}_{-0.10}$
16 Cyg A	Metc15	1.08 ± 0.02	1.229 ± 0.008	0.25 ± 0.01	0.0210 ± 0.0020	7.07 ± 0.26	...
16 Cyg A	Grid	1.02 ± 0.04	1.182 ± 0.020	...	...	5.29 ± 1.00	...
16 Cyg B	NN5	$1.085^{+0.057}_{-0.058}$	...	$0.248^{+0.041}_{-0.042}$	$0.0252^{+0.0044}_{-0.0037}$	$6.69^{+0.31}_{-0.32}$	$1.89^{+0.12}_{-0.10}$
16 Cyg B	NN1	$1.095^{+0.069}_{-0.069}$	$1.140^{+0.023}_{-0.024}$	$0.239^{+0.043}_{-0.041}$	$0.0272^{+0.0032}_{-0.0031}$	$7.14^{+0.62}_{-0.63}$	$1.91^{+0.12}_{-0.11}$
16 Cyg B	Metc15	1.04 ± 0.02	1.116 ± 0.006	0.26 ± 0.01	0.0220 ± 0.0030	6.74 ± 0.24	...
16 Cyg B	Grid	1.08 ± 0.03	1.124 ± 0.015	...	...	6.10 ± 0.50	...
Elvis	NN5	$0.967^{+0.086}_{-0.073}$	...	$0.291^{+0.055}_{-0.061}$	$0.0189^{+0.0041}_{-0.0031}$	$6.23^{+0.64}_{-0.64}$	$1.84^{+0.13}_{-0.11}$
Elvis	NN1	$0.986^{+0.094}_{-0.084}$	$1.069^{+0.033}_{-0.032}$	$0.259^{+0.068}_{-0.071}$	$0.0238^{+0.0043}_{-0.0040}$	$7.14^{+0.97}_{-1.01}$	$1.89^{+0.16}_{-0.13}$
Elvis	Rees16	1.000	1.087	0.267	0.0176	6.84	1.67
Elvis	Grid	1.037 ± 0.03	1.103 ± 0.015	...	...	6.07 ± 0.40	...

The parameters for the Sun derived from NN5 and NN1 are listed in Table-1. Our accuracy in retrieving the solar parameters is 5.5%, 2.2%, and 4.5% of the Sun's mass, radius, and age, respectively. Considering the accuracy of the solar data, the precision is subpar even if it is well within PLATO's requirements. The low number of seismic inputs employed (we used a modest number of seismic inputs anticipating the fact that a tiny set of modes is observed in faraway stars) may be partially responsible for this. More significantly, because the measurements have associated uncertainties, the quality of the inference is deteriorated by the correlations among the stellar parameters, such as the anti-correlation between M and Yi. The correlation matrix of the stellar parameters derived from the output parameter distribution is displayed. The distribution as a function of two stellar parameters is displayed in the remaining panels of the figure. M and αMLT exhibit a positive association, however M and Yi are found to be anti-correlated. Although determining the other parameters would be more precise if certain stellar parameters, like Yi, were fixed using the Helium-to-Metal Enrichment Law and αMLT using solar calibration, they also become more susceptible to significant systematic errors show that the systematics associated with the fixed Yi is comparable to the statistical uncertainties). Yi is constrained by the examination of the acoustic glitches), which can be utilized to increase accuracy.

Comparison of neural network with a conventional method

High-dimensional nonlinear functions can be approximated

using neural networks, but other mathematical functions, such as multivariate polynomials, can also be used. A multivariate polynomial has (d + N)!/(d!N!) free parameters, where d is the polynomial's dimensionality and N is its degree. For big d, this scales as dN /N! (see, for example, Bishop 1995). A larger dataset-in this case, a dense grid of models-is necessary for the more free parameters to be determined with reliability. Such approximation functions are limited in their utility by the need for a large dataset. During training, the neural network learns from the data itself to approximate the nonlinear function of several variables in terms of superpositions of nonlinear functions of a single variable. Therefore, rather than just the dimensions, the complexity of the neural network is dependent on the complexity of the underlying physical process.

We estimate the masses, radii, and ages of all four stars using the method outlined, on the grid built for the training in order to compare the results with a grid-based technique. Each star's values are shown in the fourth row of Table-1. The grid may be excessively sparse for this traditional approach, based on the comparatively erroneous parameter values. However, because to its superior scaling with dimensionality, the neural network is able to extract sufficient information from the same grid to produce accurate predictions. This demonstrates the neural networks' capacity for learning empirically.

Conclusion

We have developed an effective empirical equivalent because there isn't a formal inverse link between the evolutionary parameters and star observations. Currently, the neural network can be viewed as a succinct map of the space where all of the inputs and outputs are located, or as a high-dimensional look-up table. While it takes orders of magnitude less time, the trained network predicts stellar properties with precision comparable to the sophisticated grid modeling methods.

We observe that the network is trained optimally for the parameter regime under study and does not appear to get better when trained on denser grids. When 360,000 models are used for training, the mean square error almost reaches its asymptotic value, illustrating this. Retraining the network is computationally inexpensive (about 50 CPU hours), which is helpful when altering its inputs or outputs. We discover that increasing the number of radial orders of frequency measurements utilized as inputs has no discernible effect on the network's prediction power. The network's accurate inferences of the parameters of three real stars (the Sun, 16 Cyg A & B) and an artificial star (Elvis), whose structure is calculated using a different prescription, provide early evidence that the network is resilient to changes in the input physics.

References

1. Rauer H, *et al*. The PLATO 2.0 mission. Experimental Astronomy. 2014;38(1-2):249-330.
2. Tinetti G, *et al*. The science of ARIEL (Atmospheric Remote-sensing Infrared Exoplanet Large-survey). Experimental Astronomy. 2018;46(1):135-209.
3. Deheuvels S, *et al*. Measuring stellar rotation with asteroseismology. Astronomy & Astrophysics. 2015;580:A96.
4. Mathur S, *et al*. Magnetic fields in intermediate-mass stars revealed by asteroseismology. Nature Astronomy. 2019;3(7):628-634.
5. Shallue CJ, Vanderburg A. Identifying exoplanets with deep learning: a five-planet resonant chain around Kepler-80 and an eighth planet around Kepler-90. The Astronomical Journal. 2018;155(2):94.
6. Hekker S, Christensen-Dalsgaard J. Bayesian inference in asteroseismology. Annual Review of Astronomy and Astrophysics. 2017;55:101-137.
7. Hekker S, Aerts C. Asteroseismology and exoplanets: listening to the stars and searching for new worlds. Space Science Reviews. 2019;215(5):36.
8. Chaplin WJ, Miglio A. Asteroseismology of solar-type and red-giant stars. Annual Review of Astronomy and Astrophysics. 2013;51:353-392.
9. Aerts C, Christensen-Dalsgaard J, Kurtz DW. Asteroseismology. Berlin: Springer Science & Business Media; c2010.
10. Bedding TR, Kjeldsen H. Asteroseismology. Annual Review of Astronomy and Astrophysics. 2017;55:69-105.
11. Chaplin WJ, Miglio A. Asteroseismology of solar-type and red-giant stars. Annual Review of Astronomy and Astrophysics. 2013;51:353-392.
12. Gilliland RL. Exoplanet observations and characterization with the NASA Kepler Mission. Proceedings of the National Academy of Sciences. 2015;112(16):4864-4871.
13. Hekker S, Christensen-Dalsgaard J. Asteroseismology of red giant stars. Annual Review of Astronomy and Astrophysics. 2017;55:287-340.
14. Huber D, Chaplin WJ, Christensen-Dalsgaard J, Miglio A, Montalbán J. Asteroseismology of stars in the Kepler era and beyond. Publications of the Astronomical Society of Australia. 2015;32:e037.
15. Metcalfe TS. Asteroseismology of stellar magnetic activity. Annual Review of Astronomy and Astrophysics. 2017;55:213-246.