Using Deep Learning to Analyze the Voices of Stars

Thomas M. Boudreaux[1]

[1]Department of Physics, High Point University, One University Parkway, High Point, NC 27268

Abstract

With several new large-scale surveys on the horizon, including LSST, TESS, ZTF, and Evryscope, faster and more accurate analysis methods will be required to adequately process the enormous amount of data produced. Deep learning, used in industry for years now, allows for advanced feature detection in minimally prepared data–sets at very high speeds; however, despite the advantages of this method, its application to astrophysics has not yet been extensively explored. This dearth may be due to a lack of training data available to researchers. Here we generate synthetic data loosely mimicking the properties of rapidly–pulsating hot subdwarf B (sdBV$_r$) stars and compare the performance of different deep learning algorithms, including Artificial Neural Networks and Convolutions Neural Networks, in classifying these synthetic data sets as either pulsators, or not–observed–to–vary stars.

Synthetic Data

We developed an in-house Python suite, astroSynth, to output synthetic light curves; these are created by first generating ephemerides for "targets" and then "observing" each ephemeris with a function, thus loosely mimicking how telescopes observe. These light curves can show properties consistent with either pulsating stars, or not-observed-to-vary (NOV) stars. Ephemerides are created through the summation of sine waves with Poisson noise. An example of an astroSynth data product (Figure 1) shows the generated light curve (bottom panel) and associated Lomb-Scargle Periodogram (top panel).

We generate two different datasets for classification. The first dataset (hereafter d–I) contains 100,000 light curves built from *continuous* observations. The second dataset (hereafter d–II) contains 100,000 light curves built from *non–continuous* observations. In both d–I and d–II, half of the light curves show signals with frequencies and amplitudes consistent with those of sdBV$_r$ stars (f=833.3-16670μHz and A=0-20ppt; Heber 2016). The other half of the light curves contain only Poisson noise.

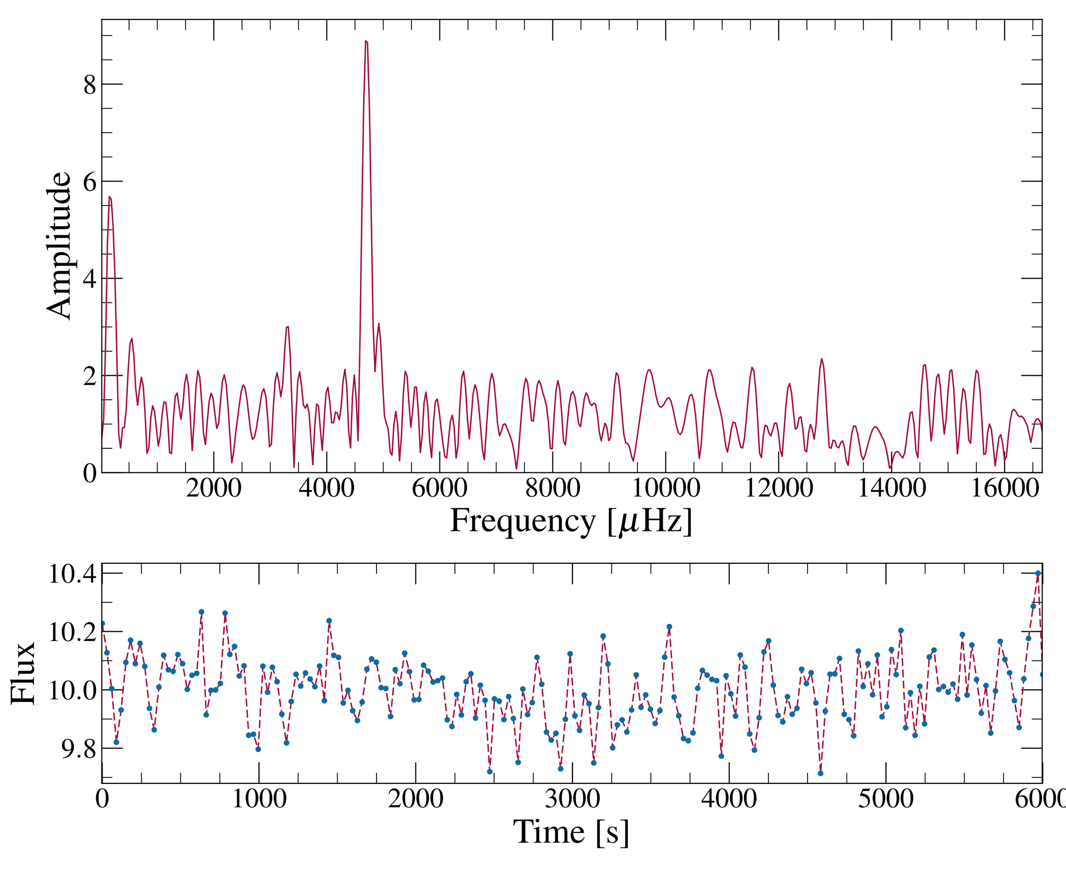

Fig. 1: Example of a synthetic light curve generated by astroSynth (bottom), along with its associated Lomb-Scargle Periodogram (Top)

Artificial Neural Network Performance

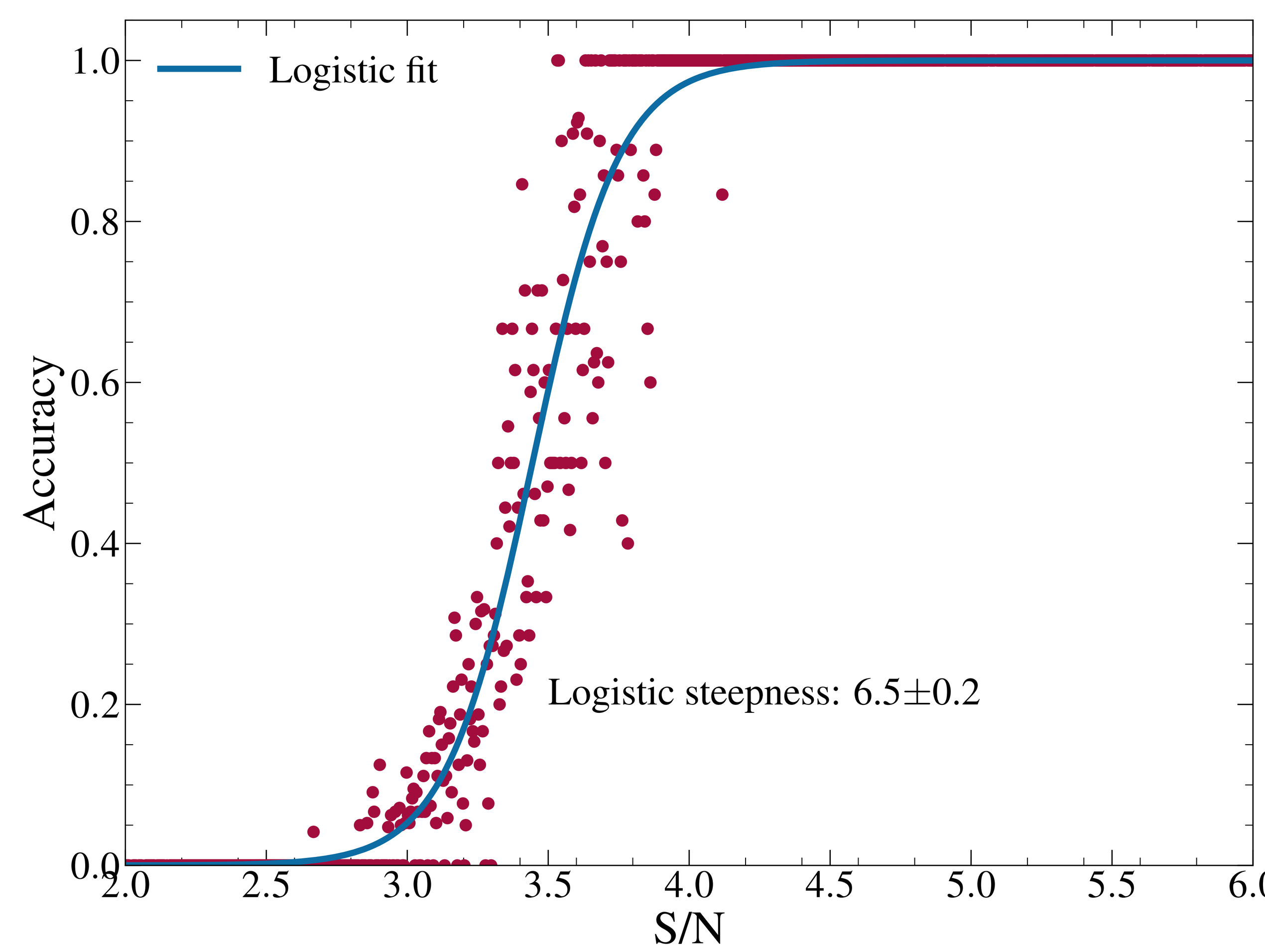

Fig. 2: Artificial Neural Network Performance vs. S/N Level

An evolution of the perceptron (Rosenblatt 1958), the artificial neural network (ANN) was an early kind of neural network that gained widespread usage. ANNs generally take a one dimensional input vector of a predefined size, perform a number of matrix multiplications against weight matrices, and apply non-linear "activation" functions to the results of these multiplications. Like all forms of supervised machine learning, ANNs must be trained. This takes the form of modifying the values of the weight matrices. Once training is complete, the network can be used for its intended purpose.

Here we construct an ANN which expects a length 503 input vector and has one output. The input vector includes the LSP, using 500 frequency bins, of each light curve, as well as the maximum peak in the LSP, median value of the LSP, and frequency of maximum peak in the LSP. The network's output is trained using 80 percent of the synthetic data set d–I, to return either a 1 for pulsating stars, or a 0 for NOV stars. Using this network we achieve a classification accuracy of 90 percent, down to a signal–noise–level of 3.44 σ (Figure 2) on the remaining 20 percent of our data set. It should be noted that with this network we did not conduct extensive hyperparameter tuning, and that if this tuning were to be done it is likely that an ANN could perform better than the one presented here.

We apply this ANN model to the classification of light curves of all known sdB stars in the GALEX mission database (Boudreaux et al. 2017). Of the 5 known sdBV$_r$ stars in this dataset, 4 are successfully identified, while one was incorrectly classed as NOV. Additionally, the GALEX data used has multiple large systematic aliases, and is in general quite noisy; consequently, many targets which are not classified as pulsating stars were incorrectly classified as pulsating stars by this network.

Convolutional Neural Network Performance

Artificial Neural Networks, presented above, are well suited for the analysis of one-dimensional data; however, because observations of real targets often involve data with large time gaps, these observations can be re-factored into two–dimensional data.

In other fields, significant research has been conducted with convolutional neural networks (CNN). CNNs are inspired by the mechanism which animals use to process visual information (Schmidhuber 2017). This network paradigm has proven well–suited for analysis of two-dimensional data. As d–II has non–continuous observations, we elect to break each light curve into multiple "sub–light curves", breaking on the gaps between observations. We then take the LSP of each sub–light curve, and stack those LSPs to generate a sliding FT. This sliding FT is then passed onto a CNN for analysis.

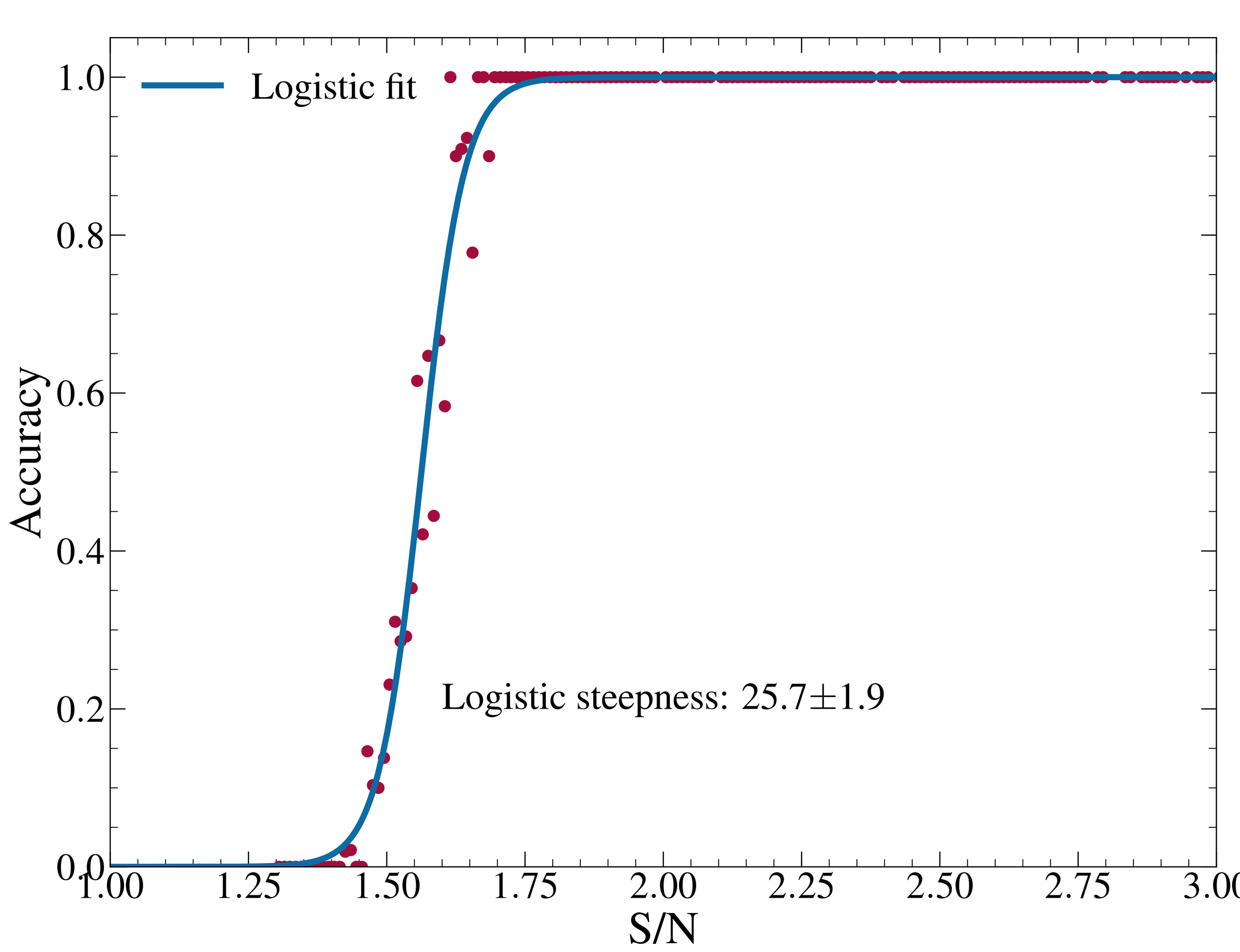

Fig. 3: Convolutional Neural Network Performance vs. S/N Level

We use 80 percent of d–II to train the network, and then test the network's performance using the remaining 20 percent of data. Figure 3 shows how the CNNs performance scales with S/N. This model achieves a 90 percent accuracy down to 1.56σ. One should also note that, when compared to the ANN, the CNN has a much steeper increase in accuracy. The increase in performance that the CNN shows over the ANN is not necessarily surprising as the CNN has the ability to detect if a signal is stable through time in a way that the ANN cannot.

Recurrent Neural Networks

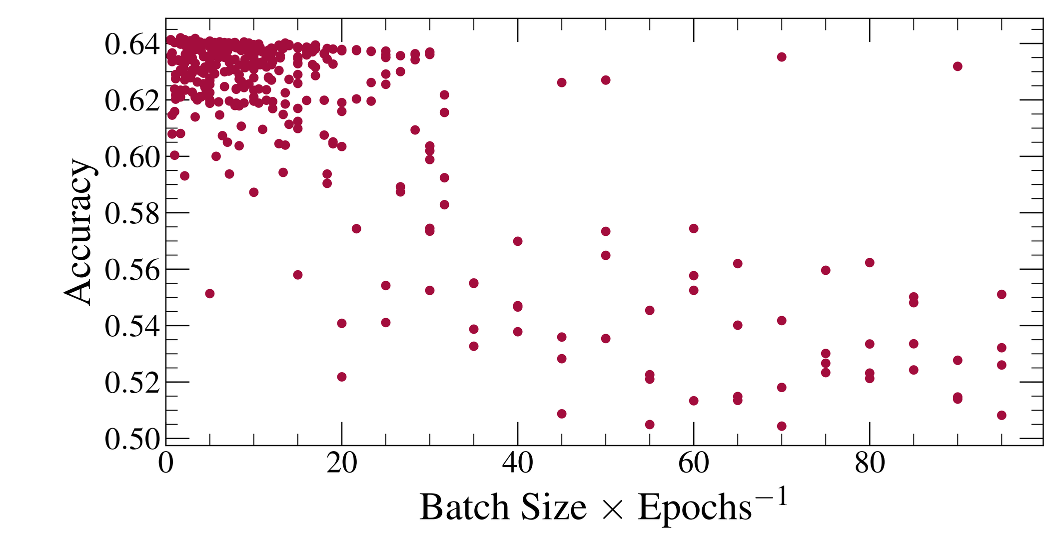

Fig. 4: Hyperparameter tuning results for an RNN investigating the ratio of batch size to epochs. Note that more than two parameters were tuned simultaneously.

The two network paradigms presented here, ANN and CNN, require a constant-sized input vector; however, it is unfeasible to guarantee that light curves will always be made up of the same number of observations. Consequently, when analyzing with either the ANN or CNN we first compute the LSP of the light curve with a **fixed** number of frequency bins (500). This allows us to simultaneously exaggerate the features that are most important for pulsation detection (periodic signals) and to guarantee a constant-sized input vector. Computing LSPs is, however, a relatively expensive process. Frequency space also loses phase information, which could conceivably be useful in breaking certain degeneracies between "true" signals and noise.

Recurrent Neural Networks (RNNs), which have generated significant excitement in machine learning literature recently, may provide a way to analyze time series data **directly**. We have built a set of recurrent networks to analyze our synthetic data in time space; however, even with significant hyperparameter tuning (Figure 4) we are unable to approach the classification accuracy seen with either the ANN or CNN.

Acknowledgments

This research has made use of NASA's Astrophysical Data System (ADS). I would not have been able to conduct this work without the unwavering support of my wonderful adviser; thank you Brad. This research was also made possible by the generous support of the High Point University (HPU) Department of Research and Creative Works (URCW), the HPU Department of Physics, the HPU Department of Math & Computer Science.

References

1. U. Heber. Hot Subluminous Stars. Publications of the Astronomical Society of the Pacific, 128(8):082001, August 2016. 10.1088/1538-3873/128/966/082001.
2. Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. Psychological review, 65(6):386, 1958.
3. T. M. Boudreaux, B. N. Barlow, S. W. Fleming, A. Vasquez Soto, C. Million, D. E. Reichart, J. B. Haislip, T. R. Linder, and J. P. Moore. A Search for Rapidly Pulsating Hot Subdwarf Stars in the GALEX Survey. Astrophysical Journal, 845:171, August 2017. 10.3847/1538-4357/aa8263.
4. Jürgen Schmidhuber. Deep learning in neural networks: An overview. Neural Networks, 61:85 117, 2015. ISSN 0893-6080. https://doi.org/10.1016/j.neunet.2014.09.003. URL http://www.sciencedirect.com/science/article/pii/S0893608014002135