nature astronomy

LETTERS

https://doi.org/10.1038/s41550-017-0321-z

A recurrent neural network for classification of unevenly sampled variable stars

Brett Naul[1]*, Joshua S. Bloom[1], Fernando Pérez[2,3,4] and Stéfan van der Walt[3]

Astronomical surveys of celestial sources produce streams of noisy time series measuring flux versus time ('light curves'). Unlike in many other physical domains, however, large (and source-specific) temporal gaps in data arise naturally due to intranight cadence choices as well as diurnal and seasonal constraints[1–5]. With nightly observations of millions of variable stars and transients from upcoming surveys[4,6], efficient and accurate discovery and classification techniques on noisy, irregularly sampled data must be employed with minimal human-in-the-loop involvement. Machine learning for inference tasks on such data traditionally requires the laborious hand-coding of domain-specific numerical summaries of raw data ('features')[7]. Here, we present a novel unsupervised autoencoding recurrent neural network[8] that makes explicit use of sampling times and known heteroskedastic noise properties. When trained on optical variable star catalogues, this network produces supervised classification models that rival other best-in-class approaches. We find that autoencoded features learned in one time-domain survey perform nearly as well when applied to another survey. These networks can continue to learn from new unlabelled observations and may be used in other unsupervised tasks, such as forecasting and anomaly detection.

The recurrent neural network (RNN) feature extraction architecture proposed (Fig. 1) consists of two components: an encoder, which takes a time series as input and produces a fixed-length feature vector as output, and a decoder, which translates the feature vector representation back into an output time series. The principal advantages of our architecture over a standard RNN autoencoder[8] are the native handling of the sampling times and the explicit use of measurement uncertainty in the loss function.

Specifically, the autoencoder network is trained with times and measurements as inputs and those same measurement values as outputs. The mean squared reconstruction error of the output sequence is minimized, using backpropagation and gradient descent. In the case where individual measurement errors are available, the reconstruction error at each time step can be weighted (in analogy with standard weighted least squares regression) to reduce the penalty for reconstruction errors when the measurement error is large (see Methods).

The feature vector is then taken to be the last element of the output sequence of the last encoder layer, so its dimension is equal to the number of hidden units in that layer. The fixed-length embedding vector produced by the encoder contains sufficient information to approximately reconstruct the input signal, so it may be thought of as a low-dimensional feature representation of the input data. Although we focus here on an autoencoder model for feature extraction, the decoder portion of the network can also be trained directly to solve classification or regression problems, as described in detail in the Supplementary Information.

To investigate the utility of the automatically extracted features, we train an autoencoder model to reconstruct a set of (unlabelled) light curves and then use the resulting features to train a classifier to predict the variable star class. Here, we use the 50,124 light curves from the All Sky Automated Survey (ASAS) Catalog of Variable stars[2]. The autoencoder is trained using the full set of both labelled and unlabelled light curves and the resulting features are then used to build a model to solve the supervised classification task. We compare the resulting classifier with a model that uses expert-chosen features and demonstrate that the autoencoding features perform at least as well and, in some cases, better than the hand-selected features.

Figure 2 depicts some examples of reconstructed light curves for an embedding of length 64 (the other parameters of the autoencoder are described in the Methods). The examples are chosen to represent the twenty-fifth and seventy-fifth percentiles of reconstruction error to show the range of different qualities of reconstruction. The model can effectively represent light curves that exhibit relatively smooth behaviour, even in the presence of large gaps between samples; however, curves that vary more rapidly (that is, those with small periods) are less likely to be reconstructed accurately. As such, we also trained an autoencoder model on the period-folded values (replacing time with phase) using the measured periods[9]. Figure 2 shows that the resulting reconstructions are improved, especially in the case of the low-period signal. The effect of period on the accuracy of autoencoder reconstructions is explored further using simulated data (see Supplementary Information). In what follows, we use autoencoders trained on period-folded light curve data.

To evaluate the usefulness of our features for classification, we identify a subset of ASAS light curves from five classes of variable stars (see Methods for details). We then train a random forest classifier[10] using the autoencoder-generated features for 80% of the samples from each class, along with the means and standard deviations of each light curve, which are removed in preprocessing (see Methods). The resulting estimator achieves 98.8% average accuracy on the validation sets across five 80/20 train/validation splits.

As a baseline, we also constructed random forest classifiers using two sets of standard features for variable star classification. The first are the features used in Richards et al.[11] (henceforth referred to as 'Richards et al. features'). These features are implemented in the Cesium project[12] and also as part of the Feature Analysis for Time Series package[13]. These features have been used by numerous studies (for example, refs [14–16]), including state-of-the-art classification performance on the ASAS survey, and remain competitive against

[1]Department of Astronomy, University of California, Berkeley, CA, USA. [2]Department of Statistics, University of California, Berkeley, CA, USA. [3]Berkeley Institute for Data Science, University of California, Berkeley, CA, USA. [4]Department of Data Science and Technology, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA. *e-mail: bnaul@berkeley.edu

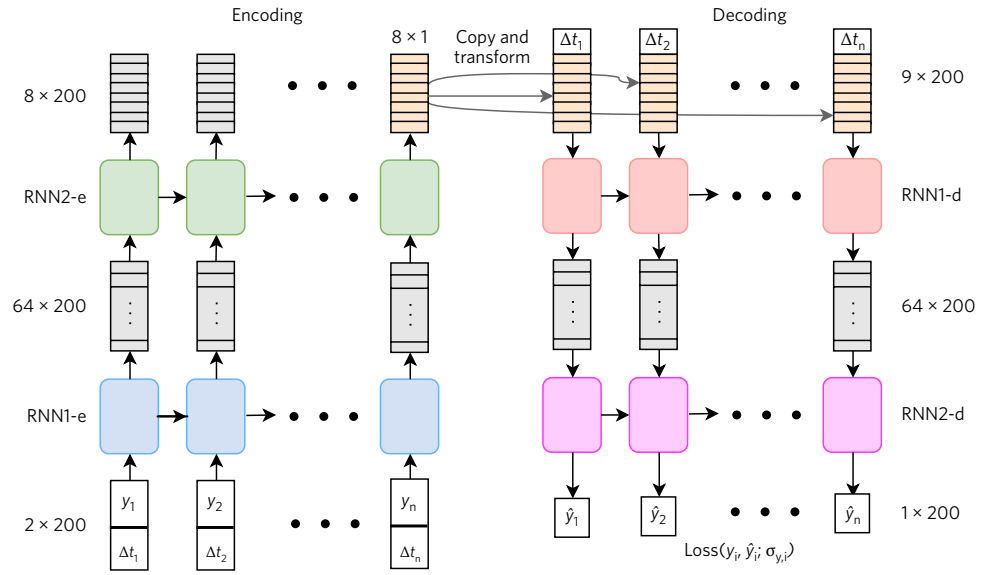

Fig. 1 | Diagram of an RNN encoder–decoder architecture for irregularly sampled time series data. This network uses two RNN layers (specifically, bidirectional gated recurrent units[28,29]) of size 64 for encoding and two for decoding, with a feature embedding size of 8. The encoder takes as inputs the measurement values as well as the sampling times (more specifically, the differences between sampling times). The sequence is processed by a hidden recurrent layer to produce a new sequence, which can then be used as the input to another hidden recurrent layer, and so on. The fixed-length embedding is constructed by passing the output of the last recurrent layer into a single fully connected layer with linear activation function and the desired output size. The decoder first repeats the fixed-length embedding n_T times, where n_T is the length of the desired output sequence, and then appends the sampling time differences to the corresponding elements of the resulting vector sequence. The sampling times are passed to both the encoder and decoder; the feature vector characterizes the functional form of the signal, but the sampling times are needed to determine the points at which that function should be evaluated. The remainder of the decoder network is another series of recurrent layers, with a final linear layer to generate the output sequence. We also apply 25% dropout[30] between recurrent layers, which we omit from the figure for simplicity. In our model, we take the number and size of recurrent layers in the encoder and decoder modules to be equal, but in general the two components are entirely distinct and need not share any architectural similarities.

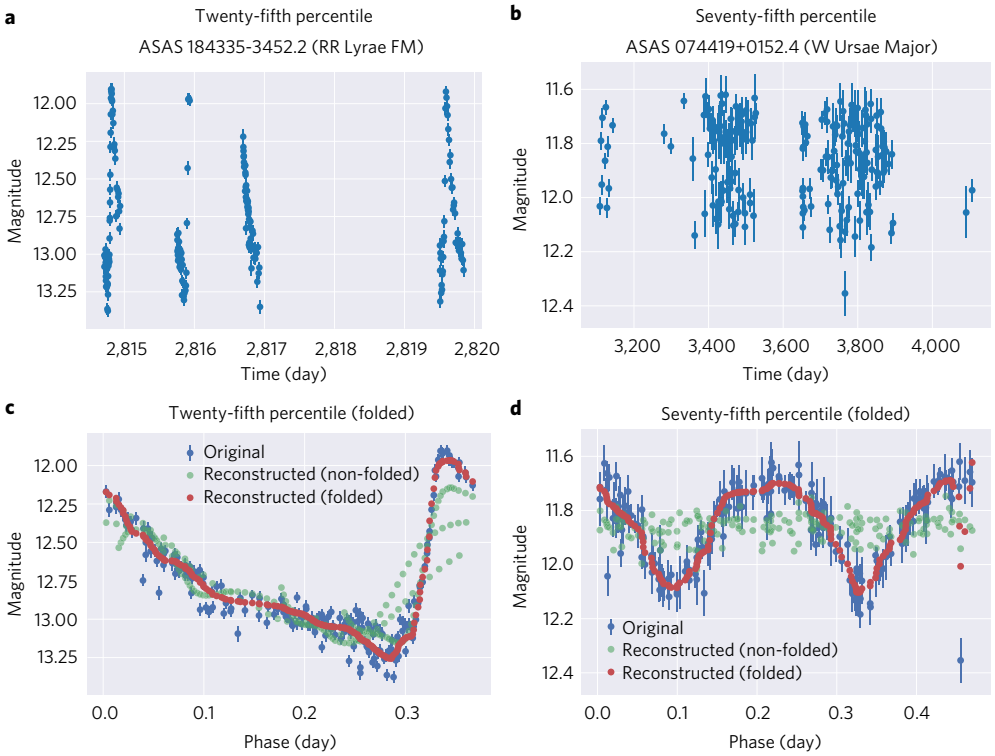

Fig. 2 | Example autoencoder reconstructions of ASAS light curves from 64-dimensional feature representation. a–d, Twenty-fifth and seventy-fifth percentile error reconstructions are shown for raw, unfolded light curves (**a** and **b**) and period-folded light curves (**c** and **d**). The ~V-band magnitudes are in the Vega system.

other classification methods[17]. The second set of features consists of those used by Kim and Bailer-Jones[18] (henceforth referred to as 'Kim and Bailer-Jones features') and implemented in the UPSILoN package. Some features are shared between the two and each set of features is an aggregation of features from many different works. In both cases, we use the same hyperparameter selection technique described in Fig. 3.

The Richards et al. features achieve the best average validation accuracy at 99.4% across the same five splits, as shown in Table 1. However, it is worth noting that the same features were also used in the labelling of the training set[9], so it is not surprising that they achieve almost perfect classification accuracy for this problem. The Kim and Bailer-Jones features, which may provide a more natural baseline, achieve 98.8% validation accuracy, comparable to that of the autoencoder model.

Our second example applies the same feature extraction methodology to variable star light curves from the LIncoln Near-Earth Asteroid Research (LINEAR) survey[19,20]. The LINEAR dataset consists of 5,204 labelled light curves from five classes of variable star (see Methods for details). Unlike in the ASAS example above, here all the available light curves are labelled, so there are no additional unlabelled data to leverage to improve the quality of the extracted features. We find that the autoencoder features outperform the Richards et al. features by 0.38% and the Kim and Bailer-Jones features by 1.61% (see Table 1). In particular, the autoencoder-based model correctly classifies all but one RR Lyrae, suggesting that perhaps some autoencoder features could help improve the performance of the Richards et al. or Kim and Bailer-Jones features for that specific discrimination task.

Finally, we followed the same procedure to train an autoencoder and, subsequently, a random forest classifier to predict the classes of 21,474 variable stars from the MAssive Compact Halo Object (MACHO) catalogue[21]. Once again, our autoencoder approach achieves the best validation accuracy of the three methods considered, averaging 93.6% compared with 90.5% and 89.0% for the Richards et al. and Kim and Bailer-Jones feature sets, respectively.

As shown, training an autoencoder RNN to represent time series sequences can produce informative high-level feature representations in an unsupervised setting. Rather than require fixed-length time series, our approach naturally accommodates variable length inputs. Other unsupervised techniques for feature extraction on irregular time-series data have also been developed (for example, clustering methods[22]), but these scale quadradically in the number of training examples, whereas our approach scales linearly. Moreover, our approach explicitly accounts for measurement noise.

Table 1 | Validation accuracies (mean ± standard deviation across five stratified cross-validation splits) for the autoencoder, Richards et al. and Kim and Bailer-Jones feature random forests

	Dataset		
Method	**ASAS**	**LINEAR**	**MACHO**
Autoencoder	98.77 ± 0.39%	**97.10 ± 0.60%**	**93.59 ± 0.15%**
Richards et al.	**99.42 ± 0.27%**	96.72 ± 0.67%	90.50 ± 0.22%
	(+0.66 ± 0.49%)	(−0.37 ± 0.45%)	(−2.74 ± 0.16%)
Kim and Bailer-Jones	98.83 ± 0.33%	95.49 ± 0.83%	88.98 ± 0.19%
	(+0.06 ± 0.32%)	(−1.60 ± 0.59%)	(−4.60 ± 0.16%)

Note that the training labels for the ASAS data were identified in part using the Richards et al. features. In parentheses are the mean and s.d. of the differences in accuracy; a negative value means the autoencoder performed better over the cross-validation folds. Bold text shows the best result for each dataset.

The resulting features are shown to be comparable or better for supervised classification tasks than traditional hand-coded features. As new sources accumulate without known labels, the unsupervised and on-line nature of such networks should ensure continued model improvements in a way not possible with a fixed number of features. When metadata are also available (for example, colour and sky position for astronomical variables), features derived from such non-temporal data can be easily used in conjunction with the auto-encoded features of the time series.

While the autoencoder approach we have described is well-suited to tasks involving a relatively large amount of (labelled or unlabelled) data, future research should study the efficacy of transfer learning, where feature representations learned from a very large dataset are applied to another problem where fewer examples are available. Another promising application is unsupervised data exploration, such as clustering; autoencoding features could be used as a generic lower-dimensional representation like t-distributed Stochastic Neighbour Embedding (t-SNE[23]) to identify outliers or anomalies in new data. Autoencoders could also act as non-parametric interpolators tuned to the domain on which the models are trained. Finally, while we have focused on single-channel time series, the proposed network is easily extensible to multi-channel time series data as well as multi-dimensional time series, such as unevenly sampled sequential imaging.

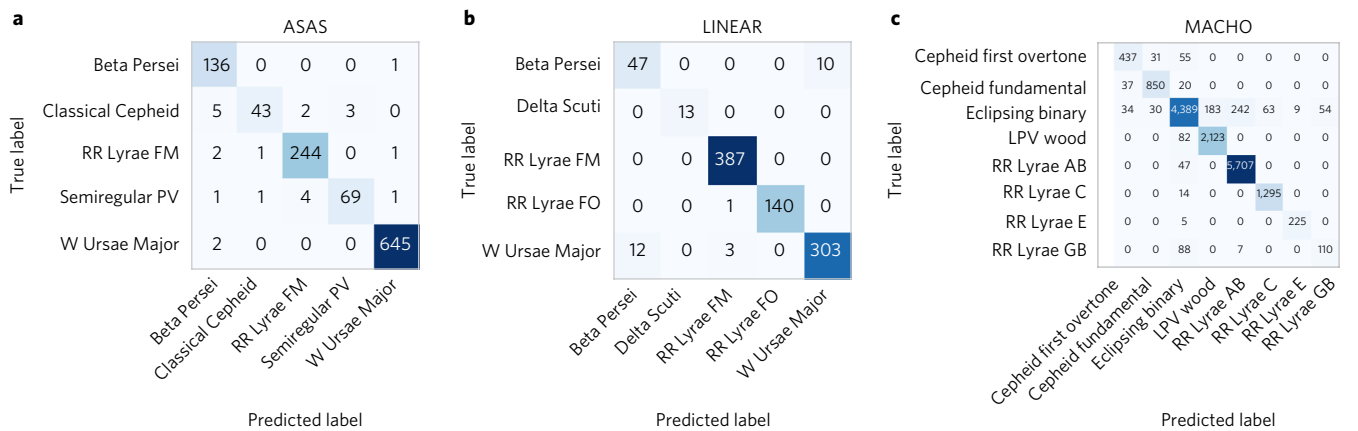

Fig. 3 | Confusion matrices for autoencoder-feature random forest classifiers for labelled variable star light curves for each survey. a, ASAS. **b**, LINEAR. **c**, MACHO. Values along the diagonals are counts of correctly classified light curves and off-diagonal values correspond to incorrect classifications (darker squares correspond to higher counts). LPV, long-period variable.

Methods

We describe here the implementation specifics of the proposed neural network classifier and the detailed properties of the datasets that were used in the above experiments.

Comparison with other neural network approaches. Many of the most commonly used approaches for time series analysis, including standard RNNs, rely on an implicit assumption that the data are uniformly sampled in time. This assumption is not unique to neural network approaches: the fast Fourier transform, which is commonly used in featurization, is only well defined for the evenly sampled, homoskedastic case. Existing neural networks that do allow for uneven sampling operate on interpolations of the data (see, for example, refs [24–26]), thereby replacing the problem with one that can be solved by standard approaches. Any form of interpolation makes implicit assumptions about the spectral structure of the data, which may be unjustified and can end up introducing biases and artefacts. These artefacts increase with sampling unevenness and are ultimately properties of the algorithm, not the data.

Loss function for known measurement errors. Typically an autoencoder is trained to minimize the difference between the input and the reconstructed values, usually in terms of mean squared error. In the case of astronomical surveys, individual measurement errors are generally available for each time step. We therefore constructed a new loss function, which weighs the reconstruction error at each time step more or less heavily when the measurement errors are small or large, respectively (in analogy with standard weighted least squares regression). In particular, our models are trained to minimize the weighted mean squared error (WMSE):

$$\mathrm{WMSE} = \frac{1}{n_\mathrm{T}} \sum_{i=1}^{n} \sum_{j=1}^{n_\mathrm{T}} \left(y_i^{(j)} - \hat{y}_i^{(j)}\right)^2 / \left(\sigma_i^{(j)}\right)^2 \quad (1)$$

where n_T is the length of the sequences, n is the number of light curves and $y_i^{(j)}$, $\hat{y}_i^{(j)}$ and σ_i^j are, respectively, the jth measurement, reconstruction value and measurement error of the ith light curve.

Neural network parameters. The autoencoders used for the ASAS and LINEAR survey classification tasks were constructed using a network architecture like that of Fig. 1, consisting of two encoding and two decoding gated recurrent unit layers of size 96 and an embedding size of 64. The network is trained using the Adam optimizer with a learning rate $\lambda = 5 \times 10^{-4}$ to minimize the weighted mean squared reconstruction error defined in equation (1).

Random forest classifier parameters. In the classification experiments, a random forest classifier is trained to predict the class of each labelled light curve from either the unsupervised autoencoder features from our method or the baseline features from ref. [11]. The hyperparameters of the random forest were chosen by performing a five-fold cross validation grid search over the following grid: $n_\mathrm{trees} \in (50,100,250)$, criterion $\in$ (gini,entropy), max features $\in (3,6,12,18)$, min leaf samples $\in (1,2,3)$. In each case, 80% of the available samples are used as training data and 20% are withheld as test data (the hyperparameter selection is performed only using the training data). The same splits are used for the autoencoder and Richards et al. features and in each case we present the results across five different choices of train/test splits.

ASAS data. The ASAS Catalog of Variable stars[2] consists of 50,124 (mostly unlabelled) variable star light curves. For the ASAS dataset, we select 349 Beta Persei, 130 classical Cepheids, 798 fundamental mode RR Lyrae, 184 semiregular periodic variable and 181 W Ursae Major class stars. The class label of each star is either manually identified or predicted with high probability (>99%) by the Machine-learned ASAS Classification Catalog and periods used in period-folding were identified programmatically as described in ref. [9].

LINEAR data. The LINEAR dataset consists of 5,204 light curves from five classes of star: 2,234 fundamental mode RR Lyrae, 1,860 W Ursae Major, 749 first overtone RR Lyrae, 291 Beta Persei and 70 Delta Scuti. All of the light curves in the LINEAR dataset were manually classified and periods used for period-folding were validated manually as described in ref. [20].

MACHO project data. The MACHO dataset consists of 21,474 light curves from eight classes of star: 7,405 RR Lyrae AB, 6,835 eclipsing binary, 3,049 long-period variable wood (subclasses A–D were combined into a single superclass), 1,765 RR Lyrae C, 1,185 Cepheid fundamental, 683 Cepheid first overtone, 315 RR Lyrae E and 237 RR Lyrae/GB blend. All models were trained on brightness and error values from the red band, although the same approaches could be used on the blue band or both bands simultaneously. Periods and labels were determined using a semi-automated procedure as described in ref. [21]. The full MACHO dataset also contains light curves from many more non-variable sources, which were not used in training either the autoencoder or the random forests in our experiments.

Data preprocessing. First, we processed the raw data from each survey as described in ref. [9], removing low-quality measurements (those with grade C or below) so that each source consists of a series of observations of time, brightness (V-band magnitude) and measurement error values, denoted (t_i, y_i and σ_i). Before training, we further preprocessed the data by centring and scaling each light curve to have mean of zero and standard deviation of one. We also manually removed light curves from our autoencoder training set, which did not exhibit any notable periodic behaviour, by computing a 'super smoother'[27] fit for each light curve with a period equal to the estimated period from ref. [9] and omitting light curves with a residual greater than 0.7 (we observed that including aperiodic sources tended to degrade the quality of the reconstructions). Finally, we partitioned the light curves into sub-sequences of length 200; this is not strictly necessary since our recurrent architecture allows for input and output sequences of arbitrary length, but the use of sequences of equal length is computationally advantageous and reduces the training time (see Supplementary Information for details). The resulting dataset consists of 33,103 total sequences of length $n_\mathrm{T} = 200$.

Data availability. All data and code for reproducing the above experiments is available online at https://github.com/bnaul/IrregularTimeSeriesAutoencoderPaper, including Python code implementing the simulations, Jupyter notebooks for reproducing the figures and trained autoencoder models and weights.

Received: 30 May 2017; Accepted: 24 October 2017;
Published online: 27 November 2017

References

1. Levine, A. M. et al. First results from the All-Sky Monitor on the Rossi X-Ray Timing Explorer. *Astrophys. J. Lett.* **469**, L33–L36 (1996).
2. Pojmanski, G. The All Sky Automated Survey. Catalog of variable stars. I. 0^h–6^h quarter of the southern hemisphere. *Acta Astronomica* **52**, 397–427 (2002).
3. Murphy, T. et al. VAST: an ASKAP survey for variables and slow transients. *Publ. Astron. Soc. Aust.* **30**, e006 (2013).
4. Ridgway, S. T., Matheson, T., Mighell, K. J., Olsen, K. A. & Howell, S. B. The variable sky of deep synoptic surveys. *Astrophys. J.* **796**, 53 (2014).
5. Djorgovski, S. et al. Real-time data mining of massive data streams from synoptic sky surveys. *Future Gener. Comput. Syst.* **59**, 95–104 (2016).
6. Kantor, J. Transient alerts in LSST. in *The Third Hot-wiring the Transient Universe Workshop* (eds Wozniak, P. R., Graham, M. J., Mahabal, A. A. and Seaman, R.) 19–26 (Los Alamos National Laboratory, 2014).
7. Bloom, J. S., & Richards, J. W. Data mining and machine learning in time-domain discovery and classification. in *Advances in Machine Learning and Data Mining for Astronomy* (eds Way, M. J., Scargle, J. D., Ali, K. M. and Srivastava, A. N.) 89–112 (CRC, New York, 2012).
8. Hinton, G. E. & Salakhutdinov, R. R. Reducing the dimensionality of data with neural networks. *Science* **313**, 504–507 (2006).
9. Richards, J. W. et al. Construction of a calibrated probabilistic classification catalog: application to 50k variable sources in the All-Sky Automated Survey. *Astrophys. J. Suppl. Ser.* **203**, 32 (2012).
10. Breiman, L. Random forests. *Mach. Learn.* **45**, 5–32 (2001).
11. Richards, J. W. et al. On machine-learned classification of variable stars with sparse and noisy time-series data. *Astrophys. J.* **733**, 10 (2011).
12. Naul, B., van der Walt, S., Crellin-Quick, A., Bloom, J. S. & Pérez, F. Cesium: open-source platform for time-series inference. in *Proc. 15th Python in Science Conf.* (eds Benthall, S. and Rostrup, S.) 27–35 (SciPy, Austin, TX, 2016).
13. Nun, I. et al. FATS: Feature Analysis for Time Series. Preprint at https://arxiv.org/abs/1506.00010 (2015).
14. Dubath, P. et al. Random forest automated supervised classification of Hipparcos periodic variable stars. *Mon. Notices R. Astron. Soc.* **414**, 2602–2617 (2011).
15. Nun, I., Pichara, K., Protopapas, P. & Kim, D.-W. Supervised detection of anomalous light curves in massive astronomical catalogs. *Astrophys. J.* **793**, 23 (2014).
16. Miller, A. A. et al. A machine-learning method to infer fundamental stellar parameters from photometric light curves. *Astrophys. J.* **798**, 122 (2015).
17. Kügler, S. D., Gianniotis, N. & Polsterer, K. L. Featureless classification of light curves. *Mon. Not. R. Astron. Soc.* **451**, 3385–3392 (2015).
18. Kim, D.-W. & Bailer-Jones, C. A. A package for the automated classification of periodic variable stars. *Astron. Astrophys.* **587**, A18 (2016).
19. Sesar, B. et al. Exploring the variable sky with LINEAR. II. Halo structure and substructure traced by RR Lyrae stars to 30 kpc. *Astron. J.* **146**, 21 (2013).
20. Palaversa, L. et al. Exploring the variable sky with LINEAR. III. Classification of periodic light curves. *Astron. J.* **146**, 101 (2013).

21. Alcock, C. et al. The MACHO project LMC variable star inventory. II. LMC RR Lyrae stars—pulsational characteristics and indications of a global youth of the LMC. *Astron. J* **111**, 1146–1155 (1996).
22. Mackenzie, C., Pichara, K. & Protopapas, P. Clustering-based feature learning on variable stars. *Astrophys. J.* **820**, 138 (2016).
23. Van der Maaten, L. & Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).
24. Charnock, T. & Moss, A. Deep recurrent neural networks for supernovae classification. Preprint at https://arxiv.org/abs/1606.07442 (2016).
25. Che, Z., Purushotham, S., Cho, K., Sontag, D. & Liu, Y. Recurrent neural networks for multivariate time series with missing values. Preprint at https://arxiv.org/abs/1606.01865 (2016).
26. Lipton, Z. C., Kale, D. C., Elkan, C. & Wetzell, R. Learning to diagnose with LSTM recurrent neural networks. Preprint at https://arxiv.org/abs/1511.03677 (2015).
27. Friedman, J. H. & Silverman, B. W. Flexible parsimonious smoothing and additive modeling. *Technometrics* **31**, 3–21 (1989).
28. Cho, K. et al. Learning phrase representations using RNN encoder–decoder for statistical machine translation. Preprint at https://arxiv.org/abs/1406.1078 (2014).
29. Schuster, M. & Paliwal, K. K. Bidirectional recurrent neural networks. *IEEE Trans. Sig. Process.* **45**, 2673–2681 (1997).
30. Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014).

Acknowledgements

We thank Y. LeCun and F. El Gabaly for helpful discussions and A. Culich for computational assistance. This work is supported by the Gordon and Betty Moore Foundation Data-Driven Discovery and National Science Foundation BIGDATA grant number 1251274. Computation was provided by the Pacific Research Platform programme through the National Science Foundation Office of Advanced Cyberinfrastructure (number 1541349), Office of Cyberinfrastructure (number 1246396), University of California Office of the President, Calit2 and Berkeley Research Computing at University of California Berkeley.

Author contributions

B.N. implemented and trained the networks, assembled the machine learning results and generated the first drafts of the paper and figures. J.S.B. conceived of the project, assembled the astronomical light curves and oversaw the supervised training portions. F.P. provided theoretical input. S.v.d.W. discussed the results and commented on the paper.

Competing interests

The authors declare no competing financial interests.

Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s41550-017-0321-z.

Reprints and permissions information is available at www.nature.com/reprints.

Correspondence and requests for materials should be addressed to B.N.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.