Exoplanet Detection with Machine Learning

Charles Zhao and Bilal Mukadam

Background

Machine learning has shown great promise in many scientific disciplines, including astrophysics. The goal of our work is to explore several machine learning methods for automatically detecting stars with orbiting exoplanets. To do this, we use data from the Kepler space telescope, which includes flux (light intensity) measurements from several thousand stars over time, as well as whether each of them has (an) exoplanet(s). Our **first analysis** uses a standard exoplanet detection technique called box least squares. In our **second analysis**, we construct several features from each time series and run these features through several “classic” (i.e. not deep learning) machine learning classifiers. In our **last analysis**, we explore two deep learning methods.

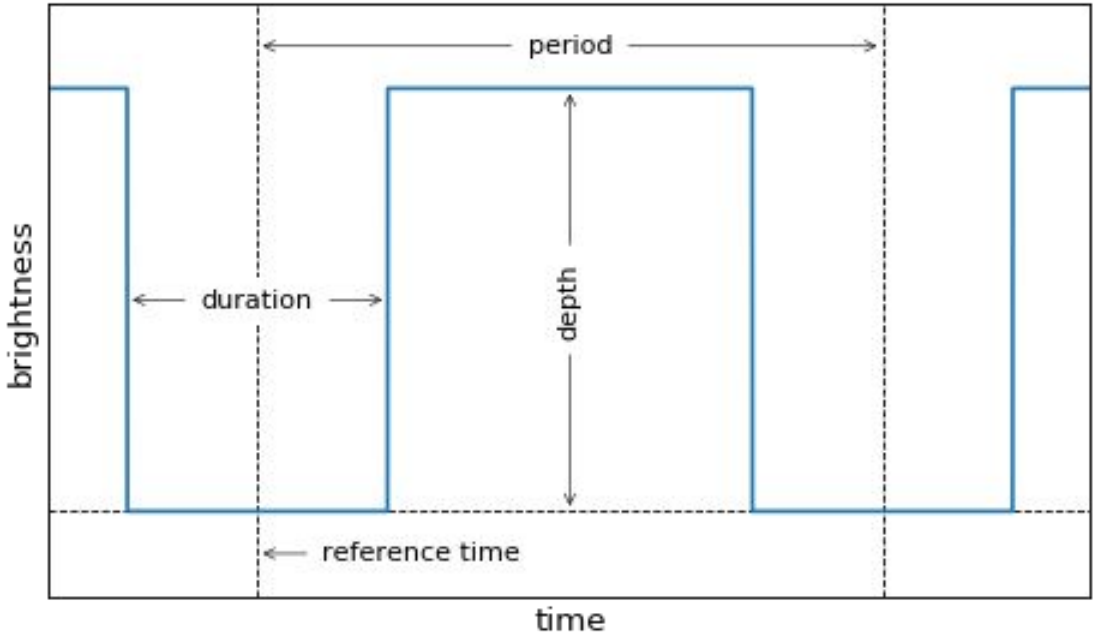
Since an exoplanet should periodically obscure part of the star it orbits, theoretically, one should be able to detect exoplanets using flux time series data. We use data collected by the Kepler space telescope and cleaned by NASA. The training set consists of 3,197 flux measurements for 5,087 stars, with 5,050 without-exoplanet stars and 37 with-exoplanet stars. The test set consists of the same number of measurements for 570 stars, with 565 without-exoplanet stars and 5 with-exoplanet stars.

Related Work

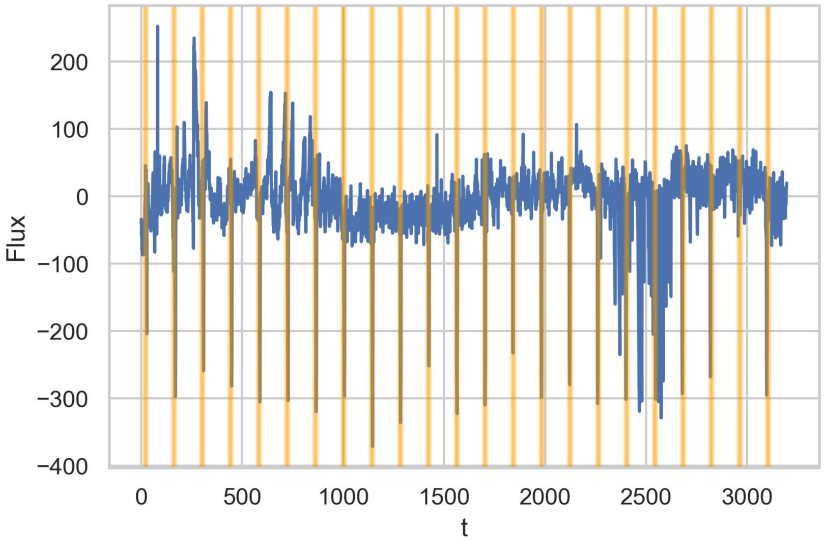
Kovács, Zucker, and Mazeh presented an algorithm called **Box Least Squares (BLS)** for finding periodic transits. This algorithm is now quite standard in exoplanet detection and analysis. BLS models the transit light curve as a periodic box, where the box represents the decrease in flux due to the transiting exoplanet. The authors found that this method works particularly well when the signal-to-noise ratio (SNR) is small and thus the periodic signal can only be detected after measuring the unknown transit many times. They concluded that when the effective SNR exceeds 6, this indicates a significant detection.

Graves and Schmidhuber presented **bidirectional long short-term memory (BLSTM) networks**, which they evaluated on framewise phoneme classification. LSTMs are a variant of recurrent neural networks (RNNs) that give them a kind of long-term memory. BLSTMs are a variant of standard LSTMs that presents each training sequence forwards and backwards to two separate LSTMs, both of which are connected to the same output layer. Therefore, for every point in a given sequence, the network has information about the sequence both before and after the given point. The authors found that bidirectional networks outperform unidirectional ones, and that LSTM is both much faster and more accurate than standard recurrent neural networks.

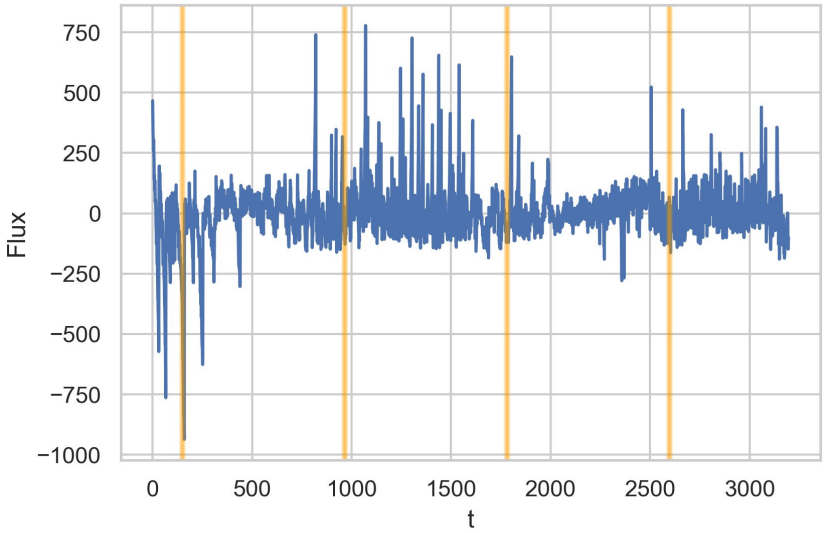
BLS Model



Data Exploration



With-Exoplanet



Without-Exoplanet

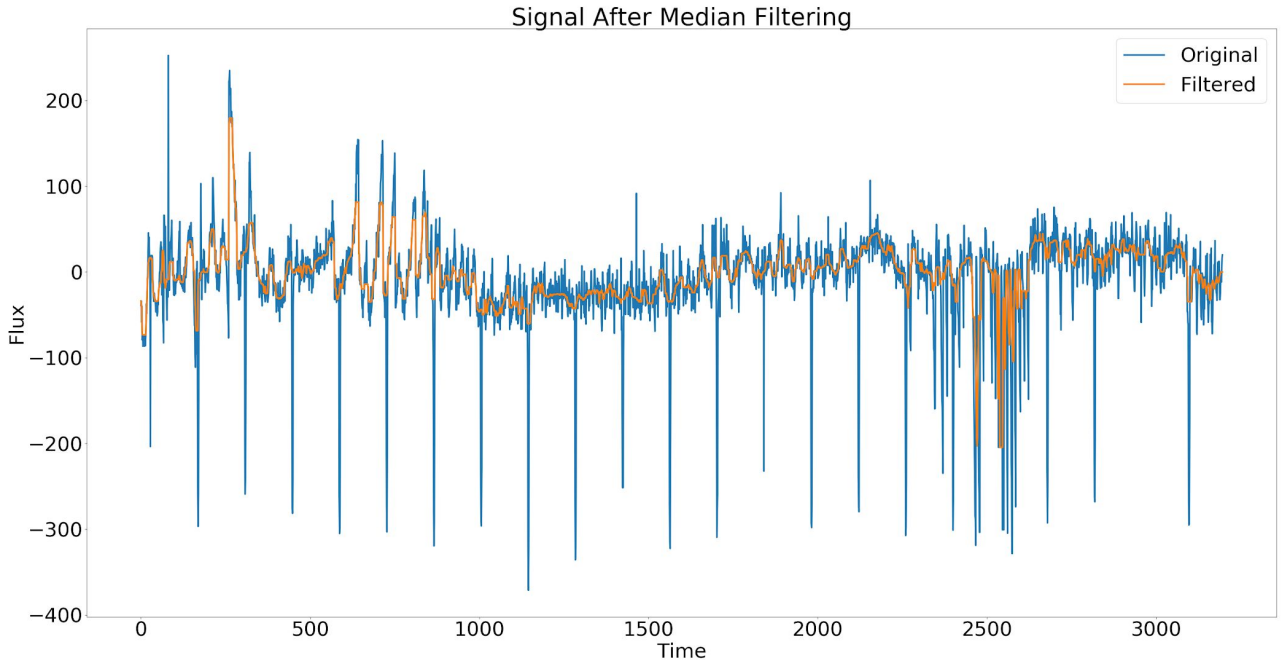
Feature Engineering and Preprocessing

For the classic machine learning methods and the deep learning methods, we first upsample the training data by repeating all the with-exoplanet data points 136 times, resulting in the training set comprising 5050 without-exoplanet and 5032 with-exoplanet stars. We then perform the following transformations using Scipy, PyWavelets, and Astropy on the time series data to construct features for each star:

- Discrete Fourier Transform to determine the 4 strongest frequencies
- Discrete Wavelet Transform to determine the 4 strongest wavelets
- Box Least Squares to determine depths, uncertainties, powers, periods, and durations of exoplanet transits

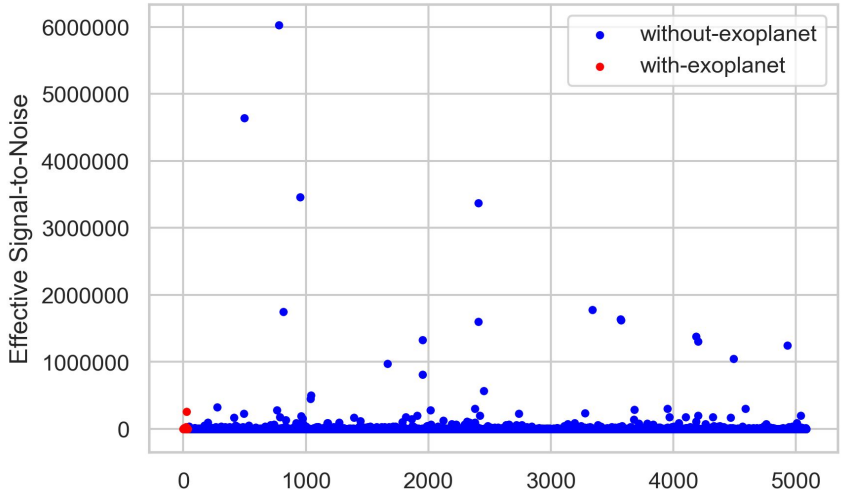
In addition, for some analyses we explored smoothing of the flux signals as a preprocessing step. This was done by using a 12-hour sliding median window, ie. the data in each window was reduced to the median. The 12-hour duration of the window and the median metric (as opposed to mean) were recommended by Professor Joshua Winn (Astrophysics Department) during our discussions with him regarding our project.

Median Filtering - 12 Hour Sliding Window

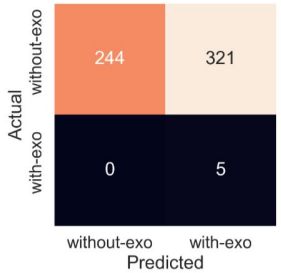


Box Least Squares (BLS) Analysis

The lowest SNR in the training set was 12.24, and the highest was approximately 6 million, which seems to be at odds with the findings of Kovács et al. We believe this is due to running BLS with different parameters. Therefore, we sought to find an optimal SNR threshold ourselves. Unfortunately, there does not seem to be a clear distinction between with-exoplanet and without-exoplanet SNRs. Using the ROC curve on the training set as guidance, we handpicked an SNR threshold of 200 so that an SNR greater than 200 resulted in a with-exoplanet classification, and an SNR below 200 resulted in a without-exoplanet classification.

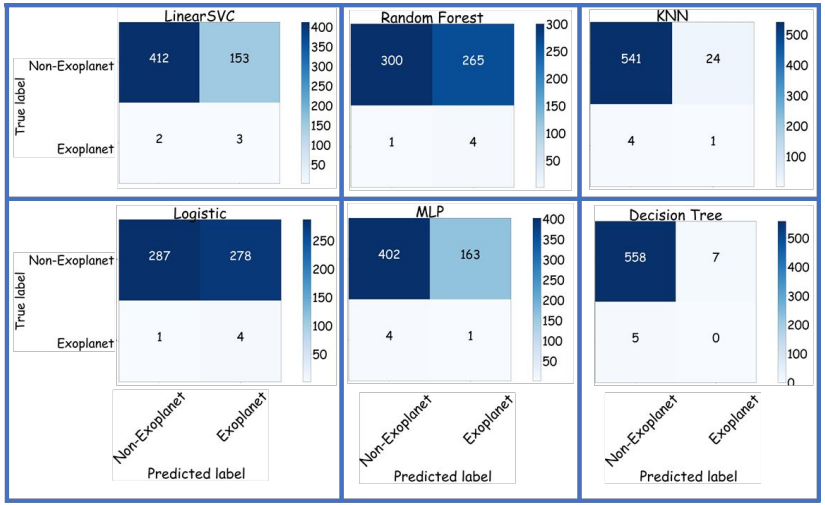


Accuracy	AUC	F1
0.447	0.790	0.031



Classical Classifiers Analysis

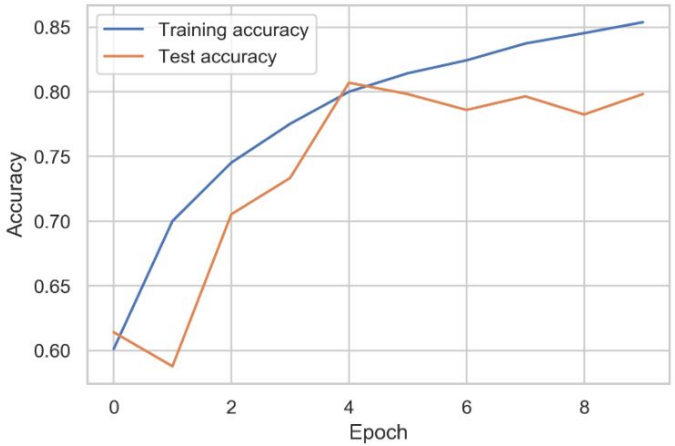
Model	Accuracy	Precision	Recall	F1-Score
SVC	99.12%	0	0	0
LinearSVC	72.81%	0.01923	0.6	0.03727
Random Forest	53.33%	0.01487	0.8	0.0292
Decision Tree	97.89%	0	0	0
MLP	84.21%	0.0107	0.4	0.02083
KNN	95.09%	0.04	0.2	0.06667
Logistic	51.05%	0.01418	0.8	0.02787

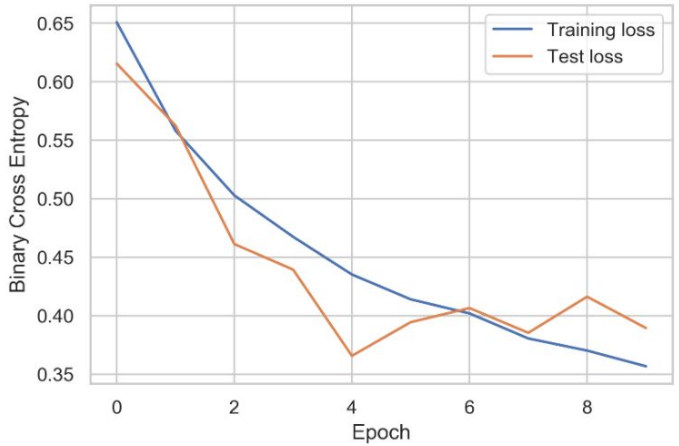


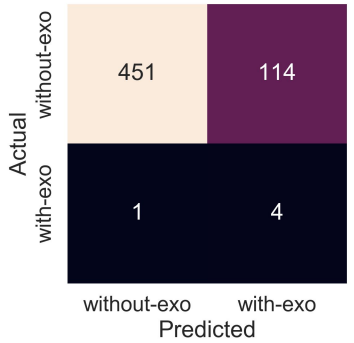
Although models like SVC, Decision Tree, and KNN have high accuracies, they are unable to achieve our ultimate goal, namely that of correctly classifying most/all of the exoplanets in the test sets as exoplanets. In addition, although Random Forest classifies 4/5 exoplanets correctly, it doesn't do great overall (53.33% accuracy).

Deep Learning Analysis - LSTM RNN

1. **Bidirectional LSTM layer** with 4 neurons; outputs the output of the last neuron.
2. **Dense layer** with 4 neurons.
3. **Dropout layer**; drops out 50% of inputs.
4. **Dense layer** with 1 neuron with sigmoid activation; outputs a number between 0 and 1.



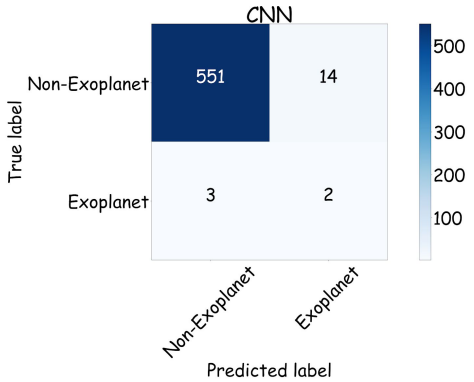




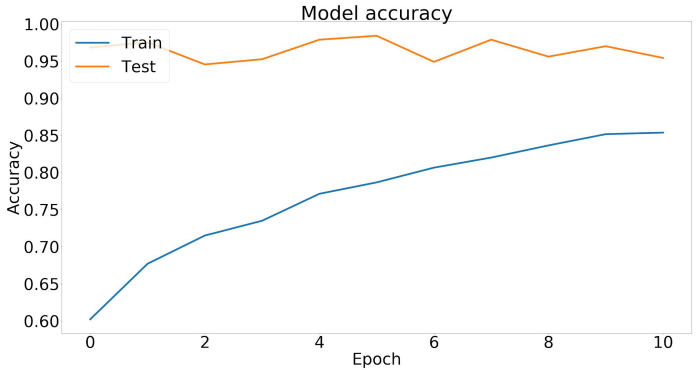
Deep Learning Analysis - Convolutional Neural Network

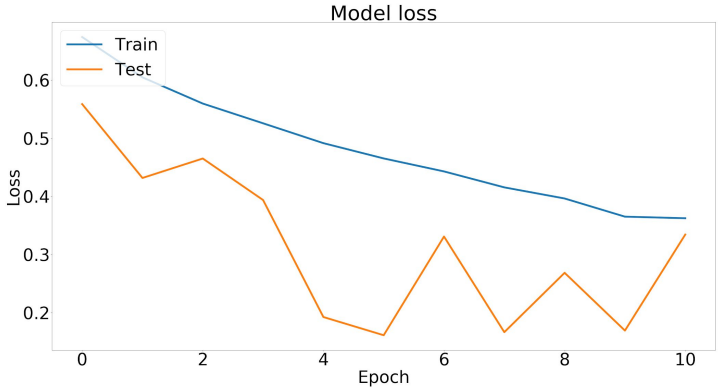
Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 45, 66, 64)	640
max_pooling2d_1 (MaxPooling2	(None, 22, 33, 64)	0
dropout_1 (Dropout)	(None, 22, 33, 64)	0
flatten_1 (Flatten)	(None, 46464)	0
dense_1 (Dense)	(None, 128)	5947520
dropout_2 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 2)	258

Total params: 5,948,418
Trainable params: 5,948,418



Model	Accuracy	AUC	F1
LSTM RNN	0.798	**0.759**	0.065
CNN	**0.971**	0.667	**0.211**





Future Directions

- More sophisticated neural network architectures. For example, Karim et al. were able to achieve state-of-the-art performance in time series classification using a LSTM Fully Convolutional Network, which combines the outputs of an LSTM network and a convolutional network.
- Examine what *kinds* of planets are classified well by our machine learning methods. Our data set did not include information about the stars or exoplanets, but other data sets with such information could help guide the process of feature engineering and model selection.
- Use new data from the Transiting Exoplanet Survey Satellite (TESS), which launched in April 2018 and will be surveying 200,000 of the brightest stars near the sun over the course of two years. It will cover a sky area 400 times larger than that monitored by the Kepler telescope,