HARNESSING THE POWER OF CNNS FOR UNEVENLY-SAMPLED LIGHT-CURVES USING MARKOV TRANSITION FIELD

Margarita Bugueño[1], Gabriel Molina[1], Francisco Mena[1], Patricio Olivares[2] and Mauricio Araya[2]

[1]Depto. Informática, Universidad Técnica Federico Santa María, Santiago, Chile,
[2]Depto. Electrónica, Universidad Técnica Federico Santa María, Valparaíso, Chile

ABSTRACT

Exoplanet detection has evolved from case by case data inspection to automatic pattern recognition methods for processing a very large number of light curves. For this reason, the use of machine learning techniques has become a common practice in the field, where deep learning models are now in the spotlight as a promising leap towards automation. However, despite being faster than manual inspection, they usually still need hand-crafted features to achieve good results. Moreover, not all methods allow real world data where a large portion of the data is missing or at least is not regularly sampled. In this paper, we propose a method that only requires the raw light curve to make an exoplanet classification without the need of additional metadata or specific formats for the time series. We transform the unevenly-sampled time series (light curves) into a 2-channel image using Markov Transition Fields, which feeds a convolutional neural network that classifies candidate transients. We conducted experiments using the Kepler Mission dataset, identifying two key results: (1) the method is competitive in terms of performance to the state-of-the-art alternatives, yet it is simpler and faster and based on this result, we also show that (2) a Markov Transition Field can be used as an effective stand-alone data product for analyzing unevenly-sampled transient light curves.

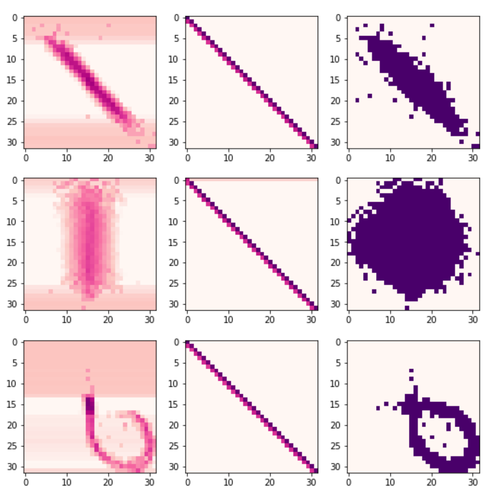
WHAT IS THE PROBLEM?

Despite there is not a clear unique way to detect and confirm exoplanets, the indirect observational methods are the most successful techniques, i.e. *transit photometry* and *radial velocity*. In any case, the variations can be quite slight and measurements are uneven which makes detection difficult.
The increasing popularity of Machine Learning techniques and the large amount of data generated by observatories, allow astronomers to detect exoplanets by reducing effort and time. The common approach used to classify a light curve is to extract hand-crafted *specialized features* from it and apply classic machine learning methods [Ric+11; HTT18] or use problem-learned representations [BMA18]. Deep Learning has been also applied using convolutional neural networks (CNNs) [SV18]. However, even though learning methods are very helpful, most of them rely on hand-crafted features and external metadata to cope with the task. Therefore, a thorough and detailed analysis of the light curves only occurs once the metadata has been collected.

PROPOSAL

The proposed method focuses on detecting transient exoplanets based only on the raw unevenly-sampled light curves. In order to simplifies the representation of long uneven time series, we generate a bi-dimensional matrix of the *semi-continuous* transitions of a time series. By taking advantage of the time information, we build 2 channel image representation: one for the observation measurements and the other one for the observation timestamps. The *semi-continuous* transitions correspond to transitions that have a delta timestamp below some maximum delta, avoiding considering discontinuous measurements under that delta.

FIGURE 1



Examples of patterns observed on Kepler light curves. **Left**: measurement channel, transitions represented as probabilities. **Center**: time channel. **Right**: a binarized representation of the measurement channel (transition/no transition).

CONSTRUCTING THE MTF

The first channel of the representation is build with an adapted version of the Markov Transition Field (MTF) [WO15] with the *semi-continuous* margin. While the second channel represents the time information from the light curve. This allows to extract temporal information from the raw light curve measurements. The meaning of a value m_{ij} in M (the MTF) corresponds to the likelihood of the transition from state i to j where $i, j \in N$. We define the states as a segmentation (linear grid) over the $[-1, 1]$ interval. We set a number for segmentation the positive measurements $[0, 1]$ (with n_{up}) and the negative measurements $[-1, 0]$ (with n_{down}). With this values we handle the detail of the representation, as a trade-off between fine-grained or coarse-grained representation. Finally, the image has a dimension of $N \times N \times 2$, with $N = n_{up} + n_{down}$.
Harnessing the power of deep learning for image processing, specifically we can combine this representation as input of a 2D convolutional neural network, so a feature map of size $k \times k$ will consider k consecutive states.

EXPERIMENTS

We validate our proposal on *Kepler Objects of Interest* (KOI) dataset of Kepler mission using *Confirmed* [C] (2281 objects) and *False Positive* [FP] classes (3976 objects). We use the Kepler detrend pipeline [Fan+11] to obtain a standard light curve representation. For metadata, we selected those features generated from the light curve. We also duplicate the dataset by mirroring each light curve. For the build of MTF we set 45 mins as maximum delta (Kepler standard sampling rate was 30 mins).

TABLE 1

n_{up} \ n_{down}	8	16	32	64
4	71.48	72.66	72.97	71.81
8	72.14	73.35	74.43	72.00
16	72.84	74.69	**75.75**	74.13
32	73.84	75.03	75.40	75.53
64	72.75	73.47	74.03	74.95

Table 1 shows the macro averaged F1-score on validation set for different number of states (n_{up}, n_{down}) on the generated MTF representation. The best *F1-score* was obtained by using $n_{up} = 16$ and $n_{down} = 32$, which is a 2-channel image of 48×48 size. Note that as N increases the performance improves. This is because the model input becomes a fine-grained representation. However, for any n_{up} up to 16 states, the *F1-score* reaches a maximum with $n_{down} = 32$, then it decays. Most of the cases reach better results when a higher number of states on the negative values is set.

TABLE 2

Method	Input shape	FP class	C class	Macro avg.
Specialized hand-crafted features + Classic Learning Methods				
Metadata	10	90.13	83.85	87.00
feets⋆	57	84.57	31.19	57.88
Feature extraction + Classic Learning Methods				
F-PCA	32	80.03	58.54	69.29
Deep Learning Methods				
1D CNN raw	70000×2	84.43	67.68	76.06
2D CNN MTF	$48 \times 48 \times 2$	84.26	69.76	77.01

Table 2 shows the classification performance for different methods and feature extraction techniques. *F1-score* per class and macro averaged *F1-score* are presented. Within the deep learning, our method obtains a moderate improvement of 0.95 *F1* over a 1D CNN. Meanwhile the classic approach of F-PCA is outperform by $\sim$ 8% and the hand-crafted features of feets by a large gap of $\sim$ 33%. Drawing on 2D CNN model, our method extracts enough information to identify transit patterns on the MTF images, being the best option below the metadata.

TABLE 3

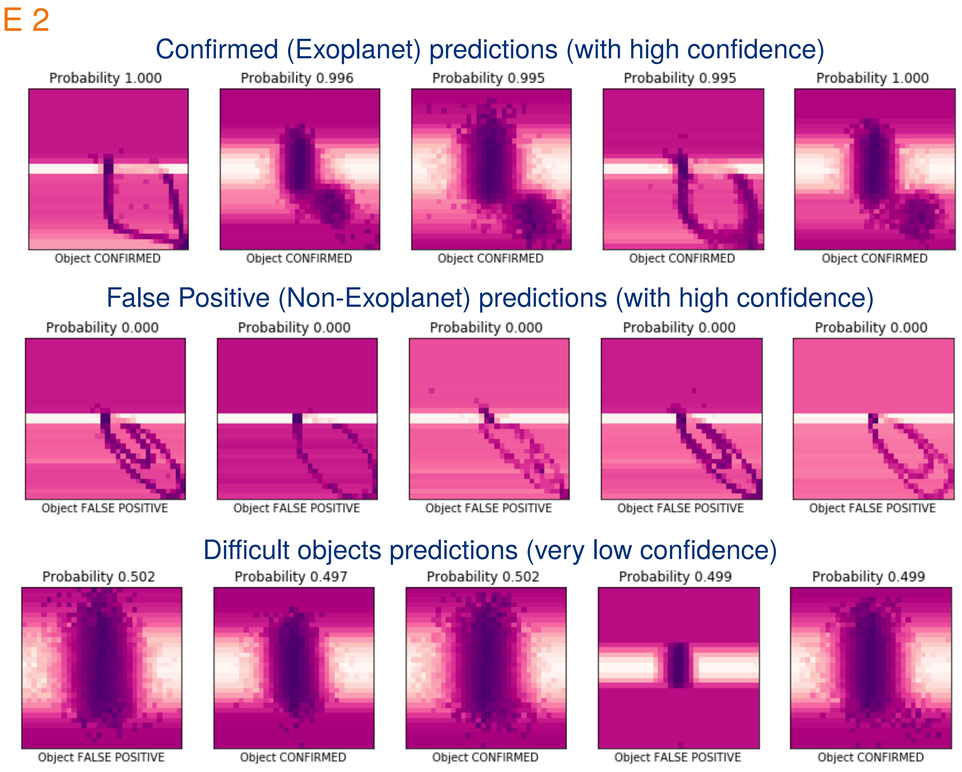
Method	Representation	Training	Predict	Total
Specialized hand-crafted features + Classic Learning Methods				
Metadata	5760	37.8 (200)	0.04	-
feets	5912500⋆	42.2 (200)	0.07	> 100000 mins
Feature extraction + Classic Learning Methods				
F-PCA	166.6	40.4 (200)	0.06	~ 4 mins
Deep Learning Methods				
1D CNN raw	0	21500 (50)	288.33	~ 360 mins
2D CNN MTF	145	1040 (200)	4.72	~ 20 mins

Table 3 shows the execution time (in seconds) by phase: i) representation (MTF), ii) training and iii) predict. The feets representation time is measured over a subset of 2500 objects (30% of the data for 2 months execution).
Our method takes 20 minutes in average for running the complete process. Note that our method focus on the representation but reduce time on the other phases. That is, using a 2D CNN model we can extract relevant information in a faster way, 18 times shorter than 1D CNN: from 6 hours to just 20 minutes as total time.

Figure 2 shows examples of MTF representations for KOI objects that 2D CNN model predicts with differents grades of reliability for Confirmed and False Positive classes. We also show the most difficult objects to classify (probability close to 0.5).

- Standard well defined transits are objects with bottom right ellipse patterns, where the objects predicted as confirmed have more thick pattern than the objects predicted as false positives.
- There seems to be a second ellipse on the false positive objects, probably a binary system.
- For the difficult objects, it can be seen a big spot on the center with no clear inclination to the bottom right. This clarifies the difficulties of classify these types of objects despite the fact that some of them have an orbiting exoplanet.
- The vertical patterns, as the one shows on Figure 1 are associated to random behavior, where there are transitions on almost all the states.
- The eccentricity of the patterns in the MTF are related to the transit period. More circular patterns (*smaller eccentricity*) indicates shorter period (faster transit). While more diagonal patterns (*high eccentricity*) indicates longer period (slower transit).

FIGURE 2



CONCLUSIONS & KEY POINTS

- A 2D matrix representation (image) for 1D unevenly-sampled light curves can be generated and combined with 2D convolutional neural net models.
- Our method is faster to execute and lighter in terms of memory consumption, yet it offers a competitive performance of 77.01% in the *F1-score* (macro averaged).
- The behavior of the exoplanets (Confirmed class) is not so clear in order to group and discriminate correctly all the patterns, while non-exoplanets appears easier to detect.
- The results over the KOI dataset of the Kepler mission takes around 20 minutes to process.
- The 2D fixed-sized representation for raw light curves offers a simpler alternative to visualize and analyze the behavior and periodicity of the transits.

Acknowledgments. Funding of ANID-Basal Project FB0008 (AC3E) and ANID PIA/APOYO AFB180002 (CCTVal).

REFERENCES

[BMA18] Margarita Bugueno, Francisco Mena, and Mauricio Araya. “Refining exoplanet detection using supervised learning and feature engineering”. In: *Latin American Computer Conference (CLEI)*. IEEE. 2018, pp. 278–287.

[Fan+11] MN Fanelli et al. “Kepler Data Processing Handbook (KSCI-19081-001)”. In: *Kepler Project Office* (2011).

[HTT18] Trisha A Hinners, Kevin Tat, and Rachel Thorp. “Machine Learning Techniques for Stellar Light Curve Classification”. In: *The Astronomical Journal* 156.1 (2018), p. 7.

[Ric+11] Joseph W Richards et al. “On machine-learned classification of variable stars with sparse and noisy time-series data”. In: *The Astrophysical Journal* 733.1 (2011), p. 10.

[SV18] Christopher J Shallue and Andrew Vanderburg. “Identifying exoplanets with deep learning: A five-planet resonant chain around kepler-80 and an eighth planet around kepler-90”. In: *The Astronomical Journal* (2018).

[WO15] Zhiguang Wang and Tim Oates. “Imaging time-series to improve classification and imputation”. In: *Twenty-Fourth International Joint Conference on Artificial Intelligence*. 2015.