Searching for disintegrating planets in the TESS data

This Work is Submitted for the Astrophysics Laboratory I class by:

Mariana Carolina Villamil Sastre

Université de Genève
Faculty of Science
Astronomy Department

Advisors
Vincent Bourrier. Ph.D
Nathan Hara. Ph.D

Geneva, Switzerland
2022

Searching for disintegrating planets in the TESS data

Mariana Carolina Villamil Sastre

Université de Genève

Abstract

Catastrophically Disintegrating Exoplanets (CDEs) are Mercury-size rocky bodies that due to the close distance with their host star, these planets can heat up to high temperatures, allowing the sublimation of the planet's surface. Only less than 10 exoplanet candidates has been confirmed as CDE's in the last 10 years. Taking into account that these planets have very short periods (4-30hr), observations could miss/confused them with other big planet transits and stellar phenomena. In this case, we are looking to identify this type of transits using light-curves from TESS telescope data. Through the implementation of a pipeline, we can detect them in an automatically way, we use an exponential model that considers a set of orbital parameters that are fit through an optimization method. This one, was selected from three different routines tested during the work. Finally, we evaluate the entire method in 500 systems from which we obtain a final list of candidates for CDE's.

Contents

Chapter 1

Objectives

1.1 General Objective

To find a list of possible candidates for disintegrating exoplanets for future CHEOPS telescope observations, employing an automatic pipeline capable to identify this kind of planets.

1.2 Specific Objectives

- Implement the optimal modeling method that fits the data processed with the pre-existing pipeline.
- Increase the sensitivity of the grid of variables used on the parameters search space.
- Run the code on at least 500 systems with the help of the Lesta supercomputer.

Chapter 2

Introduction

Exoplanets detection in the last decade has been more frequent and easier thanks to the Kepler telescope (2009). Since that, a catalog of 2.662 confirmed candidates has been created in more than 530.506 observed stars. Similarly, the TESS telescope was a mission developed to observe a bigger sky area than the one observed by the Kepler telescope. Launched in 2018, TESS focus on finding planets around stars of type G, K and M with apparent magnitudes greater than 11. This telescope can observe up to 500.000 system and covers an area 400 more significant than the one studied by Kepler. Their observations aim to find rocky planets, through the transit method, the TESS telescope performs data releases every few months from its launch. This method consist on recording the photometric data obtained from the eclipse event caused by the planet on their host star. From the observer line of sight, a reduction on star's intensity flux is evidenced periodically as shown below:

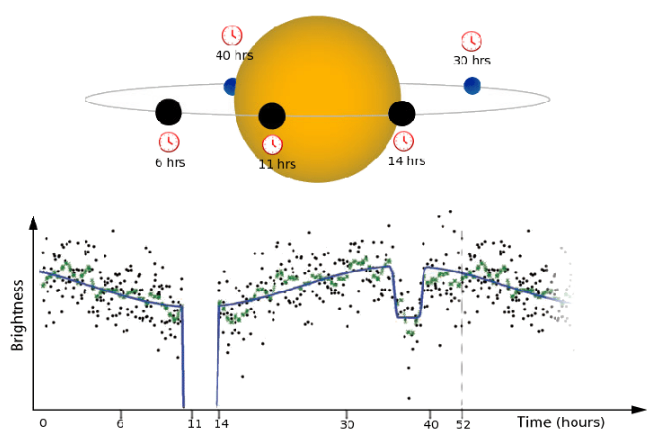


Figure 2.1: Regular light-curve obtained from the transit of system HAT-P-7b observed by Kepler [3] showing primary transit and secondary eclipse measurements (Tinetti et al. 2011)

One interesting fact from this observations is the detections that can be done on Catastrophically Disintegrating Exoplanets (CDEs), which are rocky bodies in shorter orbits and sizes similar to Mercury [7]. Due the short distance to their host star, these planets heat up to more than 2000K temperatures, allowing the sublimation of said planet's surface. This sublimated material is then placed on the trajectory of the orbit, being dust composed of metal-rich vapor[2], which will enable it to be detected by the transit method. In this way, an asymmetric light curve is obtained as the following figure shows:

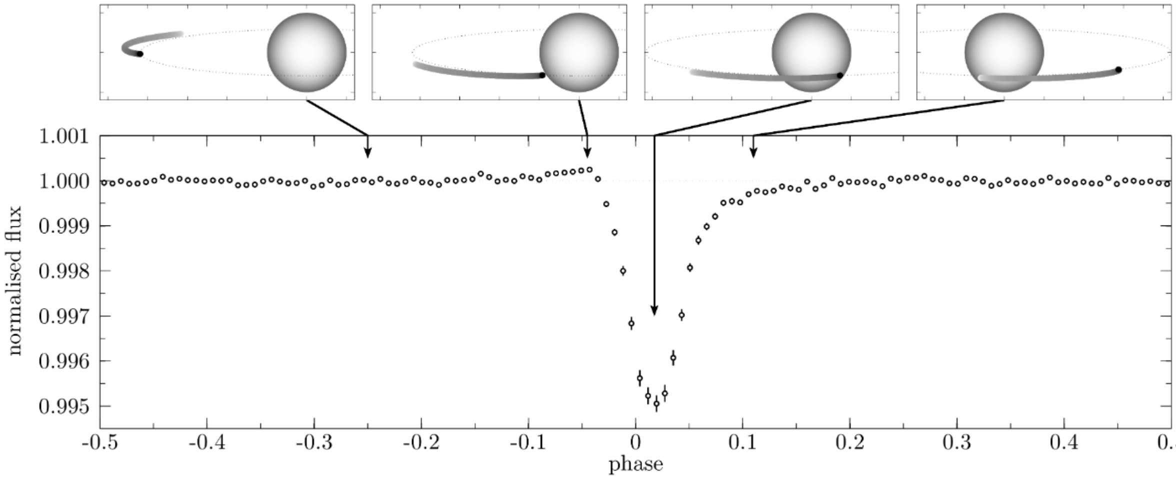


Figure 2.2: Asymmetrical light curve due to the transit of a disintegrating planet. The egress of the transit on the right hand is less sharp due to the signature of the tail of the disintegrating planet (Van Lieshout et al. 2016)

TESS mission include an automated pipeline to identify the transits and classify them as a possible candidate, but this pipeline was designed for identifying symmetrical light-curves. Because of this asymmetrical feature, TESS pipeline could have missed this kind of systems.In this way, there was the necessity to built a method capable of recognize CDE's transits and differentiate them from the regular ones.

2.1 TESS Data

It was important to understand how the data was structured in order to built a new method with different parameters. Following that, the TESS telescope was built with four wide-field cameras sensitive to wavelengths close to red. Such cameras can study an area of the sky that is 24° x 90° (Orion constellation as a scale-Figure 2). Since its operation, it has monitored both hemispheres of the sky, searching for exoplanet transits. Each observation has a duration of 27 days approximately [8].

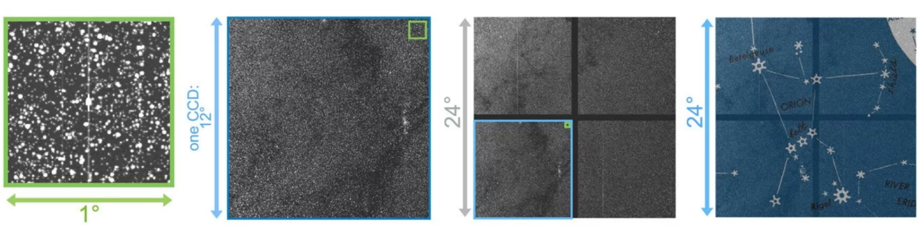


Figure 2.3: Wide field of view of one TESS camera (TESS mission description at [8])

Each data release is made up of:

- Raw full frame images
- Calibrated full frame images

- Target pixels
- Light curves
- Pixel response function
- Lists of TESS Objects of Interest (TOIs)

The calibrated full-frame imagehave a 2-minute cadence, allowing us to discern the entry and exit of transit for the brightest candidates. Also, each pixel that TESS observes will have a cadence of 30 minutes.

2.2 Previous work

Previously, this project had started implementing a disintegrating exoplanet identification pipeline (Celvin Tan's APL 2021 work). This code was structured so that it could read and pre-process the light curves obtained from the official mission page. This processing was done with the help of a package called MonoTools, which allows normalizing and detrending curves efficiently. Once the data was pre-processed, the pipeline was permitted to create an exponential model in order to emulate the shape of the light-curve from this kind of planets. Taking in to account that the dust is already present on the orbit and line of sight in the forms of tails, the transit is going to take more time on decay than a regular transit because the tails are making part of the eclipse itself. In that way, the model includes a constant for normalization, the amplitude and the modulus operator of the period as shown below:

$$flux = C - Ae^{\frac{(t\%P)-t_0}{\tau_{left}}} \qquad 0 < \forall(t\%P) < t_0$$

$$flux = C - Ae^{\frac{t_0-(t\%P)}{\tau_{right}}} \qquad t_0 \leq \forall(t\%P) \leq P$$

Parameters	**Description**
Period (P)	The time taken for the disintegrating exoplanet to complete 1 revolution around the host star.
Time of transit minimum (t_0)	The time at which the minima of the transit occurs.
Extinction factor of leading tail (τ_l, τ_r)	The time is taken for the light curve on the left/right half to decay by a factor of e^{-1}. As the dust extinction lifetime is constrained to be less than 1 period,each half of the extinction factor are constrained to the values between 0 and 0.5
Amplitude(A)	Amplitude value of the light curve for linear fitting to datasets
Constant (C)	Constant value of the light curve for linear fitting to datasets

During this process, a grid of parameters was created, with each possible value on the periodic space. For each set of variables, the χ^2 is calculated in such a way that the pipeline is allowed to make a mapping of all the possible combinations until it finds the one that best fits the data. This grid has a dimension of (8x25000)

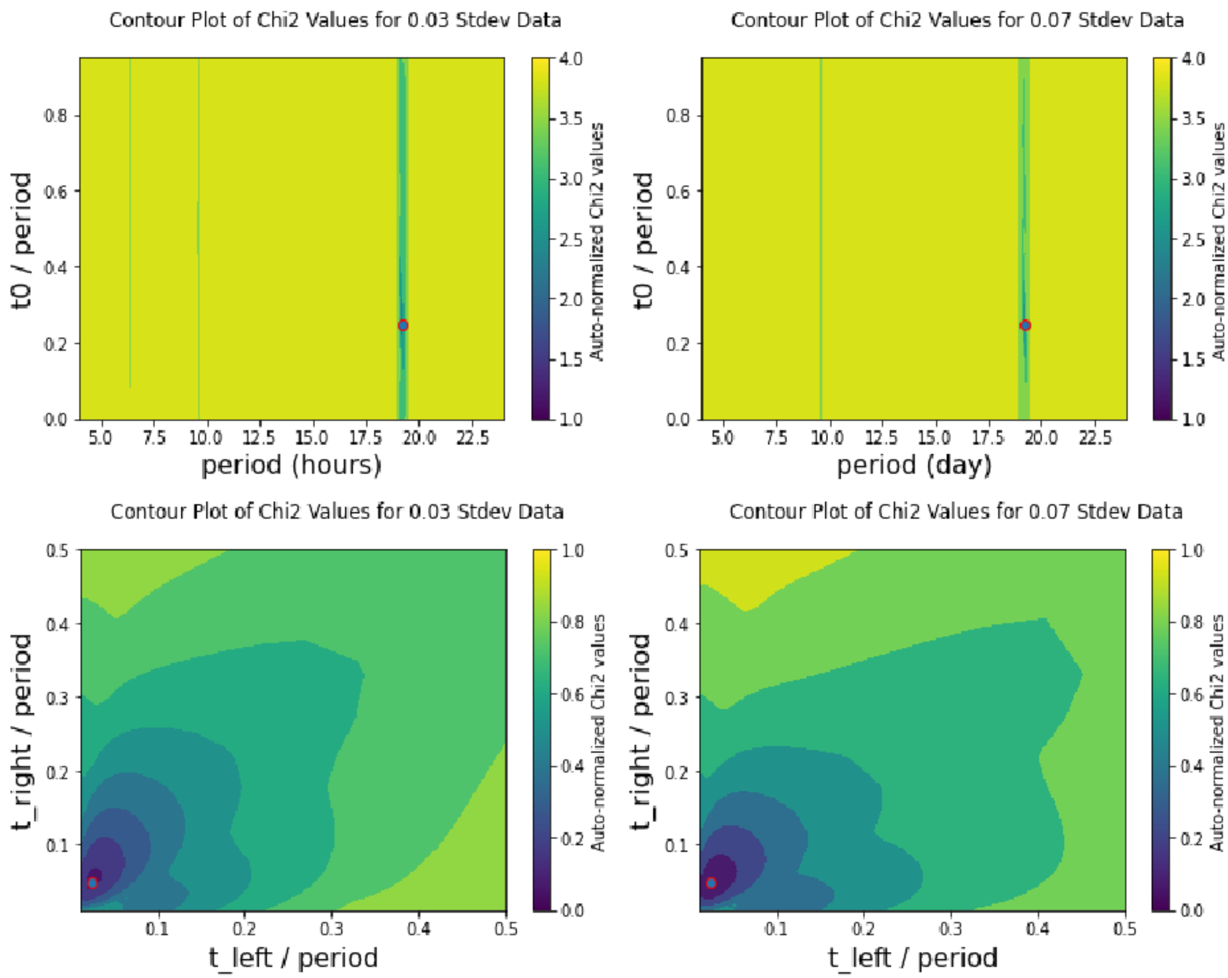


Figure 2.4: Contour map of the different variables values outputted by the grid system [9]

2.3 Parameters grid problem

Now, that the structure of the first pipeline version has been reviewed, it is necessary to understand its limitations:

- The old routine was designed to run through each possible period, in order to calculate the set of parameters that depends on this quantity (t_0 , τ_l , τ_r), and then calculate the variance excess values in order to choose the lowest value which means the one that better fits the data. However, it gives as a result a thick parameter grid. In other terms, the pipeline couldn't compute enough combinations of values between the parameters that lead us explore the entire periodic space that we are interested in.

- This process was still too computationally time-consuming, it could take up to 10 hours for each system to calculate the entire contour map, which implies that trying to explore shorter periods or at least more systems could take large quantities of time.

- Due to time constrains, the code could only be tested in 125 targets, of which only 17 met the specific conditions: Systems with only one planet around, magnitude less than 10MAG, headers of light-curves well disposed and good quality data.

- As we are looking for exoplanets with the size of Mercury, these periods are expected to be very short. In this way, the grid considered at the old version of this work [9], had an spacing between each set of parameters too big to being unable of detecting the values associated with a Catastrophic Disintegrating Exoplanet (CDEs).

In this way, it was necessary to reduce the computational time to test the pipeline in more systems and carry out a statistical analysis that would allow improving the classification of these systems more effectively.

Chapter 3

Methodology

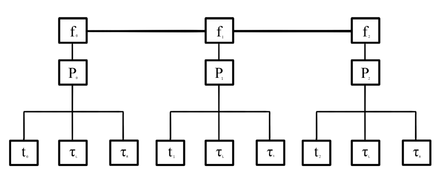
To solve the grid parameters problem, it was necessary to test different methods to calculate the best set of parameters that fit the data quickly and efficiently. All the procedures shown below were adapted to the pipeline so that in all the three cases, the pre-processing was the same (MonoTools). Still, the method of finding the set of values is different for each case, except for the normalization constants, which were found using the Least Squares method in all cases.

3.1 Hierarchical grid of parameters

For this method the old grid was adapted with a series of changes:

- Equal spaced in frequency intsead of spaced period.
- Hierarchical order, which means that when the parameters of the grid are all independent, the number of elements in the grid grows fast. Here, the period of the planet puts constraints on the rates

Implying that the period is structured in terms of frequencies, which varied between $\frac{1}{4}hr$ and $\frac{1}{30}hr$, setting a spacing small enough to allow the exploration of all possible frequencies. Thus, studying the periods that were not sensitive to the old parameter grid. Likewise, it was established that the extinction factors and the minima transit time depended on the period, considering that the first ones are described as the time it takes for the light curve to decay exponentially on each side of the characteristic transit. In this way, a more refined grid was created with a size of (8x2160000), which means the implemented pipeline iterated 250,000 times and in each iteration performed the calculation of all the parameters to compare and find the best set with the smallest excess variance value (χ^2). Below is a diagram of what the grid created with this method would look like:



1 2 ...

3.2 Automatic hyperparameter optimization package Optuna

Several packages implemented in Python were searched to reduce the computational time significantly, which would allow an automatic search of a parameter grid with robust statistical methods. It is then when OPTUNA appears, a package implemented relatively recently by (Akiba et al. 2019), which allows creating a space of parameters to find the best set autonomously without the need to calculate all the values corresponding to each set of parameters that make up a grid, as in the previous case. Optuna allows the search for hyperparameters through an optimization process (minimize and maximize), an objective function that takes a set of hyperparameters as an input. In this case, it is the exponential model and returns the validation of these values along with a score. This function constructs the parameter space dynamically without relying on externally defined static variables [1]. Parameters such as the period, the extinction factors, and the minimum transit time are defined as a "trial object." From this, Optuna builds the objective function through these trial objects, constructing the parameter space. With a random walk, Optuna finds the trial with the best score [4].

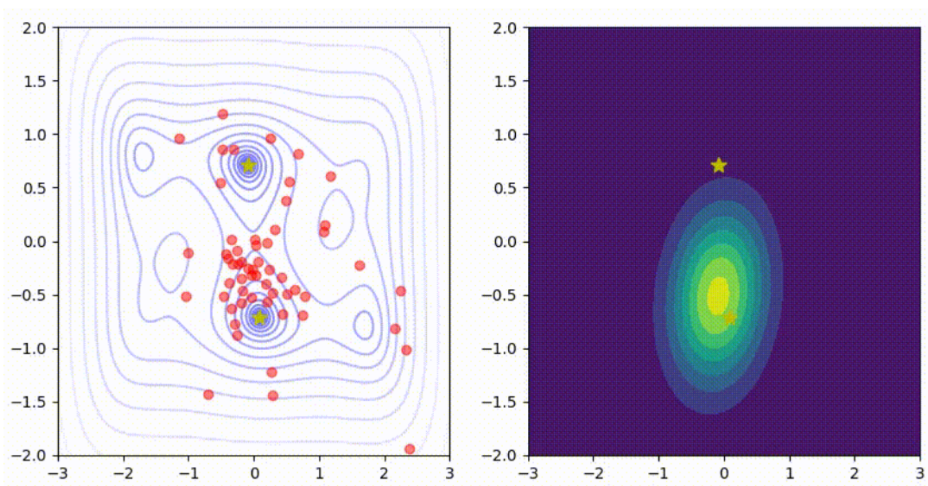


Figure 3.1: Automated search for optimal hyperparameters using Python conditionals (Shibata, 2021)

From this method, it is possible to highlight that the computational time required for each system is only a couple of hours, and parallelization in a supercomputer is relatively easy. There is already a pre-implemented code to divide the work of the search of the hyperparameters in several nodes of the same server, which is a great advantage since it is not necessary to modify the structure of the pipeline with which we have been working. Also, the package allows reproducing contour plots automatically with tools included in it, which reduces programming time significantly. Finally, Optuna is free access software.

3.3 Box Least Square Method (BLS)

Finally, the last method was tested to find the value of the optimal period and from this to see the other parameters that best fit the model following that initial value. The BLS method models the light curve as a box function with defined depth and models the light flux outside the transit with an average normalization value [7]. This method fixes the transit depth and reduces the computational time required. This method is more sensitive to detecting exoplanets primarily associated with gas giants since the depth of the transit is greater than the noise levels introduced by the instrumentation, stellar activity, and the same method of modeling the box [5]. So in this method, the options are limited to a binary response. That is, the data is either in transit or outside of it.

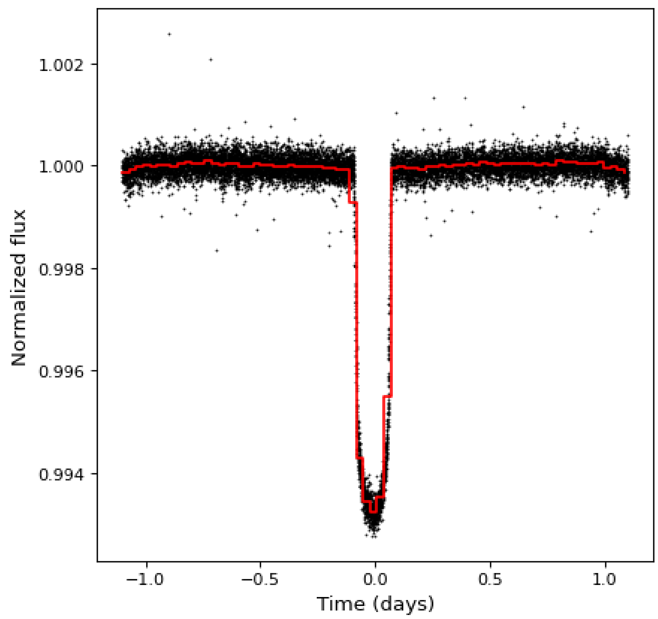


Figure 3.2: BLS modeling for a regular transit (astropy.timeseries documentation)

BLS allows us to calculate the periodogram of the transit through a grid of periods uniform in frequencies and find the period with the maximum value of power. The method does not consider the exponential component of the transit corresponding to the entry and exit of itself. It works more like a step function [7].

Each period found in the periodogram performs the phase-folding of the data, which is a permutation of the original time series, ie. there is a phase described with the modulus operator as in the model mentioned in the previous chapter, that lead us studied the entire time series on the estimated period-interval. Therefore, all the points are binned, allowing the algorithm to take into account various ranges of these bins and analyze the best one through the Least Squares method to determine the power of the periodogram. In this way, the technique makes a fit of the data with the box functions.

Chapter 4

Implementation

4.1 LESTA Super computer

For all the methods mentioned in the previous chapter, it was necessary to configure the pipeline on an available server. When performing each procedure's computation separately, various problems were found at the hardware level.Therefore, it was necessary to send all the processes to the LESTA supercomputer, managed by the Department of Astronomy of the University of Geneva. This computer was chosen and not the general Yggdrasil cluster due to the waiting times in each queue. Although the latter has a higher storage capacity, being a cluster of the entire university community, the times in line were longer. While the times were shorter for the LESTA cluster, the calculations were initially run in a single core. So the start of execution of the tasks was almost immediate.

A 20Gb memory was assigned to all jobs and a maximum of 60 hours. It is strongly recommended to read the LESTA-documentation, to configure all the necessary packages to run both the pre-processing of the data and the different methods of searching for parameters. In this case, all the calculations were carried out from the **scratch** desktop, and more than 38 jobs were sent that allowed correcting the errors that were presented until obtaining the most optimal and efficient results.

It is necessary to clarify that: installing the packages on the MonoTools server presented problems with the Python versions available in LESTA (Python 3.7), as dependencies require newer or older versions. To solve this problem, it was necessary to update the versions of TensorFlow that require MonoTools dependencies (See Eleanor documentation on Github). The only methodology for which jobs were sent to different cores was the Optuna algorithm. As mentioned above, the documentation available for this package includes a simple code to search for hyperparameters in other cores without modifying the pre-existing pipeline.

4.2 Comparison between methods

Initially, the three methods were tested for the **ticid2234723** system with all available data points. Subsequently, they were binned every 20 minutes to simplify the visualizations and the results of the model that best adjusted to that system. In all cases, the optimal period was found with which the phase-folding of the data was carried out. Obtaining the following results:

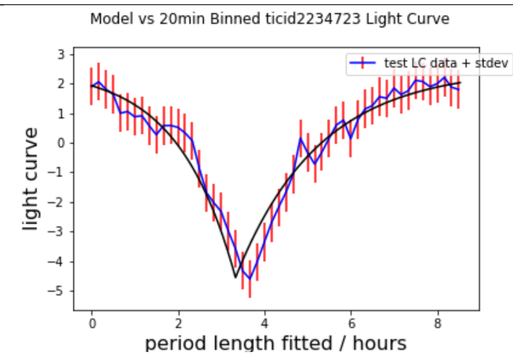
<table>
<tr><th>Method</th><th>Final Set of parameters</th></tr>
<tr><td>


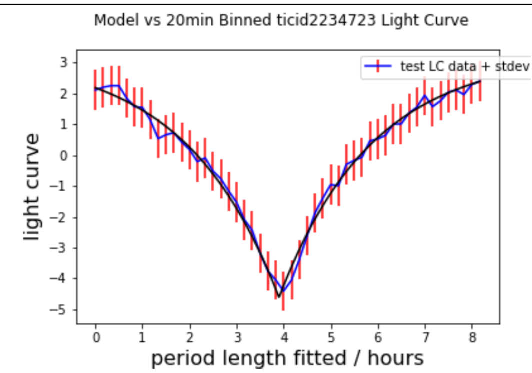
Table 4.1: Hierarchical grid of parameters
</td><td>
P=8.54 $\pm 4.9hr$
t_0=200 $\pm 92.17min$
τ_l=79.74 $\pm 25.6min$
τ_r=115.2 $\pm 38.7min$
C=2.49$\pm$1.84
A=7.04$\pm$2.56
Redχ^2=10.9 $\pm$2.016
</td></tr>
<tr><td>


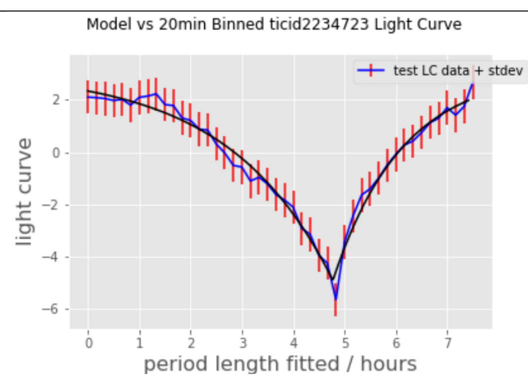
Table 4.2: Optimization algorithm
</td><td>
P=8.32 $\pm 3.14hr$
t_0=234 $\pm 89.8min$
τ_l=119.34 $\pm 72.2min$
τ_r=138.26 $\pm 68.16min$
C=3.28 $\pm$1.2
A=7.90 $\pm$1.9
Redχ^2=9.1 $\pm$6.65
</td></tr>
<tr><td>


Table 4.3: Box Least Square
</td><td>
P=7.43 $\pm 0.68hr$
t_0=286 $\pm 50min$
τ_l=121.55 $\pm 40.42min$
τ_r=80 $\pm 1.20min$
C=3.10 $\pm$2.2
A=8.02 $\pm$3.8
Redχ^2=9.26 $\pm$5.45
</td></tr>
</table>

Also, the differences in computational capacity were substantial as shown below:

Method	**Consuming computational time (hr)**
Hierarchical grid	7.2
Optuna	3.5
BLS	6.1

In this way, more features were considered to help discern which was the best method to use in a dataset of 500 systems.

While the hierarchical parameter grid could scan the entire frequency space, as shown above, it took a long time running on the cluster to made all de calculations in a single system. As it had to do more than 200,000 iterations, it was necessary to find a faster method. On the other hand, although Optuna was the fastest, the calculations made by this method(includind the χ^2)could correspond to a local minimum, when ideally, we are looking for the global minimum, because it's the set of parameters that are more like the data. However, when testing the three methods, it was found that all of them converged to the same range of optimal periods (8hr aprox), which allowed us to confirm that Optuna was indeed finding the global minimum as expected.

Finally, the Box-Least Square method resulted to be not too sensitive to short periods as those of planets with orbits and sizes similar to Mercury. If the Signal/Noise ratio is not large enough, they can be confused with the noise involved at an instrumental and mathematical level. In addition, this method models the box-shaped transits, which would serve as a regular transit. Still, the objective of this work is to find disintegrating planets, which describe an asymmetric light curve that does NOT follow the shape of a box in particular. Also, the computation time that was very similar to the grid did not make it so advantageous concerning the other methods.

In this way, the new grid and Box-Least Square methods were not taken into account to analyze the more than 500 systems. Instead, Optuna was chosen to be the method to implement within the pre-existing pipeline.

Chapter 5

Results

5.1 Classification

After choosing Optuna as the primary method, the entire pipeline was run on 500 systems. Of which 95 met the necessary conditions to be processed by the code. However, 30 of them had a header with bad formatting, and 15 had low-quality data. Therefore, it was possible to classify 50 systems within the initially accepted parameters.

It is necessary to remark that in this version of the work, the condition of only studying systems that only have one exoplanet and whose apparent magnitude was less than or equal to 10mag was maintained (TESS is sensitive to these magnitudes). Also, the stellar variability of the host star was not taken into account.

However, it was necessary to discern this classification list in more detail, as systematic errors could give a 'false positive' within the list. Therefore, for all 50 pre-candidates, the following parameters were extracted:

- Amplitude: A
- Amplitude error: σ_A
- $\frac{A}{\sigma_A}$
- χ^2
- τ_l , τ_r

A threshold value was set from these values to distinguish between the strongest candidates and a normal exoplanet. This being the case, it was expected that all the systems that fulfilled the condition of $A > 3\sigma_A$ with a relatively low χ^2 could be considered in a pre-definitive list as candidates for disintegrating planets. However, it was decided to take into account a final classification filter. Systems that meet the first condition and that in turn meet high values of τ_r and τ_l could be considered in a final list of candidates.

Following these criteria, 8 systems were found that met the first condition as shown below:

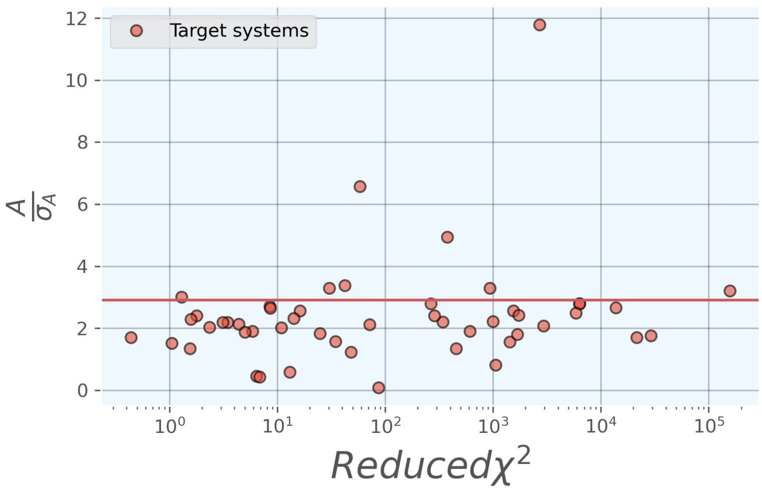


Figure 5.1: Target systems that met the first condition

To better visualize it, those eight systems were plotted on a 3D graph, where the extinction factors and the reduced χ^2 were shown to differentiate the systems that met the second condition. Recall that the reduced χ^2 is defined as the χ^2 divided by the number of data points corresponding to the light curve of that system. The 3 points that are observed in the right part of the graph are the ones that most interest us for the classification, as shown below:

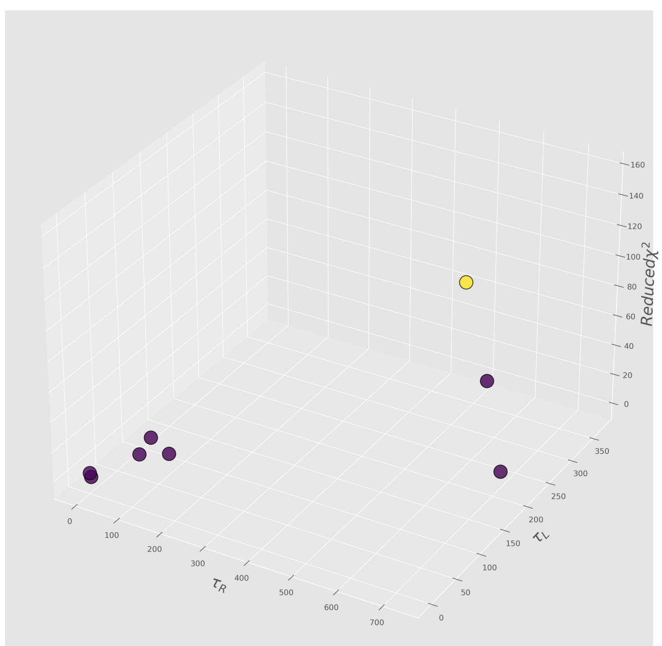


Figure 5.2: 3D plot of the targets systems that met both conditions

5.2 Remarkable systems and Statistical analysis

Following the conditions mentioned above. 3 systems were selected that met all the filters set for this classification. The systems then are:

- 367366318
- 467722869
- 116260314

From these systems, the parameters found by Optuna performed 5000 iterations in a random walk in search of the best set of parameters. Taking into account that the 'Objective value' refers to the model values obtained during the Bayesian examination, it is possible to show that the regions of the contour map with the smallest values correspond to the values with the smallest χ^2 that is associated with the final results of the fit calculated by Optuna. It is necessary to emphasize that the scale for each parameter was defined in the same way as in the parameter grid discussed above. That is, τ_r and τ_l are defined depending on the period and in a logarithmic scale to follow the exponential form of the transit model being studied. Is necessary to clarify that these parameters were defined in the same way, but the 'trials' were different on every iteration, that's the reason they look very similar on their correlations as shown below:

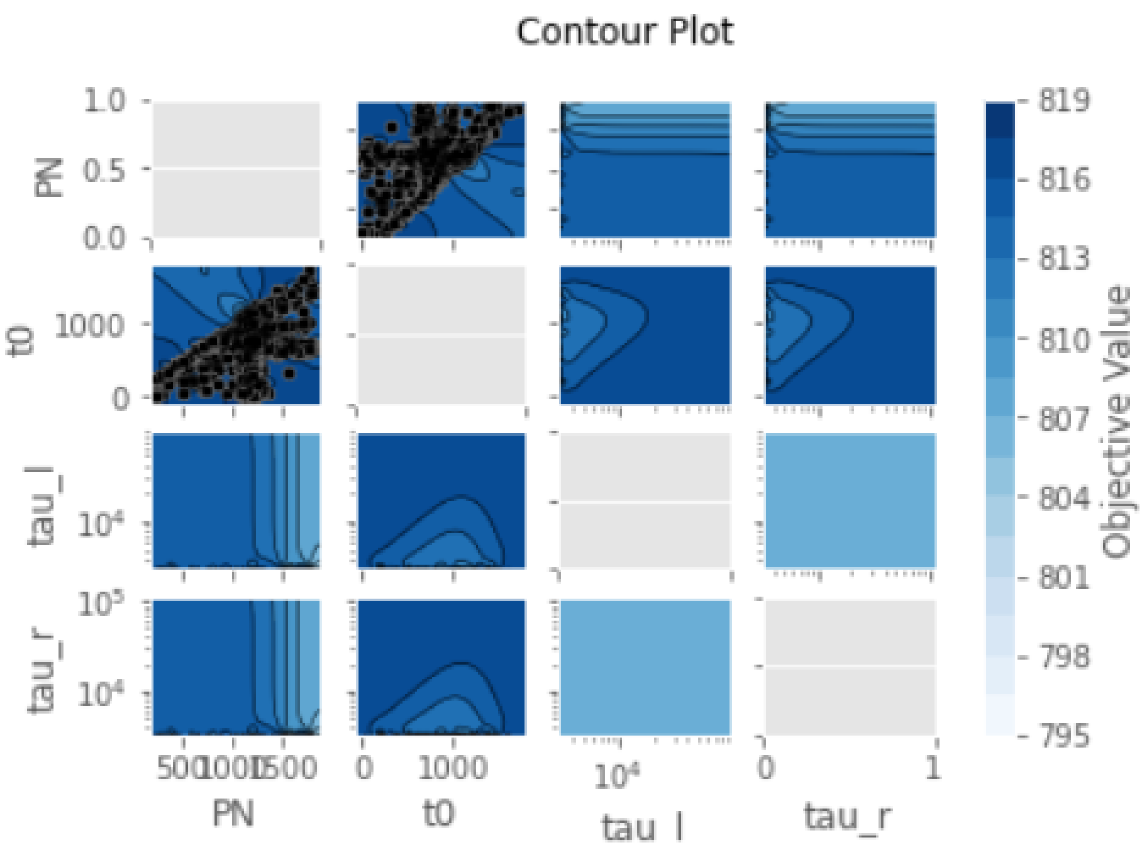


Figure 5.3: Contour map for the hyperparameter search system 367366318

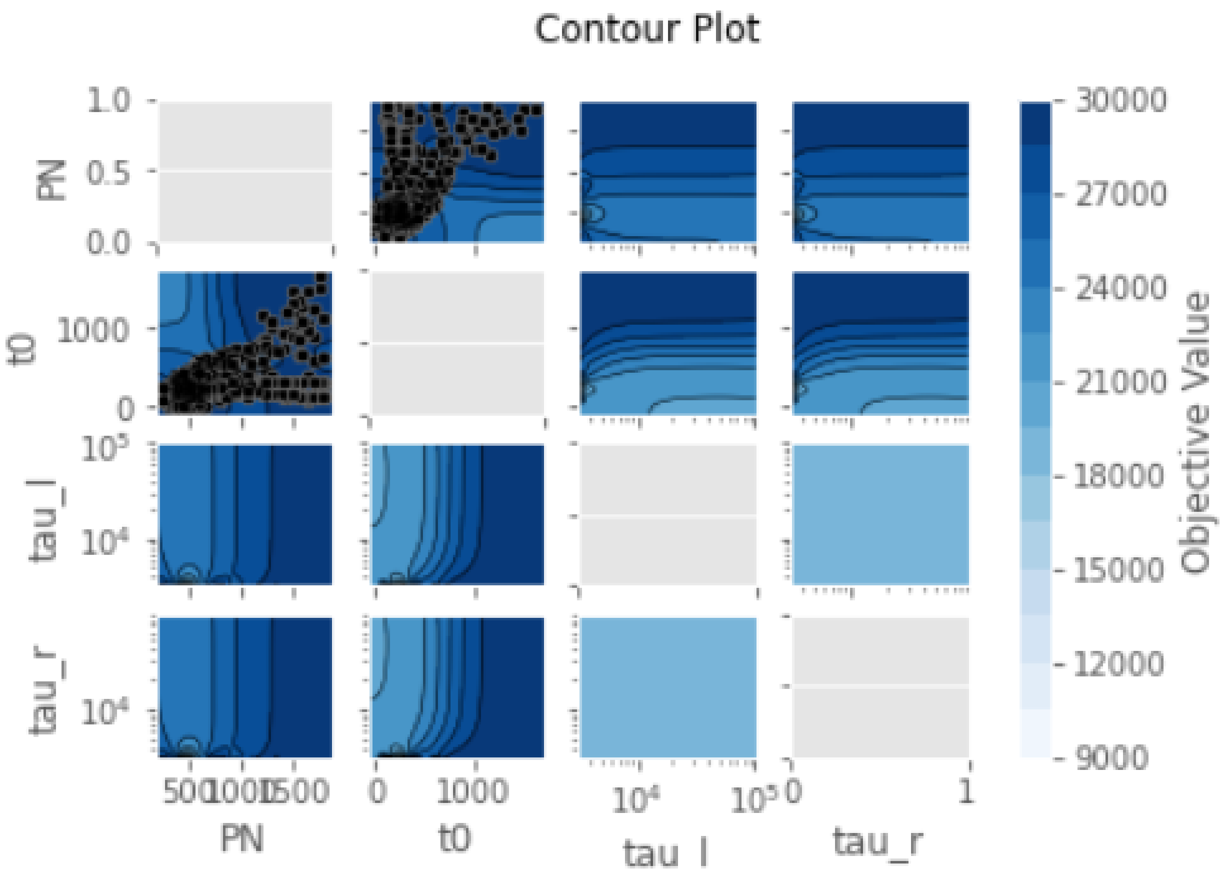


Figure 5.4: Contour map for the hyperparameter search system 467722869

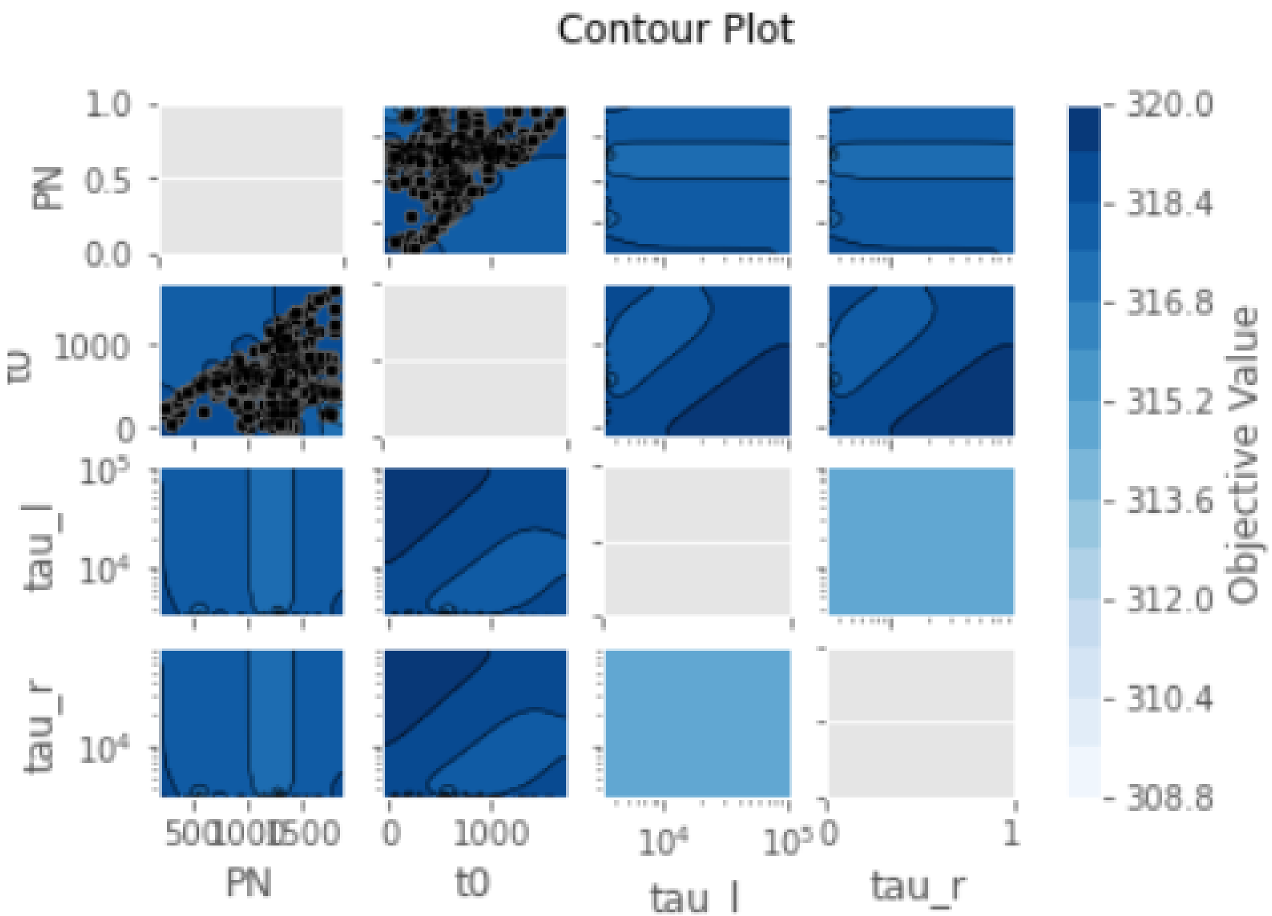
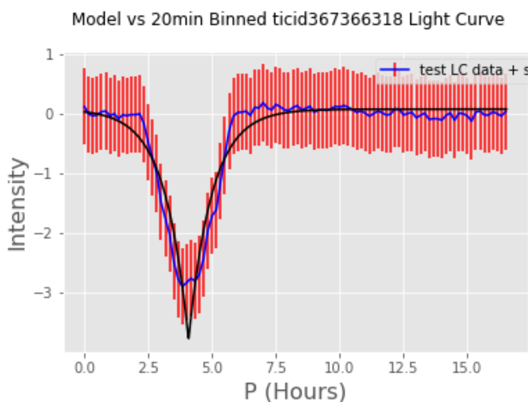
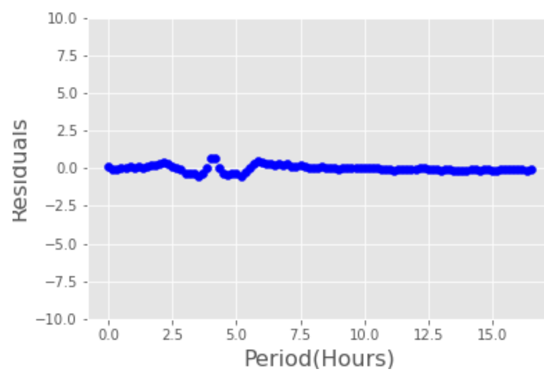


Figure 5.5: Contour map for the hyperparameter search system 116260314

A statistical analysis was performed for each system to verify that the model fits the data observed by the TESS telescope. Initially, for the first system, through the computation of the residuals, it is possible to follow that the model made a correct approximation of the extinction factors since the difference between these curves is close to zero, as seen in Figure (b). However, the depth of the eclipse is greater in the model than in the data by a difference of 0.6 intensity units. The rest of the curve has an average of zero, as expected. Likewise, an analysis is made in the uniform Fourier space in frequencies (Figure C). There are maximum peaks that describe the amplitude studied in the residuals, only the real part of this transform is taken to simplify the analysis. Finally, Figure (d) shows the relative frequency of the residuals for each value using a histogram, which follows a distribution centered at zero and whose limits range from -0.6 to 0.6, indicating that the model was not exact but approximate.

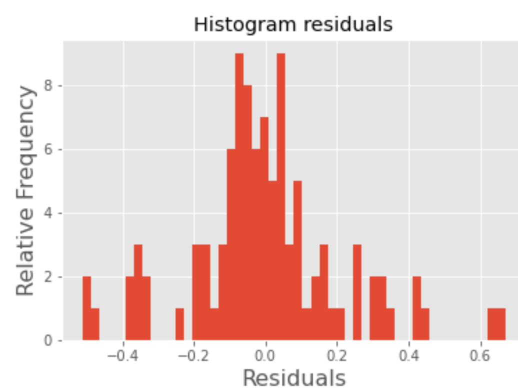


(a) Best fit Vs data binned every 20 minutes



(b) Residuals (Model-Data)

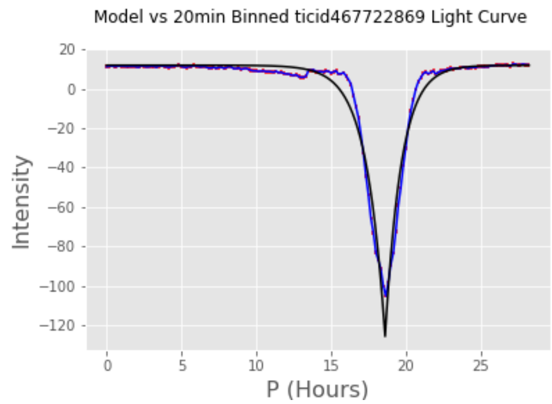
(c) Residuals Fourier Transform



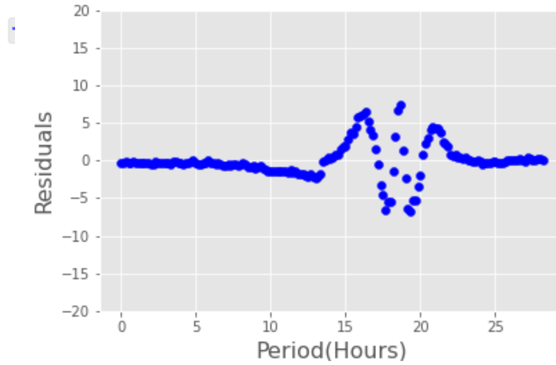
(d) Residuals histogram

For the second system, the same analysis was developed. As in the previous case, the modeled depth was also greater, and is evidenced in the residuals where three maximum peaks are obtained. It is important to note, that the histogram shows a distribution whose highest relative frequency is close to zero. It is necessary to clarify that we binned both the data, the data error, and the model every 20 minutes to analyze the residuals and to facilitate their visualization.

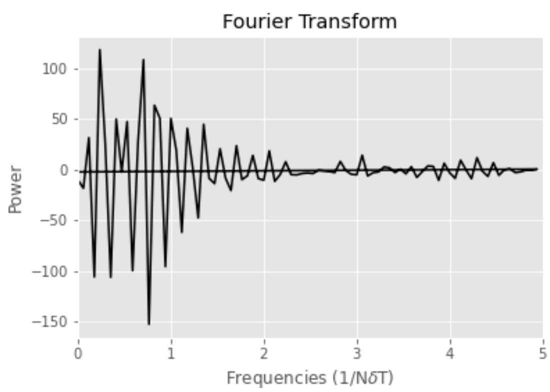
In this case, although the system is in the **Candidate Target List (CTL)**, it has not been entered as a **TESS Objects of Interest (TOI)** or a **Community TESS Objects of Interest (CTOI)**, that is, you have not reported yourself as a confirmed candidate. This database is managed by the exofop-TESS community, where collaborators from different institutions introduce stellar and planetary parameters that they find on a particular system.



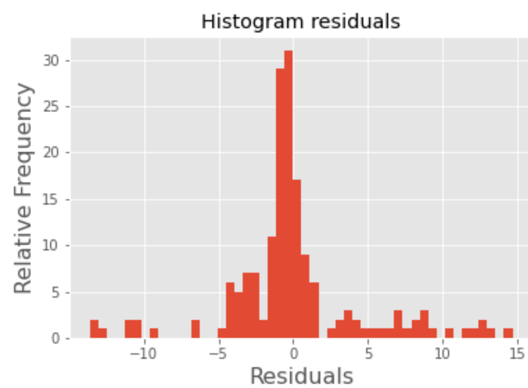
(a) Best fit Vs data binned every 20 minutes



(b) Residuals (Model-Data)



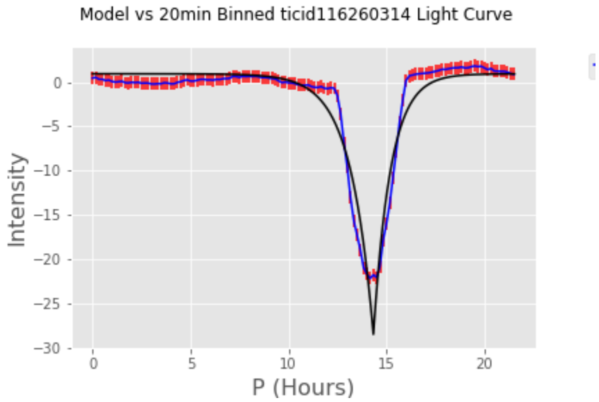
(c) Residuals Fourier Transform



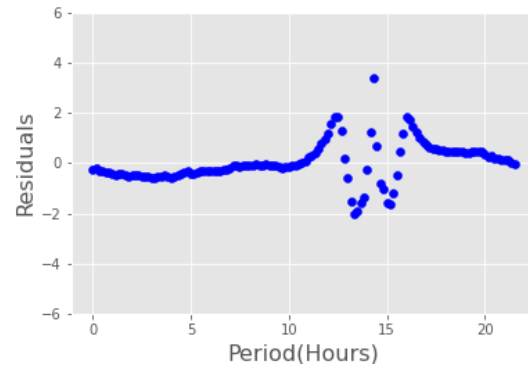
(d) Residuals histogram

Finally, the third system shows a more asymmetric light curve than the first two. Although this analysis is purely quantitative, it is possible to show that the model again calculates an excessive value of amplitude reflected in the residuals and even more so in the maximum peaks shown by the Fourier transform. In this case, the histogram distribution is found with a higher relative frequency density in the range of [-1.0].

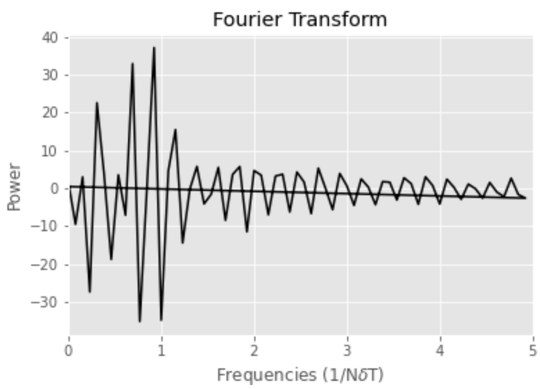
For this system, as in the previous case, it is in the **CTL's** list, but there is no record of planetary parameters or in the **TOI's** or **CTOI's** list.



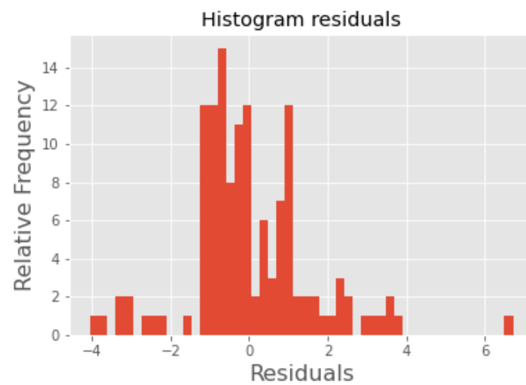
(a) Best fit Vs data binned every 20 minutes



(b) Residuals (Model-Data)



(c) Residuals Fourier Transform



(d) Residuals histogram

From the analysis of these three systems we can find that the model certainly is not a perfect to model the data from observations. Altough, is pretty approximate and can identify the transits differing them from the poor quality data, also it places the t_0 as is expected at the same time with the impact factors. Which means that this parameter shouldn't be changed on the way it's defined for future work.

Chapter 6

Conclusions

In conclusion, we optimized the pipeline to identify disintegrating exoplanets by implementing three different hyperparameter search methods. After testing each technique, the one that reduced the computation time by a factor of almost 40% was chosen, and the set of parameters that best fit the observed data was found. We discarded the other two methods for two reasons: exaggerated computational times and because they were modeling transits of regular and giant exoplanets. Once the optimization algorithm was implemented in the pipeline, we tested it on 500 systems included the ones from the last data release (11-11-2021). From these 500 systems, 95 met with the first filter of conditions that we imposed while reading the data (Systems with only one planet, whose magnitude was less or equal than 10 magnitudes, this because we are looking for targets for CHEOPS future observations, and this telescope is more sensitive to this magnitudes). From these 95 systems, 30 had a wrong format in the header that did not allow the pre-processing of the light curves, and 15 systems had poor data quality, ie. They had a high dispersion with less than 100 data points when the usual was that the light curves had more than 18.000 data points.

In this way, the pre-processing of 50 light curves was made. However, it was necessary to discern which pre-candidates could be considered as disintegrating exoplanets. So, we identified two classification criteria. Initially, the targets that met the condition $A > 3\sigma_A$ and had a relatively low χ^2 concerning the other targets could proceed to a second classification phase. Which consisted of studying systems with high values of τ_l and τ_r. Following these specifications, three systems were found that passed all the mentioned filters. Being the systems with TICID: **367366318, 467722869, 116260314**.

We performed statistical analysis on the residuals of each model generated in each system by the Optimization algorithm. We evidenced that in the all the three cases, the algorithm identified the transit, found the correct period that allowed the phase-folding of each curve, and calculated the other parameters that fit the data. However, it is observed that the depth from the transit in all the systems was greater than the data. Which implies that the model is not describing perfectly the data and for a future work is highly recommended to include new parameters during the modeling. Also, the algorithm (Optuna) needs to iterate more times in the random walk. For this case, 5000 iterations were set, and systems with more than one planet around were not considered, nor was stellar variability considered.

In the three systems, there is no record that they are confirmed as exoplanets in the **exofop-Caltech** database. Again, for future work it is recommended to study further into these systems, including all the variables mentioned above, to verify if these are new exoplanets found through this pipeline. Also, a continuation of this work may consider using the method in the more than 200,000 systems that the TESS database has. The optimization algorithm has parallelization facilities when using a cluster as a tool, making it possible to search for hyperparameters faster and more efficiently.

Bibliography

[1] Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M. (2019, July). Optuna: A next-generation hyperparameter optimization framework. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery data mining (pp. 2623-2631).

[2] Bodman, E. H., Wright, J. T., Desch, S. J., Lisse, C. M. (2018). Inferring the composition of disintegrating planet interiors from dust tails with future James Webb Space Telescope observations. The Astronomical Journal, 156(4), 173.

[3] Heller, R., Hippke, M., Rodenbeck, K. (2019). Transit least-squares survey-II. Discovery and validation of 17 new sub-to super-Earth-sized planets in multi-planet systems from K2. Astronomy Astrophysics, 627, A66.

[4] Hippke, M., Heller, R. (2019). Optimized transit detection algorithm to search for periodic transits of small planets. Astronomy Astrophysics, 623, A39.

[5] Jones, M. H., Haswell, C. A., Barnes, J. R., Staab, D., Heller, R. (2020). A possible transit of a disintegrating exoplanet in the nearby multiplanet system DMPP-1. The Astrophysical Journal Letters, 895(1), L17.

[6] Kovács, G., Zucker, S., Mazeh, T. (2002). A box-fitting algorithm in the search for periodic transits. Astronomy Astrophysics, 391(1), 369-377.

[7] Van Lieshout, R., Rappaport, S. (2017). Disintegrating rocky exoplanets. arXiv preprint arXiv:1708.00633.

[8] TESS Observations. (s. f.). TESS Mission. Recovered on January 8th,2022, from https://tess.mit.edu/science/observations/

[9] Tan, C. (2021, January). Search for Disintegrating Rocky Planets. Department of Astronomy, University of Geneva.

[10] Tinetti, G., Encrenaz, T., Coustenis, A. (2013). Spectroscopy of planetary atmospheres in our Galaxy. The Astronomy and Astrophysics Review, 21(1), 1-65.