

DEVELOPING ASTROPHYSICAL DATA MANAGEMENT & PROCESSING TOOLS

Burcu Yılmaz

Faculty of Science, Physics

burcu.yilmaz@ug.bilkent.edu.tr

Emre Vardar

Faculty of Engineering and Natural Sciences, Computer Sciences

emrevardar@sabanciuniv.edu

Efe Öztaban

Faculty of Management, Undeclared

efeoztaban@sabanciuniv.edu

Ersin GÖĞÜŞ

Faculty of Engineering and Natural Sciences

Yuki KANEKO GÖĞÜŞ

Faculty of Engineering and Natural Sciences

Abstract

The main aim of this project is understanding how to use astrophysical data and how to find out the patterns in the data. Also learning Linux commands, coding in Python and running codes in the university's server is required for this project. TESS data in fits format will be plotted by Python code which will be executed in the university's server called Cosmos. Then this data will be transformed from flux vs. time into power spectrum vs. frequency by using FFT. With the graph from new data, periodic behaviors in the data can be seen easily.

Keywords: TESS, Exoplanets, Transit Method, Fast Fourier Transform, Astrophysical Data Analysis.

1.Introduction

Our purpose in this project was realizing the periodical behaviors in the astrophysical data which comes from TESS mission. The Transit Exoplanet Survey Satellite (TESS) is a space telescope which is operating in NASA's Astrophysics Explorer Mission by MIT, Cambridge, Massachusetts. Before TESS mission, Kepler Space Telescope was operated by NASA for exoplanet discovery. [4] Kepler mission started in 2009 and after nine years of operation, it was retired in 2018. TESS is designed to replace Kepler space telescope and search for exoplanets in 400 times larger area than that covered in Kepler's mission. With these purposes, TESS was manufactured and expected to complete the 2-year primary mission. [1] TESS was launched on April 18, 2018 via SpaceX's Falcon 9 rocket and TESS started to operate on 7 August, 2018.

TESS will mainly focus on the nearest and brightest main sequence stars to detect exoplanets by using the transit method in its two-year primary mission. In its two-year mission, the way TESS can observe the sky is diving sky into 26 different sectors. 13 of these sectors are in Southern hemisphere and the other 13 sectors are in the Northern hemisphere. Each sector is size of 96 degrees by 24 degrees. [1] TESS spends one year on each hemisphere and started its observation on Southern hemisphere. In addition, each sector is observed by 27.4 days.

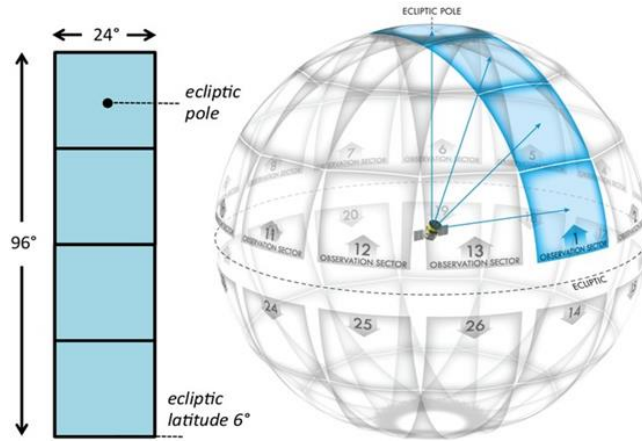


Figure 1. 26 sectors which will be observed by TESS.

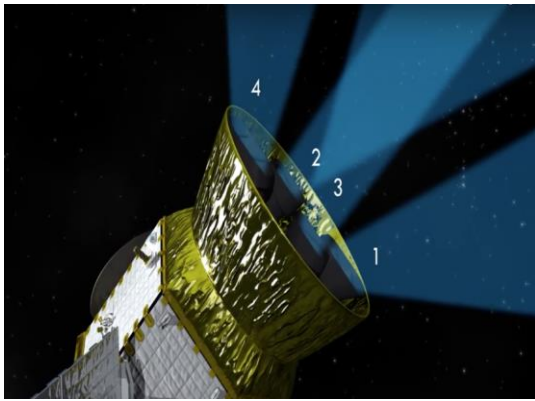


Figure 2 TESS's four wide field optical CCD cameras.

TESS has 4 wide field optical CCD cameras for these observations. CCD stands for Charged Coupled Device and it is a circuit which forms light sensitive elements called pixels. The circuit is attached on a silicon surface and the working principle of this device is using the photoelectric effect. When photons strike the silicon surface, this creates the emission of free electrons. [6] CCD cameras work better under low temperature so that thermal energy can help exciting the electrons. CCD cameras are able to detect periodic drops in brightness, change in the flux, which is essential to detect exoplanets.

Transit method is one of the methods to detect exoplanets in the space. This method is also used in TESS mission. The word "transit" refers to cases where the nearer object moves in front of the more distant object. This method based on detecting these transit motions by looking at flux from distant object. Simply, when a planet passes in front of the star (based on observation location) it creates dips in the detected flux. Because, the planet blocks some of the light coming from that

star during transit move. [6] Also, transit method lets scientists to find more information about the size of the planet and also the size of the star. It is because the flux that came from star directly related to the power and mass of the star. And the magnitude of the dips related to the planet size since exoplanet will block flux in proportion to its radius. The transit method is one of the most efficient methods in detecting exoplanet. As it is represented in the chart, purple colors are show the detected planets by the transit method (fig.3). Furthermore, transits can also provide scientists with a great deal of information about the planet that is not otherwise measurable. With transit method the period of the exoplanets can be detected too. But because of the principles of the TESS mission, it is only able to period of shorter than 27.4 days.

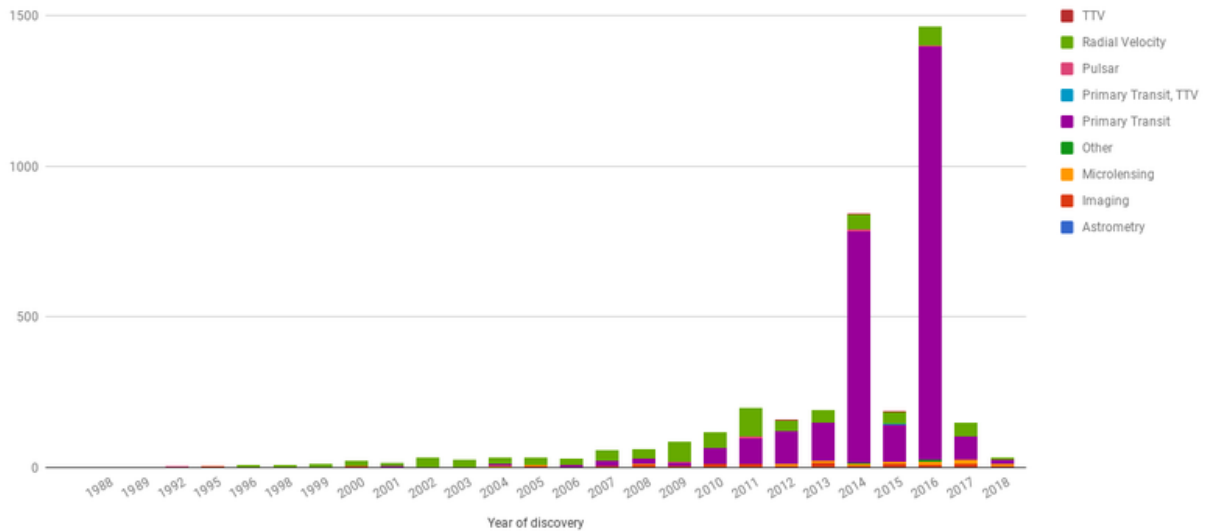


Figure 3. The discovery methods which is used for new exoplanet detection.[3]

The astrophysical data which comes from TESS mission in fits format. Flexible Image Transport System (.fits) is a data storage type which is mostly used for astrophysical data. It is way more developed compared to .jpeg but in some sense, they have similar jobs. This format is endorsed by NASA and the International Astronomical Union. FITS support is available in a variety of programming languages that are used for scientific work, including C, C++, C#, Java, MATLAB and Python. The FITS Support Office at NASA/GSFC maintains a list of libraries and platforms that currently support FITS. This data format can save data in multi-dimensional arrays. It is one of the most efficient ways of archival storage of astrophysical data thanks to the table containing information storage and header keywords. In this project, we used “SAP_FLUX”, “TIME” headers.

2.DEVELOPING ASTROPHYSICAL DATA MANAGEMENT & PROCESSING TOOLS

2.1 MobaXterm

MobaXterm is an improved terminal for Windows with an X11 server, a tabbed SSH customer and more than a few other network tools for distant figuring (VNC, RDP, telnet, rlogin). MobaXterm takes all the vital Unix commands to Windows desktop, in a single moveable exe file which works out of the box.

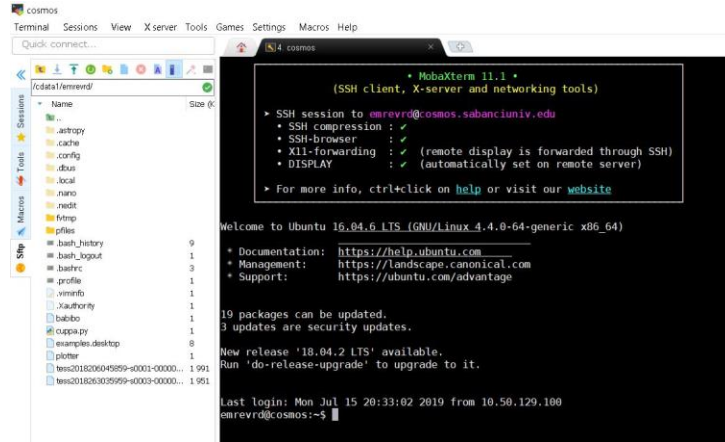


Figure 4. The home page of the Cosmos server in MobaXterm.

In this project we used to MobaXterm to connect the school's server called cosmos which saves TESS data. Due to the data already exist in the cosmos server we run our code in the MobaXterm terminal. It is basically because there is a huge amount of data that we are not able to download. So, it is our gate to reach the data. In addition, we are able to read the data in fits format via MobaXterm. After we write our codes in python, we also run our code in that server.

```
Last login: Tue Aug 6 14:58:57 2019 from 10.50.129.100
emrevrd@cosmos:~$ cd /andromeda2/
emrevrd@cosmos:/andromeda2$ ls
tess
emrevrd@cosmos:/andromeda2$ cd tess
emrevrd@cosmos:/andromeda2/tess$ ls
ffi lc_s1 lc_s10 lc_s11 lc_s2 lc_s3 lc_s4 lc_s5 lc_s6 lc_s7 lc_s8 lc_s9
emrevrd@cosmos:/andromeda2/tess$ cd lc_s1
emrevrd@cosmos:/andromeda2/tess/lc_s1$ ls
tess2018206045859-s0001-0000000008195886-0120-s_lc.fits  tess2018206045859-s0001-0000000234510995-0120-s_lc.fits
tess2018206045859-s0001-0000000008196058-0120-s_lc.fits  tess2018206045859-s0001-000000234511278-0120-s_lc.fits
tess2018206045859-s0001-0000000008196173-0120-s_lc.fits  tess2018206045859-s0001-000000234511396-0120-s_lc.fits
tess2018206045859-s0001-0000000008196285-0120-s_lc.fits  tess2018206045859-s0001-000000234511748-0120-s_lc.fits
tess2018206045859-s0001-0000000008196324-0120-s_lc.fits  tess2018206045859-s0001-000000234512382-0120-s_lc.fits
tess2018206045859-s0001-0000000008196402-0120-s_lc.fits  tess2018206045859-s0001-000000234512399-0120-s_lc.fits
tess2018206045859-s0001-000000024195569-0120-s_lc.fits  tess2018206045859-s0001-000000234513372-0120-s_lc.fits
tess2018206045859-s0001-000000024195690-0120-s_lc.fits  tess2018206045859-s0001-000000234513798-0120-s_lc.fits
```

Figure 5. Commands and the files in the Cosmos server in MobaXterm.

2.2 Linux commands

In order to reach the TESS data, which is stored in the university server called cosmos, we need to learn the basics of Linux commands. The most used commands are as following;

- cd, allows user to change directories
- ls, shows all major directories and files
- mkdir, creates a directory
- rmdir, removes a directory
- cp, copies a directory or a file
- gedit, helps editing files
- idle, directly open a python idle
- python, opens a python file
- fv, open fits data with headers
- mv, move a directory or file

These are the common commands that help us in this project. In addition to these commands, we use the python which is already downloaded on our server. Since TESS data is stored in the cosmos, we can easily reach those data by using the downloaded python.

2.2 Python and fits data format

In order to detect the periodical behaviors in data, it is required to filter, process and plot the data. We used Python programming to do these procedures. [1] As it is mentioned, we use the TESS data from our server called Cosmos. This data was downloaded for us into server and categorized in terms of sectors. There are 11 folders for 11 different sector and in each file, there are many fits format data files for one star in that sector.

Select	TIME	TIMECORR	CADENCENO	SAP_FLUX	SAP_FLUX_ERR	SAP_BKG
D	E	J	E	E	E	E
All	BJD - 2457000, days	d		e-/s	e-/s	e-/s
Invert	Modify	Modify	Modify	Modify	Modify	Modify
1	1325.2944031	4.0942650E-05	70444	2.8373990E+03	7.1018030E+00	8.3952440E+02
2	1325.2957920	4.0963950E-05	70445	3.2069930E+03	7.3880740E+00	8.3715250E+02
3	1325.2971809	4.0985050E-05	70446	3.3925970E+03	7.5234860E+00	8.3650870E+02
4	1325.2985699	4.1006270E-05	70447	3.4090220E+03	7.5353260E+00	8.3532280E+02
5	1325.2999588	4.1027480E-05	70448	3.4324190E+03	7.5679330E+00	8.4306450E+02
6	1325.3013477	4.1048700E-05	70449	3.4643650E+03	7.5795270E+00	8.4146650E+02
7	1325.3027366	4.1069930E-05	70450	3.4767920E+03	7.5901210E+00	8.4042000E+02
8	1325.3041255	4.1091170E-05	70451	3.4680250E+03	7.5889700E+00	8.4385380E+02
9	1325.3055144	4.1112410E-05	70452	3.4549950E+03	7.5843550E+00	8.4880180E+02
10	1325.3069033	4.1133650E-05	70453	3.4831640E+03	7.6041090E+00	8.4666670E+02
11	1325.3082922	4.1154900E-05	70454	3.4724840E+03	7.5997510E+00	8.5241390E+02
12	1325.3096811	4.1176160E-05	70455	3.5359520E+03	7.6516720E+00	8.5675040E+02
13	1325.3110700	4.1197420E-05	70456	3.4347030E+03	7.5757590E+00	8.5351730E+02
14	1325.3124589	4.1218690E-05	70457	3.4616610E+03	7.6024300E+00	8.6107700E+02
15	1325.3138479	4.1239960E-05	70458	3.5321810E+03	7.6544710E+00	8.6065140E+02
16	1325.3152368	4.1261250E-05	70459	3.4591800E+03	7.6050410E+00	8.6517240E+02
17	1325.3166257	4.1282530E-05	70460	3.4834490E+03	7.6270450E+00	8.6717710E+02
18	1325.3180146	4.1303820E-05	70461	3.4498590E+03	7.6048390E+00	8.7107780E+02
19	1325.3194035	4.1325120E-05	70462	3.4774040E+03	7.6307280E+00	8.7997200E+02
20	1325.3207924	4.1346420E-05	70463	3.4733500E+03	7.6305780E+00	8.7811750E+02

Our primary mission was reading that fits format data and reaching the “header” which are related to our project (fig 6). These headers are “SAP_FLUX” and “TIME”.

Figure 6. The fits formatted data with headers in the Cosmos server in MobaXterm.

We tried to reach that data and store it in a Python list to plot it. For this purpose, we use AstroPy library in Python. This library allow us to read fits formatted data and transform it to “numpy.ndarray” type data. Then we transformed that data into a list to plot it. We used matplotlib library to plot our data. We firstly plot our data flux vs. time (fig 7).

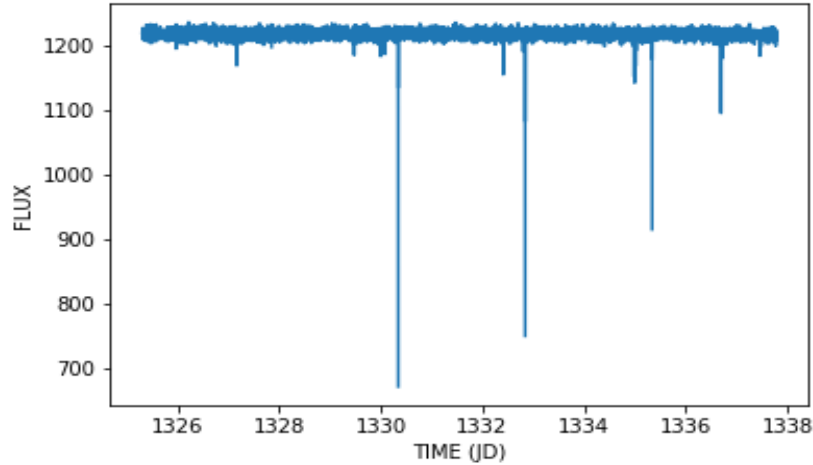


Figure 7. Flux vs. Time graph with errors.

It can be seen that, there are error which is caused by detector errors or other light sources in the sky in our plot. We developed an algorithm to get rid of that errors and reach a data which has as minimum error as possible. With that algorithm, we erased the corrupted data from our data set. These corrupted data were showed “nan”. After that process, we also filter our data from huge instant changes due to misreading of detectors. We create an algorithm to detect these instant changes and erase them. This algorithm reads all the flux data from a list and looking average for every 5 consecutive flux value. If the extraction of the flux value from the average value is too high and this instant increase or decrease is not continuing, the algorithm removes that value from flux list and the corresponding time value from time list. After that filtering process, we plot the new and processed data (fig 8). In order to make these calculations, we also use NumPy library on Python. This processed data became ready for other processes like Fast Fourier Transform (FFT).

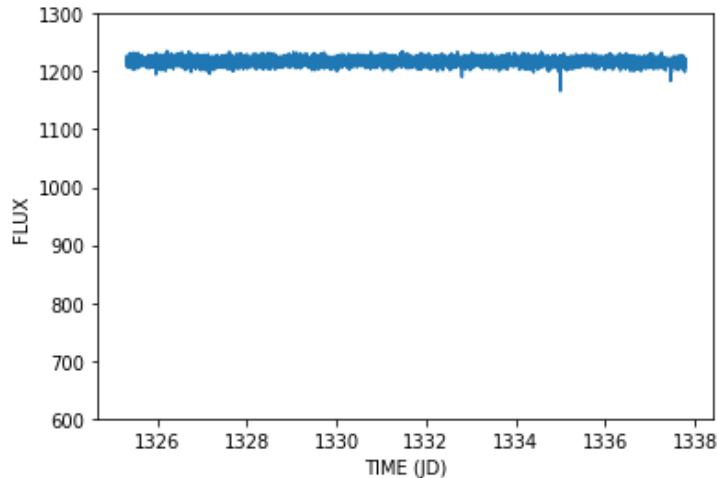


Figure 8. Filtered Flux vs. Time data graph.

2.3 Fast Fourier Transform (FFT)

Fast Fourier Transform (FFT) is a computational tool for examining waves. Fourier Transform basically decomposes a wave into harmonic sine and cosine waves. However, FFT does the same process with discrete but equal intervals. FFT is used in digital spectral analysis, filter simulation and related fields. Discrete Fourier Transform examines time and frequency domain as periodic. [5] The difference of FFT is that FFT is the computational version of DFT and it is more efficient than DFT. Also, FFT reduces the computation time excessively. [3]

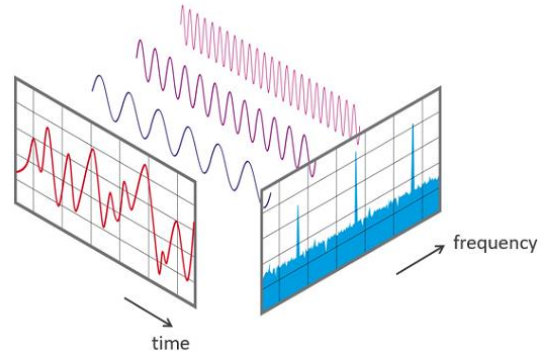


Figure 9. A model for showing how does FFT work

We use Fast Fourier Transform (FFT) to clear noise from our data and highlight the periodical behaviors in our data. After we implement the FFT method to our data, we are expecting to reach the frequency of the most powerful periodical behavior. In order to implement FFT to our data we use NumPy and Scipy.FFTpack library in Python. We implement FFT function to the filtered data. This function gave use a NumPy.Ndarray which includes complex numbers. To transform this array with complex numbers, we use that equation:

$$Power = \frac{2 * |fft\ value|^2}{(sum\ of\ all\ flux\ value)}$$

With that equation, we achieved a list of real numbers. This list is power spectrum list which is created by the implementation of FFT to flux data. After we reached power spectrum, it is also required to create a frequency list from time data. We created this frequency list. The minimum element of this list is one over length of time interval.

$$f_{min} = \frac{1}{(time_{max} - time_{min})}$$

And the maximum element of the frequency list is one over two times of the difference in two consecutive time value.

$$f_{max} = \frac{1}{2 * (difference\ between\ two\ consecutive\ time\ value)}$$

This list will start with minimum element and end with maximum element. Elements will increase by the amount of minimum frequency from minimum frequency to maximum frequency. With these computations, we create the list of frequencies. After these implementations, we have achieved power spectrum and frequency lists instead of flux and time lists.

In this project, to understand FFT we first created a pure sine wave using python. Since we create the wave, we know its frequency. We add some random numbers to sine wave and these numbers will behave as noise which can be seen in the TESS data. [5] After achieving the wave with noise, we apply FFT and observe whether it will give the expected frequency or not. We used the methods which was explained in the previous paragraph to this wave. As expected, the results of FFT gives the frequency that we determined. Thus, the working principle of FFT can be understood in detail.

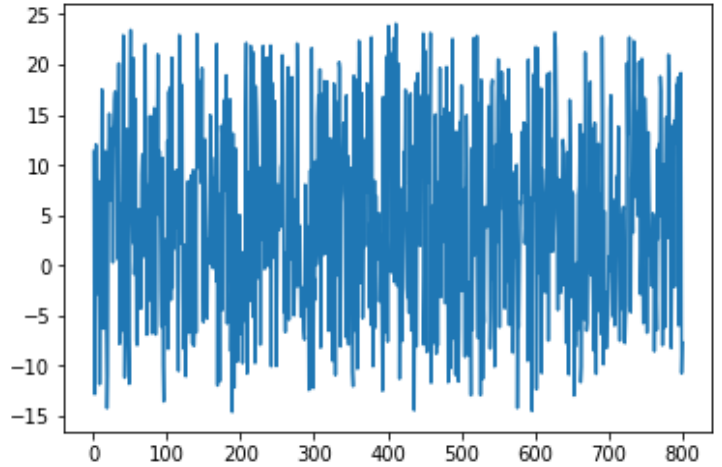


Figure 10. A noise added sinus wave with 0.3 frequency.

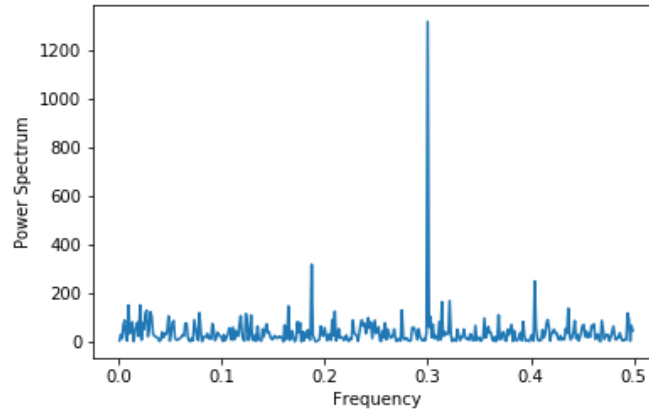


Figure 11. The power spectrum vs. frequency graph of FFT implemented wave in figure 10.

After we understand the principle of FFT thanks to this simple example, we implement these methods to our data. After the implementation of FFT methods, we plotted a new graph with power spectrum vs. frequency data.

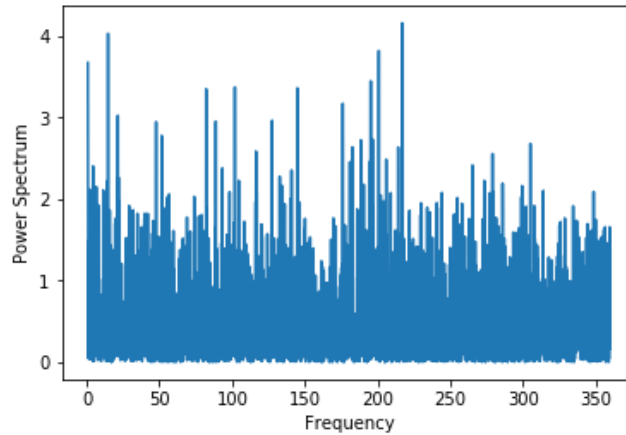


Figure 12. The FFT implemented TESS data which is in figure 8

3. Sample Implementations of Our Code

We implement our code on some of the different data which picked from Cosmos server. Here is the flux vs. time and power spectrum vs. frequency graphs for different stars:

a)

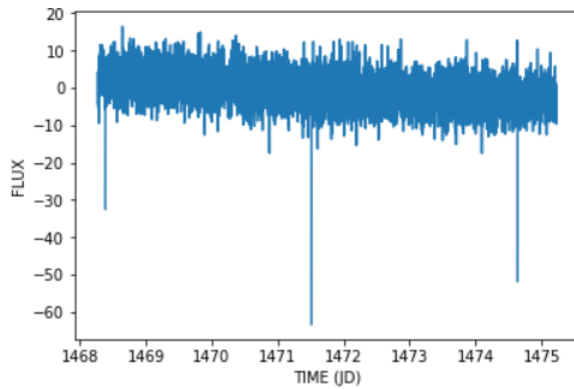


Figure 13. Flux vs. Time graph from a star in sector 1

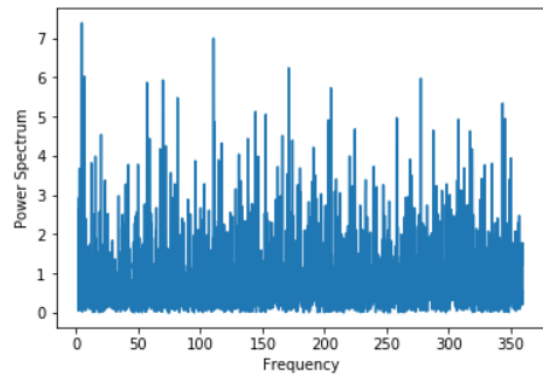


Figure 14. Power spectrum vs. frequency graph from a star in sector 1

b)

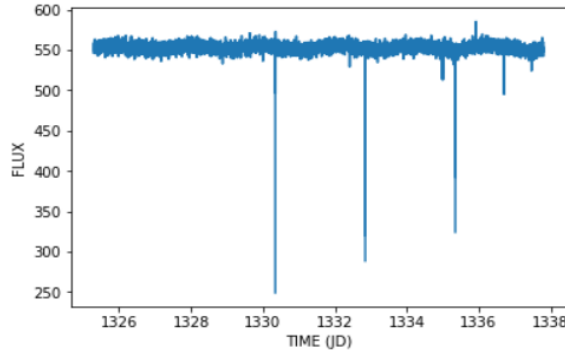


Figure 15. Flux vs. Time graph from a star in sector 7

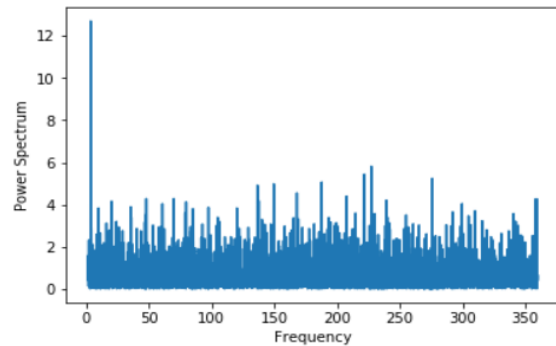


Figure 16. Power spectrum vs. frequency graph from a star in sector 7

c)

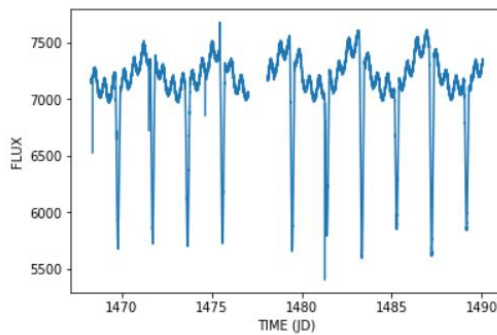


Figure 17. Flux vs. Time graph from a star in sector 6

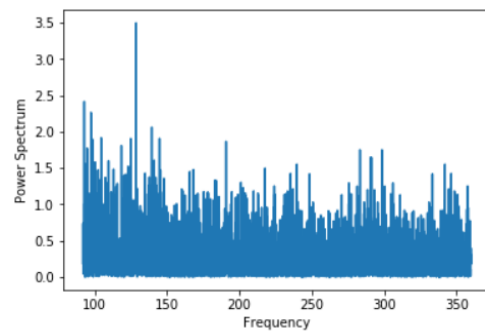


Figure 18. Power spectrum vs. frequency graph from a star in sector 6

d)

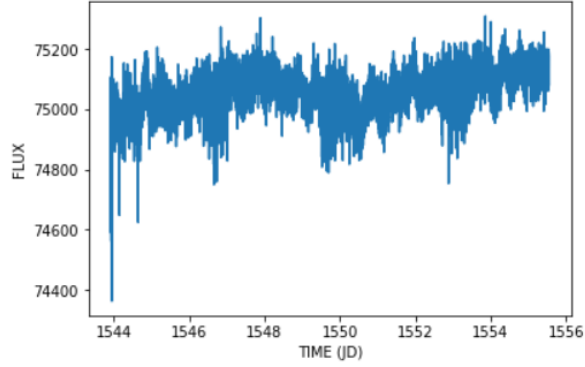


Figure 19. Flux vs. Time graph from a star in sector 9

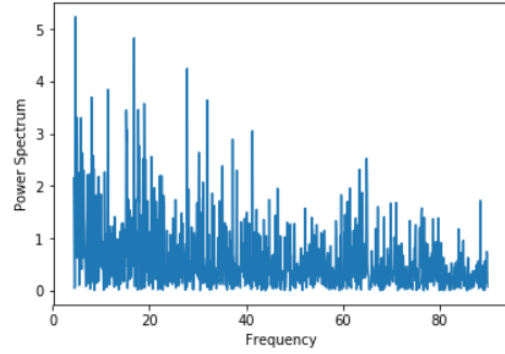


Figure 20. Power spectrum vs. frequency graph from a star in sector 9

4. Analyses of the Graphs

After we implement our code into TESS data, we reached power spectrum vs. frequency graphs for that specific stars which we took data from. As it can be seen, in flux vs. time graphs of most of the stars, there is not a periodical behavior. Because, there is not a significant dip in flux vs. time graphs. These graphs are in fig 8, fig13, fig15 and fig19. Since there are not noticeable dips in flux graphs of that stars, there are not significant increases in power spectrum graphs too. So, we cannot identify any periodical movement in those data.

However, in fig 17, it can be seen that, there are many dips in flux vs. time graph. So, it can be said that, there might be a periodical behavior in that data. It is expected to see a sharp increase in power spectrum vs. frequency graph on the frequency of that periodical behavior, due to these dips. As it is expected, there is a significant increase in power spectrum graph in fig 18. So, it might be the frequency of that periodical behavior we saw in fig17. A closer look for that dips in fig 17, can be seen in fig 21 below.

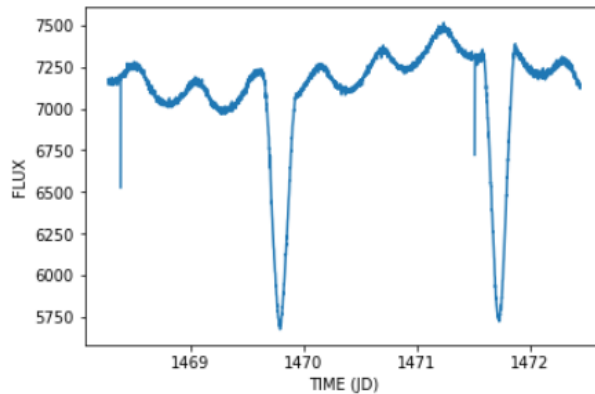


Figure 21. A closer look for the graph in fig 17.

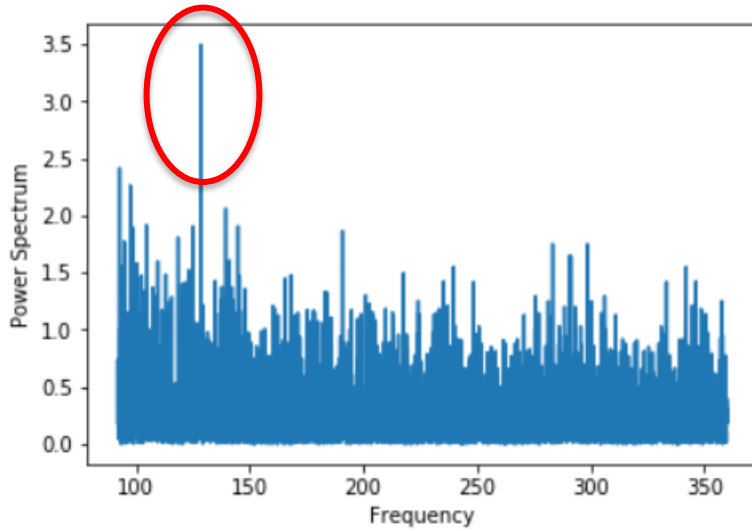


Figure 22. The significant increase in the power spectrum graph of the star in fig 17.

After we identify that noticeable increase in power spectrum graph, we wanted to the reliability of that data. In order to check the reliability, we calculated the chance probability and detection probability. If the detection probability is high, we can say that this peak in the graph is referring a real periodical behavior.

Chance probability and detection probability is calculated with these formulas below.

$$P_{Chance} \sim e^{\frac{-Power_{max}}{2}}$$

$$P_{Detection} = (1 - P_{Chance})$$

Before we implement these formulas to our data, we firstly normalize our power spectrum data to 2. Then we take the maximum power value of the peak and make calculations.

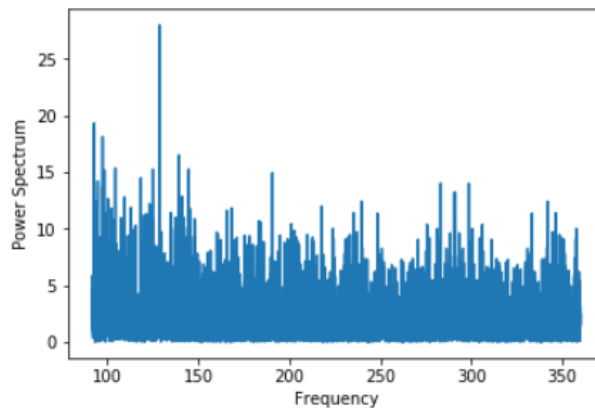


Figure 13. The graph after the normalization. (Maximum power value of data is 27.976)

$$P_{Chance} \sim e^{\frac{-27.976}{2}}$$

$$P_{Chance} \sim 8.415 * 10^{-7}$$

$$P_{Detection} = (1 - 8.415 * 10^{-7})$$

$$P_{Detection} = 0.9999991584$$

As we can see from the detection probability, that peak is reliable enough to say that there is a periodical motion in that specific data (tess2018349182459-s0006-0000000464838153-0126-s_1c). The frequency of that motion is 128.615376.

So, we detect a significant power peak in one of our sample data set and we check if it is reliable enough to consider. In conclusion, we identify a periodical motion in that data.

5. Conclusion and Future Work

In this project we have developed an algorithm which filter and plot TESS data and search for periodical behaviors in flux. By this research, we have learned TESS mission, exoplanet search methods and Fast Fourier Transform (FFT). We implement the code for just some of the data in our data set and we detected a probable periodical motion in one of the sample data.

However, as a future work, it can be tried to implement the code for all of the TESS data and looking for more significant periodical behaviors in all stored data in Cosmos server. In addition to that, the stars, which has periodical motion in front, can be examined more to discover new celestial bodies.

References

- [1] Barclay, T. (2019, June 27). TESS. Retrieved from <https://heasarc.gsfc.nasa.gov/docs/tess/>
- [2] Cochran, W. T., & Cooley, J. W. (1967). What Is The Fast Fourier Transform? Proceedings of the IEEE, 55(10), 1664-1674. Retrieved August 6, 2019.
- [3] Exoplanet Exploration: Planets Beyond our Solar System. (2019). Retrieved 6 August 2019, from <https://exoplanets.nasa.gov/>
- [4] Jenkins, J. M. (2016). Processing and managing the Kepler mission's treasure trove of stellar and exoplanet data. 2016 IEEE International Conference on Big Data (Big Data). doi:10.1109/bigdata.2016.7840971
- [5] Klis, M. (1989). Fourier Techniques in X-Ray Timing. Timing Neutron Stars, 27-69. doi:10.1007/978-94-009-2273-0_3
- [6] Ricker et al. 2015, Journal of Astronomical Telescopes, Instruments, and Systems, Volume 1, id. 014003