Using Machine Learning to Discover Exoplanets

Göktuğ Gökmen

Ankara University Department of Electrical and Electronics Engineering

goktuggokmen_2000@hotmail.com

Yusuf Emre Baysal

Ankara University Department of Electrical and Electronics Engineering

yusufemrebaysal99@gmail.com

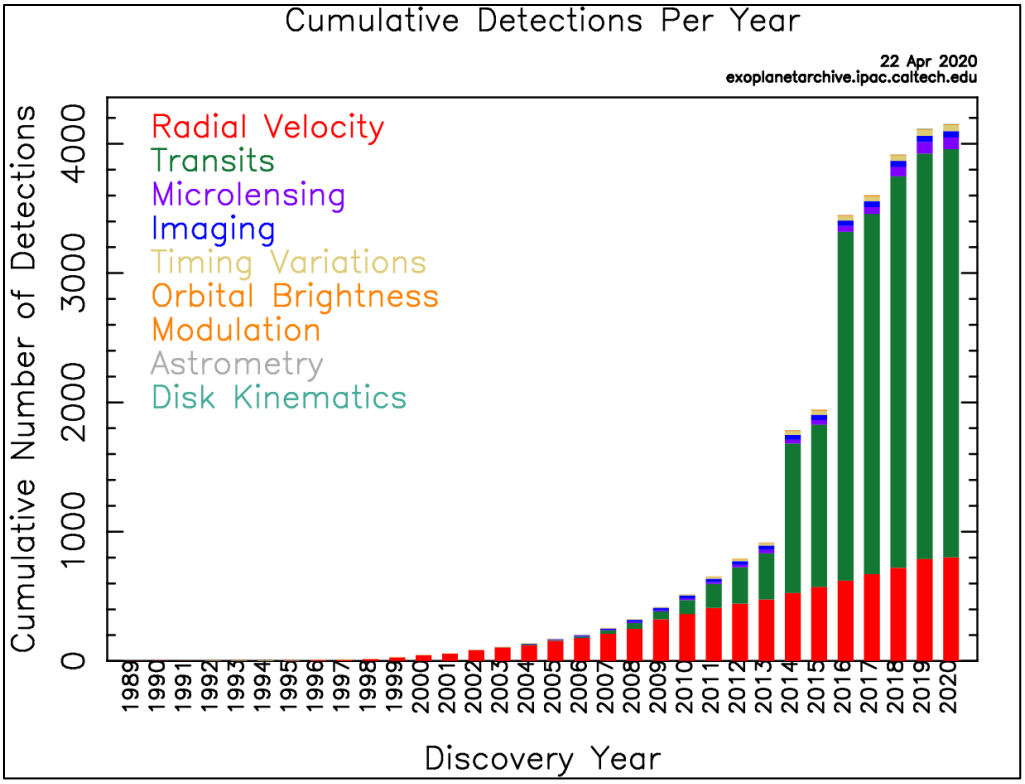
Abstract- For years, scientists have used data from NASA's Kepler Space Telescope to look for and discover thousands of exoplanets. This procedure is time consuming and complicated. To speed up the process of discovery, we designed two machine learning models to train with the data provided from Kepler Space Telescope's Campaign-3 and to spot candidate systems where there could be exoplanets present. Our study shows that machine learning can be used to predict these phenomena reliably. In both models, accuracy of the predictions is over %90.

I. Introduction

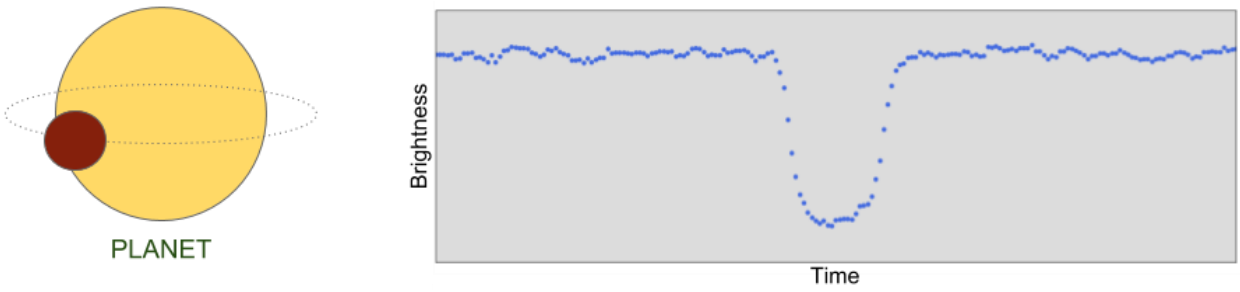
Spotting exoplanets is an important part of astronomy, so much so that NASA launched Kepler space telescope into the orbit in 2009, with the sole purpose of finding new exoplanets. There are several methods to spot exoplanets but the most conventional one is called "transit photometry".

Transit photometry method is quite simple, but also effective. It tracks the luminous flux (light intensity) of a target star. If there is a celestial body in the orbit of the star, the light emitted from the star "dims" periodically since the object blocks the light from the star. There are 3333 planets found by this method and there is more to come.

The biggest problem in this method is the sheer number of stars around us. Manually checking each flux value of thousands of stars for years is simply not possible. This is where machine learning comes into the equation. If we can train an algorithm to find the candidate systems where there could be an exoplanet, it would decrease the time it takes to find new exoplanets exponentially, and this is our aim for this project.

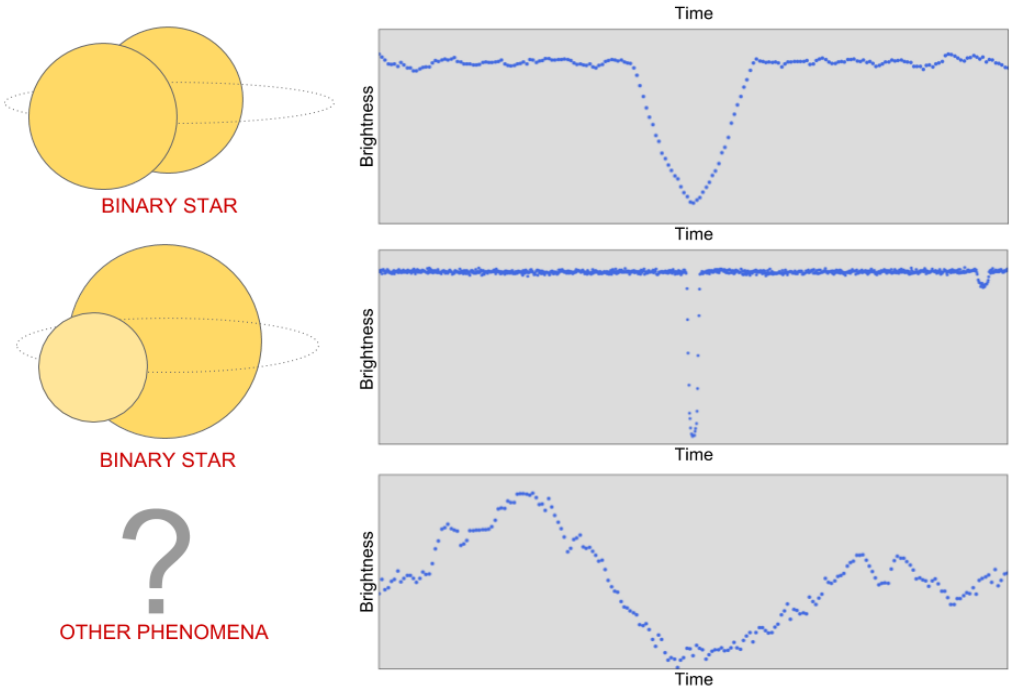


Exoplanet Discoveries Per Year



An Exoplanet in Transit

While transit method looks perfect for discovering exoplanets, this phenomenon can happen in different scenarios. However, we will be looking at exoplanet scenarios, not other phenomena.



II. Dataset

Dataset is highly inconsistent with small number of discovered exoplanets. There is also instrument noise in the readings. Dataset is split into two parts.

Train set:

5087 rows or observations. 3198 columns or features. Column 1 is the label vector. Columns 2 - 3198 are the flux values over time. 37 confirmed exoplanet-stars and 5050 non-exoplanet-stars.

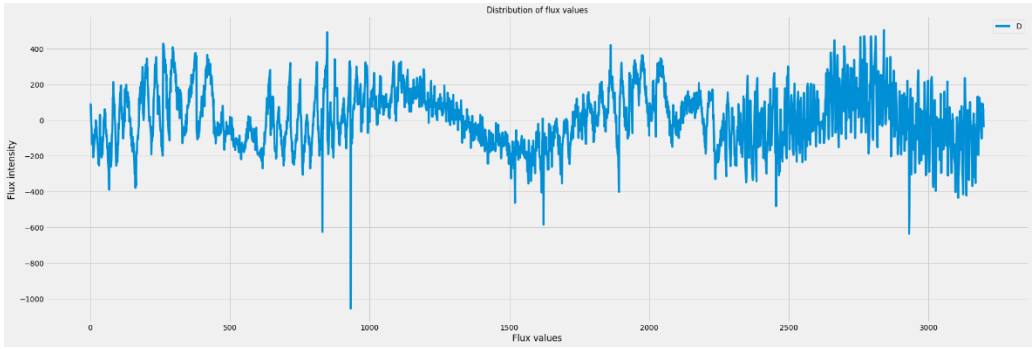
Test set:

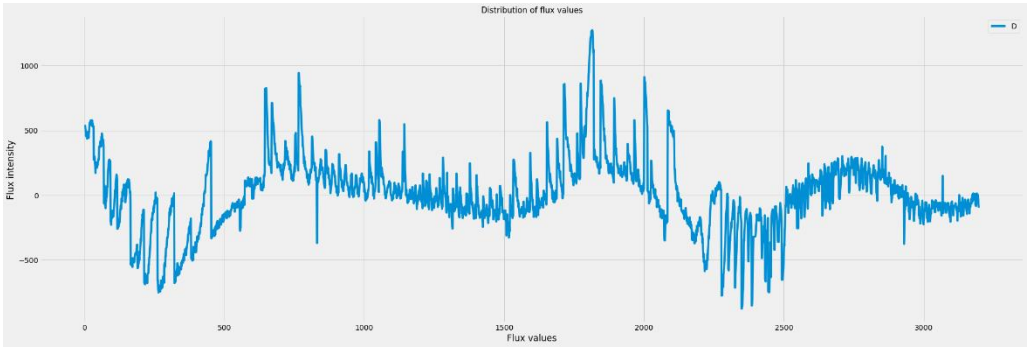
570 rows or observations. 3198 columns or features. Column 1 is the label vector. Columns 2 - 3198 are the flux values over time. 5 confirmed exoplanet-stars and 565 non-exoplanet-stars.

The data presented here are cleaned and are derived from observations made by the NASA Kepler space telescope. Over 99% of this dataset originates from Campaign 3. To boost the number of exoplanet-stars in the dataset, confirmed exoplanets from other campaigns were also included along with all observations from Campaign 3. The datasets were prepared late-summer 2016.

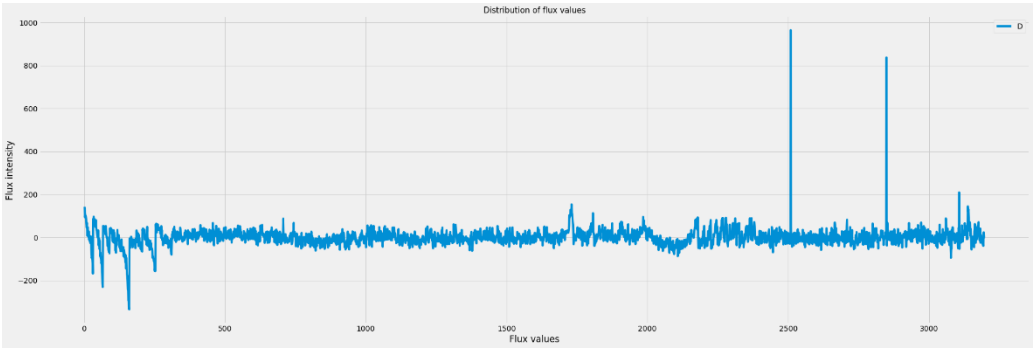
Then, what is our findings from dataset? Let's look at the graphs from our dataset, plotted by matplotlib library via Spyder.

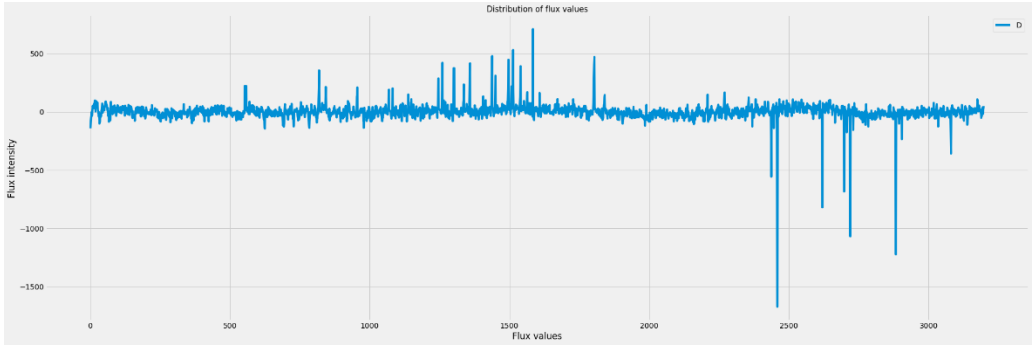
Index 0 and index 2 exoplanet-star line plots respectively.





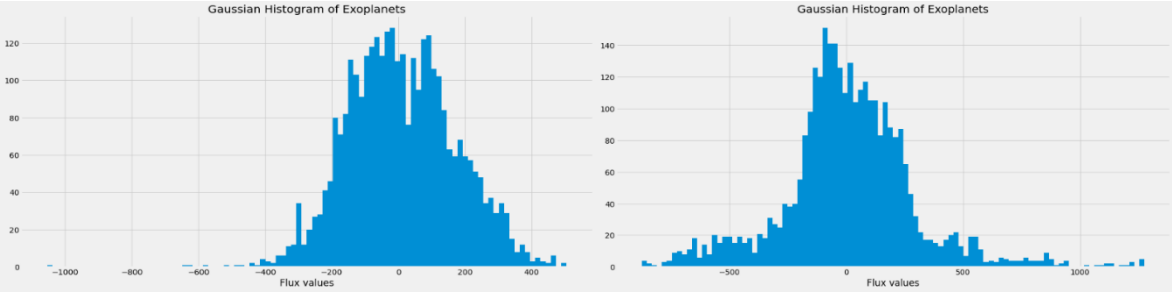
Index 37 and index 39 non-exoplanet-star line plots respectively.



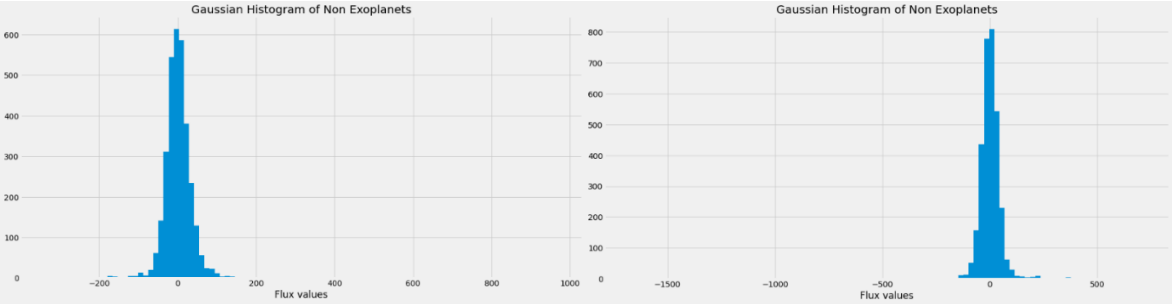


We can see that luminous flux changes periodically in stars with exoplanets. We can observe this phenomenon better in gauss distribution histograms.

Index 0 and index 2 exoplanet-star histogram graphs respectively.



Index 37 and index 39 non-exoplanet-star histogram graphs respectively



III. Methods

We used two different methods to train our algorithm with the given dataset. The methods used are Decision Tree Machine Learning Algorithm and Convolutional Neural Network Deep Learning Algorithm (CNN).

A. Convolutional Neural Network

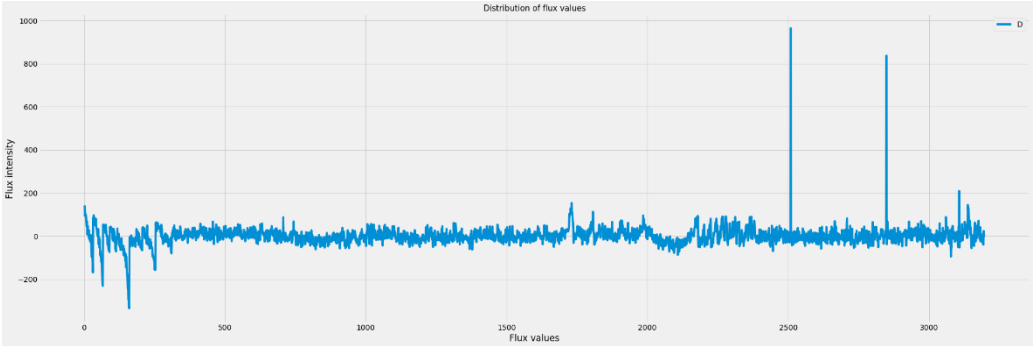
Our convolutional neural network contains an input layer named C1, two hidden layers C2 and C3, and lasty the sigmoid output layer C4. Having two hidden layers is optimal for a network with a dataset of this size. Epoch of our model is 30 but we also used early stopping to eliminate the chance of overfitting. Since our dataset is quite noisy, we used Adam gradient decent optimizer.

Network Name	Block Name	Layer Name	Number of Kernels	Kernel Size	Stride	Number of Neurons	Neuron Type
NW1	C1	Conv1 Pool1 Dropout1 Flatten1	10 2 - -	2 2 - -	2	3196 1598 1598 15980	ReLU
	C2	Dense1 Dropout2	-	-	-	48 48	ReLU
	C3	Dense2	-	-	-	18	ReLU
	C4	Dense3	-	-	-	1	Sig

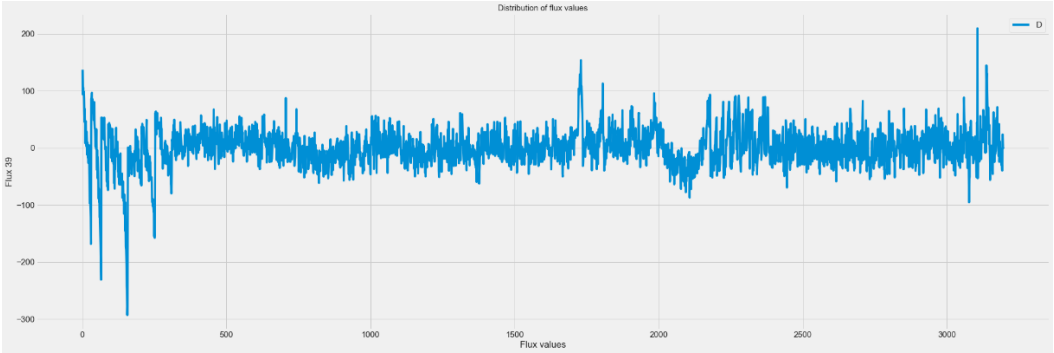
1)Pre-processing

The dataset labelled the exoplanet stars with number 2 and non-exoplanet stars are labelled with 1. Due to the sake of convenience, we chanced the labels with 1 and 0 respectively.

There are significant number of outliers in the dataset because of instrument noise, transmission errors and other space phenomena that is not exoplanets. In order to train our model better, we tried to reduce the weight of these outliers. We substituted the mean flux of each star from its maximum flux value to eliminate the outliers.

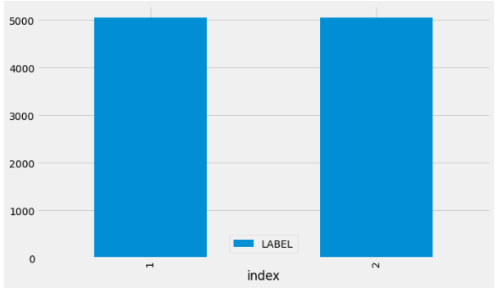


Flux Values with Outliers



Flux Values without Outliers

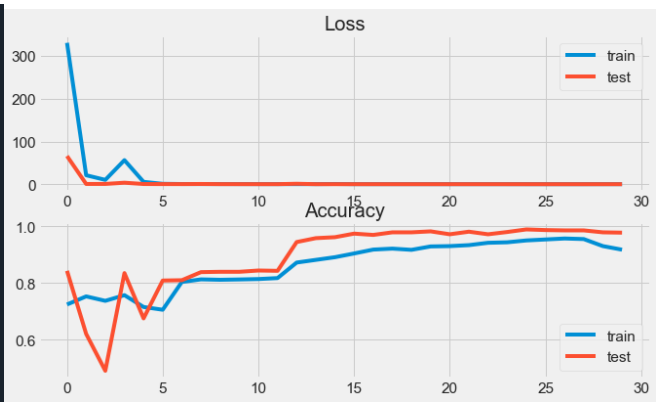
The next big problem with our dataset is the lack of sufficient stars with exoplanets to train our model. There are only 37 exoplanet systems even though there are over 5000 non-exoplanet ones. To counteract this problem, we increased the number of exoplanet systems such that the number of exoplanet and non-exoplanet systems are the same. We used SMOTE method to artificially increase the dataset observations.



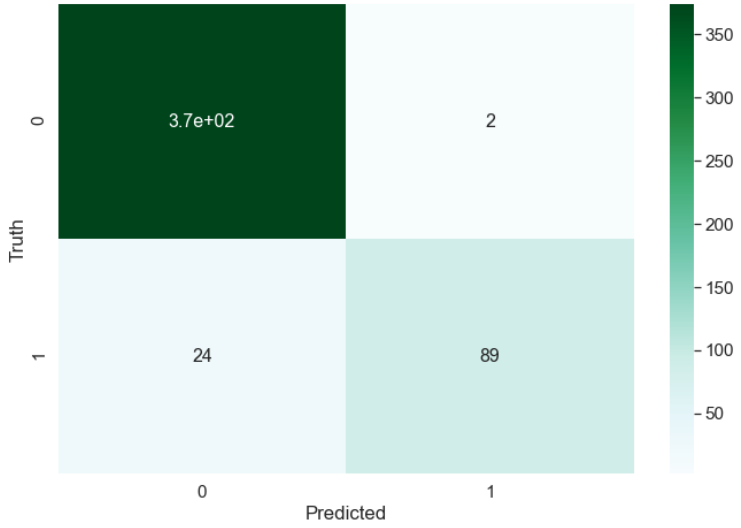
2)Results

Our finding show that the model can correctly predict the outcome with %91 accuracy in the test dataset.We can see that the accuracy begins to settle after the 15th iteration of the algorithm. If it were to happen earlier, the early stopping function would have stopped the algorithm if the accuracy had stayed the same for 7 or more iterations.

```
          Prediction Metrics

Accuracy:     0.910
Precision:    0.926
Recall:       0.664

F1 Score:     0.773
ROC AUC:      0.860
Balanced
Accuracy:     0.824

Sensitivity:      0.664
Specificity:      0.984
Geometric Mean:   0.808
```



The confusion matrix shows that the model predicted almost all of the non-exoplanet systems. 89 of the 113 exoplanet systems are guessed correctly and only 24 of the 113 exoplanet systems is predicted as non-exoplanet system. Also, there are 2 false positives that is considered as exoplanet systems.



B. Decision Tree

The second method we used is decision tree machine learning algorithm. The working principle of decision tree algorithm is suited very well for our dataset. There are portions of observations in which the flux lowers significantly because of the exoplanets. If the branches of the decision tree can split the data where the dimming happens, it can perform better for classification and regression.

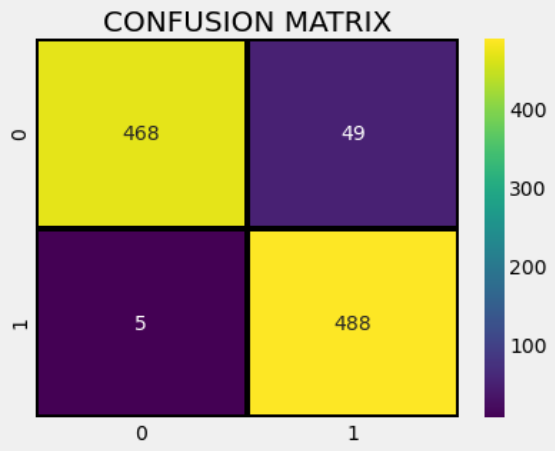
1)Pre-Processing

The same methods we used for pre-processing in the CNN part is also used in this part. Normalization of the data and artificial oversampling is needed to create a tree model that is accurate.

2)Results

The decision tree model predicted the type of system with the accuracy of %97 for non-exoplanet systems and %91 for exoplanet systems. The biggest problem in decision tree algorithms is overfitting, which can be seen in this case. The decision tree split more than needed which caused false positives in the predictions. Only 5 of the 493 exoplanet systems wasn't detected by the model.

```
Validation accuracy of Decision Tree is 0.9396039603960396

Classification report :
              precision    recall  f1-score   support

           1       0.97      0.91      0.94       517
           2       0.91      0.97      0.94       493

    accuracy                           0.94      1010
   macro avg       0.94      0.94      0.94      1010
weighted avg       0.94      0.94      0.94      1010
```



IV. Comparison

Both methods performed relatively well in predicting the test dataset. Raw numbers suggest that decision tree method performed better than CNN. For a sample of this size, it was expected since there were not enough data to create an optimal convolutional network. But if we give a detailed look to the results, we can see that the decision tree model produced more false positive results which can be a big problem in bigger datasets. Since this algorithm is designed to work on the universe, the biggest dataset there is, this problem could worsen exponentially. With these finding in mind, we determine that CNN is a better fit for this task.

V. Conclusion

As we suggest in the comparison part, CNN method is the better option to undertake this problem since the larger sample sizes can affect decision trees accuracy to the point where the branches get so complex the model becomes unusable. If we use a more balanced dataset, the accuracy of the CNN model can be increased. Also, more convolutional layers can produce a better model for learning.

If we look at this project in broader terms, machine learning can be revelational for the field of astronomy. This project is an example that astronomical phenomena can be studied with the help of machine learning. Spotting exoplanets is just the beginning to what we can achieve by using this technology to study space in a better way.

VI. References

https://www.kaggle.com/srahuliitb/rsk-exoplanetshunt-randomforest

https://www.kaggle.com/antonzv/exoplanet-hunting-top-score-using-smote-and-cnn

https://www.kaggle.com/max398434434/classification-with-highly-imbalanced-data

https://www.kaggle.com/arjunsingh88/exoplanet-prediction-cv-accuracy-100

https://www.kaggle.com/nageshsingh/exoplanet-exploration-using-ml

https://medium.com/dvlpr/exoplanet-hunting-in-deep-space-with-machine-learning-4db85d5f7769

https://www.kdnuggets.com/2020/01/exoplanet-hunting-machine-learning.html

https://ai.googleblog.com/2018/03/open-sourcing-hunt-for-exoplanets.html

https://lweb.cfa.harvard.edu/~avanderb/kepler90i.pdf

https://lweb.cfa.harvard.edu/~avanderb/Deep_Learning_2.pdf