Hunting Extra-Solar Planets using Machine Learning

Anurag Malik1, Ishita Rastogi2, Ishvinder Singh3, Kartik Tiwari4, Sanskriti Agarwal5, Shivansh Mathur6

Associate Professor1

B.Tech Scholar2,3,4,5,6

Department of Computer Science & Engineering, Moradabad Institute of Technology1,2,3,4,5,6

Moradabad, India

[anurag\_malik@rediffmail.com1](mailto:anurag_malik@rediffmail.com1), [ishitarastogi2324@gmail.com2](mailto:ishitarastogi2324@gmail.com2), [ishvinder133@gmail.com3](mailto:ishvinder133@gmail.com3),

[tkartik88@gmail.com4](mailto:tkartik88@gmail.com4), [sanskriti02911@gmail.com5](mailto:sanskriti02911@gmail.com5),[shivanshmathurr@gmail.com6](mailto:shivanshmathurr@gmail.com6)

*Abstract*—*In this paper, we are going to use machine learning techniques for the detection of exoplanets based on the dataset provided by Kepler. Astronomers are already working on the detection of such planets using different techniques but here we will be using transit approach and train different models for the given data. The different models will have different accuracy and will be compared after the complete training of the model and the model with the maximum accuracy will be considered. According to the data till November 18,2021, Kepler has confirmed the discovery of 2402 planets and have shown 2361 planets as true positives. We aim at discovering more such exoplanets or search about the discovered true positives. Our approach will be comparatively less time consuming as compared to the already used approaches.*

Keywords—***exoplanets , machine-learning , Kepler, TESS,***

astronomy, transit approach

# Introduction

The study of planets, stars and other astronomical objects has always fasinated the human species. From the discovery of Moon in 1610 till today we have come a long way in our search of celestial bodies. Just like Earth and it's other siblings that revolve around our native star Sun, we have discovered other stars and planets that revolve around them. The planets that revolve around star other then Sun are called Exoplanets or Extra Solar Planets.

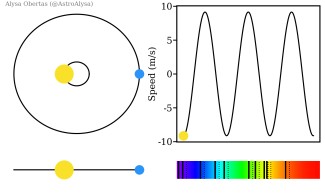
Since the lauch of Kepler satellite, the number of discovered exoplanets have increased significantly and in 2018 NASA launched TESS (Transiting Exoplanet Survey Satellite) aimed specifically to find more such planets the list is expected to grow more. However the task is not very simple due to complex data analysis and manual processing of this huge data, but with advancements in technology the process is being machine-controlled with the assistance of Machine Learning and Deep Learning techniques.

There are various ways to detecting exoplanet signals. Some of them are :

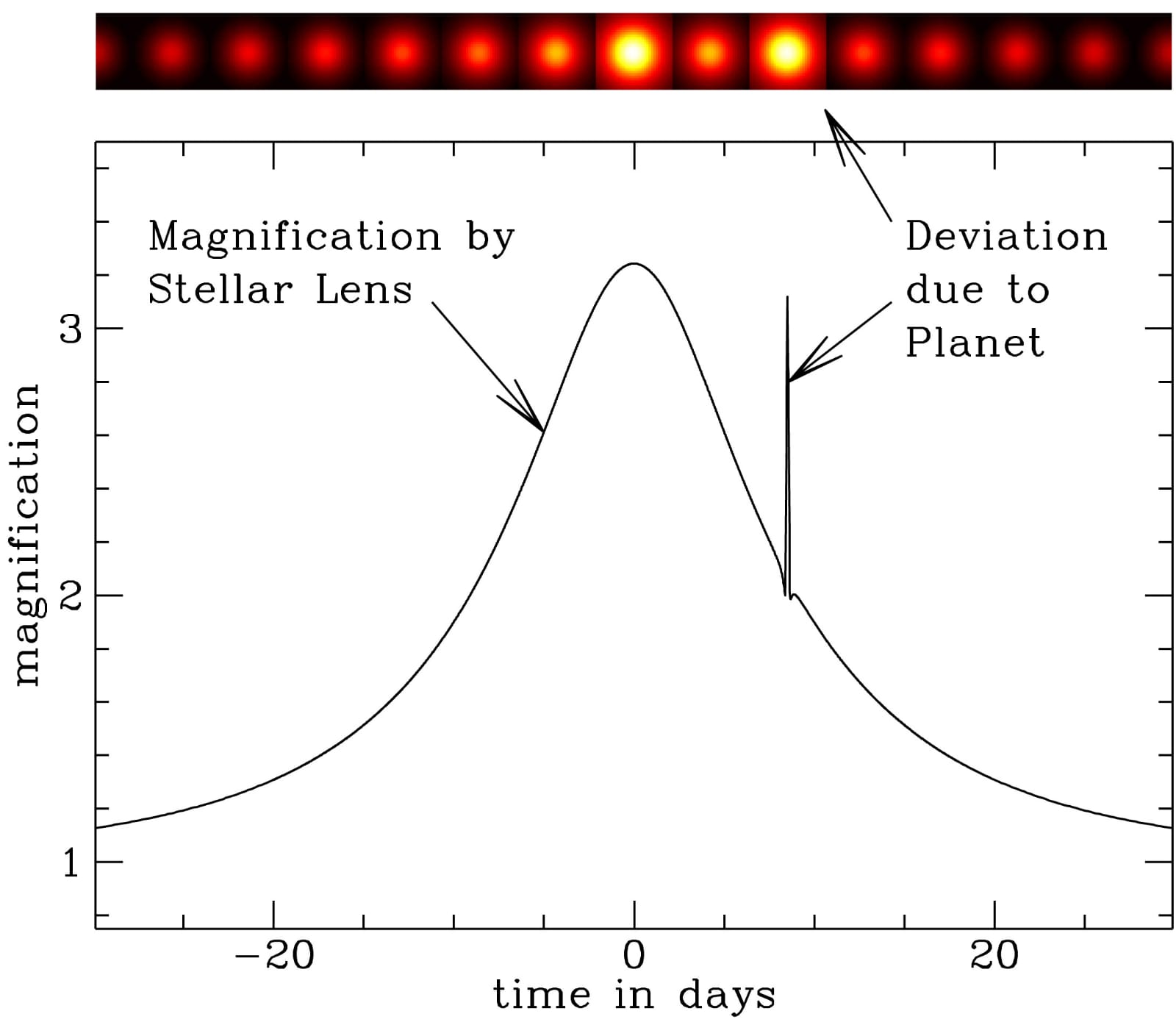
**Direct Imaging[4]** The exoplanet is imaged directly with the help of massive telescopes fitted with accommodative optics and coronagraphs. The technique is most sensitive to the hotter, bright (young) and large exoplanets on wide and/or eccentric orbits (large sky projected separations). The separation from the host star permits for spectra to be obtained directly and permits for the direct activity of the light.

Figure 1: Direct Imaging method of exoplanet detection

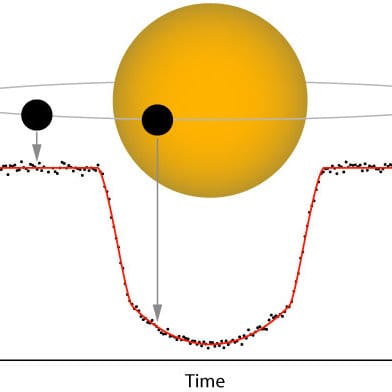
**Radial Velocity[5]** The exoplanet is detected by measure the Doppler effect within the host star light, a consequence of the attractive force affects between the 2 bodies. The technique is most sensitive to exoplanets with an outsized mass orbiting near to their host star perpendicular to the plane of the sky. The radial velocity technique permits for a minimum mass (dependant on orbital inclination) to be calculated.

Figure 2: Radial Velocity method

**Microlensing** The exoplanet is detected by measuring characteristic light curve changes caused by changes within the lensing effect ascertained once a star with a planet passes in front of a remote star. The technique is restricted to distant one time events and by the shortage of correct determinations of the world and orbit parameters. It is however a really valuable technique because of the lack of strong radii or mass biasses creating it ideal for applied math population studies.

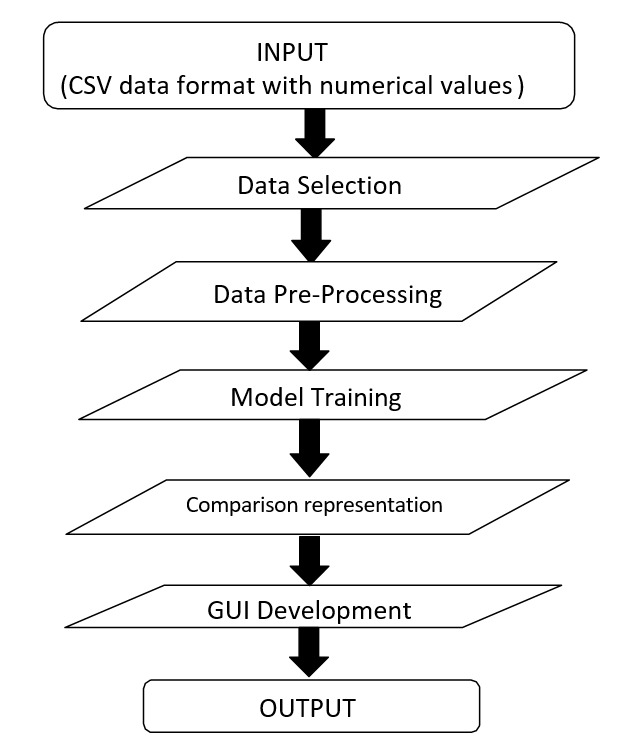
Figure 3: Graph for exoplanet detection using Microlensing

**Transit** The exoplanet is detected by measuring a periodic decrease within the flux received from the host star, as a consequence of the exoplanet transiting ahead of the host star. The transiting technique is most sensitive to massive exoplanets orbiting near to their host star stars and provides an correct determination of the planetary radius relative to the host star. It is also the most popular and successful method and till date more than 2000 exoplanets have been found using Transit Method.

Figure 4: Transit method of exoplanet detection

In this paper, we are going train some Machine Learning models to sight whether or not a given object is an exoplanet candidate or not using the data available on Mikulski Archive for Space Telescopes also called as MAST archive.

The general flowchart of our approach is as follows

Figure 5: General solution flowchart

# Problem Statement

We are trying to determine whether a signal is exoplanet candidate or not using transit approach with the help of different machine learning models. The aim is to find planets which may be a bit similar to Earth and will help us to identify if there is any other system existing which is similar to our solar system. The major technique used till now is the Transit approach which includes the monitoring of dips in the brightness of the stars and identifying the appearance of some planets which are causing these dips. But currently , the graphs of the these dips are plotted manually and also monitored manually so we focus on discovery of these planets without using manual approach.

# Existing Solutions

According to our litreture survey, there has been multiple solutions given for automating the vetting process of exoplanets and even today new discoveries and new exoplanets are being discovered. There are various methods of detecting exoplanets but in our survey we found that Transiting Method is the most used as well most successful method among all other methods.

Deep Learning approaches made by Christopher J Shallue and Andrew Vanderberg[1] showed high results with around 98% accuracy. There model was able to successfully discover two new exoplanet belts around stars Kepler-80 and Kepler-90.

Other approaches used classification models like Decision Tree Classifier, Random Forest Classifier, SVM, KNN, etc were trained and tested. In a paper by Abhishek Malik, Benjamin P Moster and Christian Obermeier[2], a gradient boosted tree model was trained over 789 extracted features and it yeilded accuracy of 91%.

Shafi in his article 'Exoplanet detection using AI'[3] talked about hunting extra solar planets using CNN and SVM and trained them on combined data of Kepler, K2 and TESS.

Pat Brennan[8] found an circumbinary planet called 'TIC 172900988 b' which revolves around two stars and hence categorised as Circumbinary which means a planet revolving around two stars. The planet has been found using the transit method and is found to be a gas giant like Jupyter but even larger in size.

# Proposed Solution

After studying about the early methods used for the exoplanet discovery and their needs we have found the transit method to be the most effective way which can be used for the exoplanet discovery at the best success rate. It will be based on different dips caused due the to the planet crossing a star. These dips will be monitored continuously and the values received will be used for further detection. For the implementation we are using the data provided by MAST (Mikulski Archive for Space Telescopes). The data contains the values for various transit parameters. We will be using machine learning technique to train different models like the logistic regression, decision tree classifier and random forest classifier. These models will be classifying all the signals as either exoplanet candidate or false positive. Now, these models will be working on raw data, standardised data as well as normalised data. For checking which model will be the best we will be comparing these models based on their precision score, f1 score, accuracy, recall and confusion matrix. The one with the best values will be selected for the detection. Also we will be checking on how our models are more efficient than the models mentioned.

## Data Gathering

There are various publically hosted resouces available that contains loads and loads of data. NASA's Exoplanet Exploration Program or ExEP[6] is an online archive that holds observational data from satellites like Kepler, K2 and TESS. Also California Institute of Technology (CalTech) also hosts an online archive that contains tools and data captured by Kepler and TESS satellites. But we have used another publically available resource called as Mikulski Archive for Space Telescopes that contains a plethora of data from various satellites and missions including Kepler, K2, TESS and many more. The data is available is raw format, csv format, .tpf(Target Pixel File) format and we can also obtain data obtained through different pipelines as well. It is extremely fast and has a great search feature that helps to find very specific range of data along with an extremely friendly user interface. Hence MAST is the platform of choice for this project.

Table 1: Some important features in dataset

|  |  |  |
| --- | --- | --- |
| **S No** | **Feature** | **Description** |
| 1 | koi\_disposition | Exoplanet Archive Disposition |
| 2 | koi\_period | Orbital Period |
| 3 | koi\_time0bk | Transit epoch |
| 4 | koi\_duration | Transit Duration |
| 5 | koi\_depth | Transit Depth |
| 6 | koi\_prad | Planetary Radius |
| 7 | koi\_teq | Equilibrium Temprature of planet |
| 8 | koi\_model\_snr | Transit signal to noise ratio |

## Data Cleaning and Processing

We downloaded the data in .csv format from MAST and the data contains 9564 rows and 50 columns. However upon close observation we found that some of the columns were enirely null as well as some were irrelevant for our purposes hence we dropped some of the columns.

To fill columns having some null values we used two methods, if data was numerical we filled it using the mean which is the average of the column and if the data was categorical we used mode which is the most frequent data in the column.

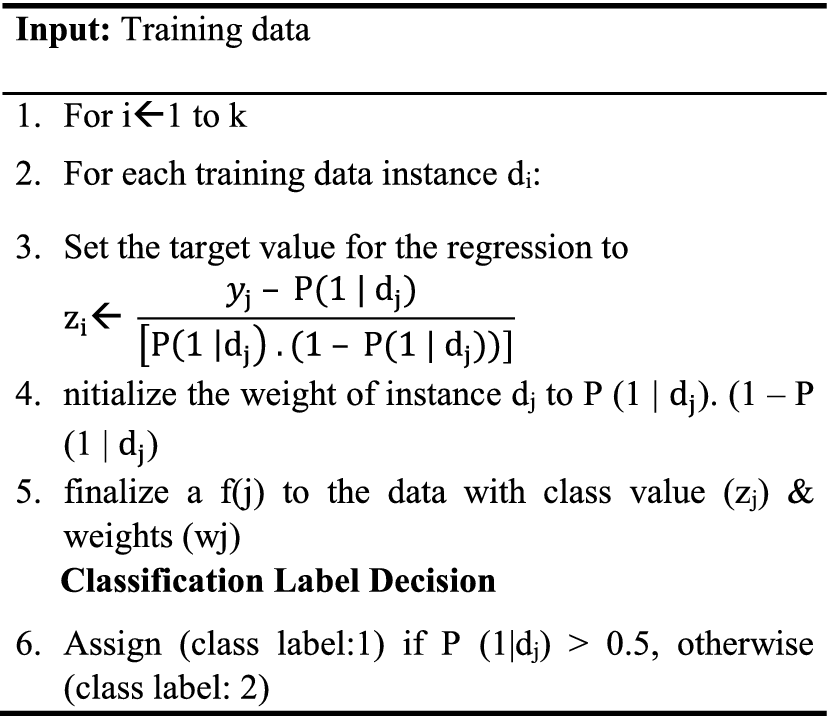
Also we found that the categorical data was nominal in nature hence to convert it from categorical to numerical data we used one-hot encoding technique. Also we slpit the data into training and testing set in the 70-30 fashion i.e. 70% percent data for training the model and 30% remaining for testing. During splitting we set randomness to 1 and set the shuffle flag to True.

Lastly since our data contains values ranging very differently we decided to perform normalization and standardization using MinMaxScaler and StandardScaler respectively so that the data comes down to a more reasonable range. The MinMaxScaler changes all the values to range between 0 and 1 where as StandardScaler ranges down the values between -1 and 1.

## *Model Selection*

For our project we decided to train three different models on three datasets (main data, normalised data and standardised data). The three models are :-

a. Logistic Regression[9] - Logistic regression is a method of modeling the probability of a distinct outcome given associate input variable. the foremost common logistic regression models a binary outcome; something that may take 2 values like true/false, yes/no, and so on. Multinomial logistic regression can model eventualities wherever there are more than 2 potential distinct outcomes. logistic regression could be a helpful analysis technique for classification issues, wherever you are attempting to work out if a brand new sample fits best into a class. As aspects of cyber security are classification issues, like attack detection, logistic regression could be a helpful analytic technique.

Figure 6: Logistic Regression Algorithm

b. Decision Tree Classifier[9] - Decision Tree is a supervised learning technique that may be used for both Classification and Regression issues, however largely it is most popular for solving Classification issues. It's a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the choice rules and every leaf node represents the end result. In a decision tree, there are two nodes, that are the Decision Node and Leaf Node. Decision nodes are accustomed to create any decision and have multiple branches, whereas Leaf nodes are the output of these choices and don't contain any further branches. The decisions or the test are performed on the idea of features of the given dataset. It is a graphical illustration for obtaining all the potential solutions to a problem/decision supported given conditions. It is known as a decision tree because, just like a tree, it starts with the root node, that expands on more branches and constructs a tree-like structure.

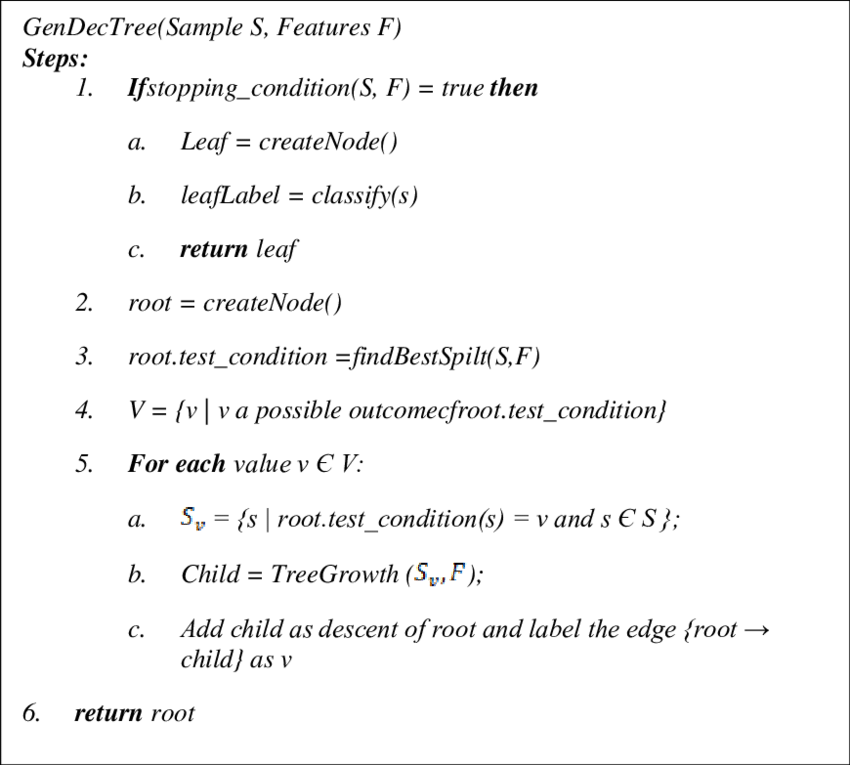
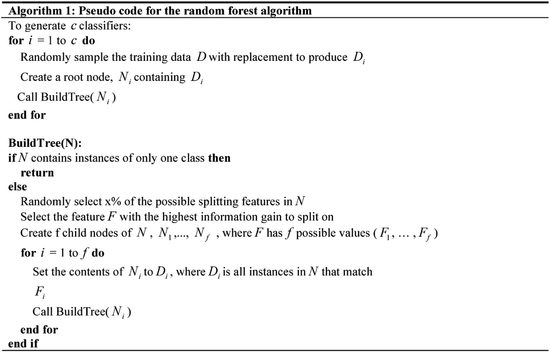


Figure 7: Decision Tree Algorithm

c. Random Forest Classifier[9] - Random Forest is a supervised machine learning algorithm in which multiple Decision Tree Model work together as a committee. Every individual tree within the random forest spits out a {class|a category} prediction and also the class with the most votes becomes our model’s prediction.The fundamental idea behind random forest is a straightforward however powerful one — the knowledge of crowds. In data science speak, the rationale that the random forest model works so well is: A large variety of comparatively unrelated models (trees) operating as a committee can exceed any of the individual constituent models.

Figure 8: Random Forest Algorithm

*D. Evaluation*

After training our models with data we tested them using our test data set and evaluated Confusion Metrix, Accuracy, Precision, Recall and F1 Score.

The test results of all are three models is given below

Table 2: Result for Logistic Regression Model

|  |  |  |
| --- | --- | --- |
| **Dataset** | **Accuracy** | **F1 Score** |
| Main | 79.634% | .7975 |
| Normalized | 79.540% | .7968 |
| Standardized | 79.751% | .81673 |

Table 3: Result for Decision Tree Model

|  |  |  |
| --- | --- | --- |
| **Dataset** | **Accuracy** | **F1 Score** |
| Main | 75.657% | .7665 |
| Normalized | 75.978% | .7708 |
| Standardized | 76.596% | .7751 |

Table 4: Results for Random Forest Model

|  |  |  |
| --- | --- | --- |
| **Dataset** | **Accuracy** | **F1 Score** |
| Main | 79.887% | .8054 |
| Normalized | 80.985% | .8114 |
| Standardized | 82.318% | .8326 |

V. Conclusion And Future Scope

From our results we can say that in our case random forest classifier works the best with accuracy of 82% and F1 Score of .82. Also we have choosen the classical machine learning approach instead of deep learning method like Convolutional Neural Network (CNN) because classical algorithms are easy to implement and far more less complex and require less time as well space to train. They also require less data as compared to deep learning algorithms. Also our score were comparable to the results of other algorithms we have discussed in this paper. Our understanding for exoplanets have come a long long way since the discovery of first exoplanet in 1995. In early stages the only extra solar planets found were big gas giants and hot Jupiters but now with the advancement in our technology and sophisticated satellites we are able to detect many Earth like exoplanets as well. The data has been pouring in since then and has helped to improve our understanding of our solar system and it's creation. Also with the lauch of James Webb Telescope there will be enormous increase in the amount of data that we are currently receiving hence we plan on improving our system to predict the data from additional satellite like TESS and also improve the overall accuracy of the model.

Acknowledgement

We are grateful to our Institution for guiding us to use this opportunity. We would also like to thank our guide Mr. Anurag Malik and our Project Committee for the continuous guidance provided by them.

We, as a team, thank each other for all the co-operation and support required to complete this paper. All the authors whose papers were used for reference for our paper were a great source of information.

##### References

1. Christopher J. Shallue, Andrew Vanderburg, “Identifying Exoplanets with Deep Learning ”
2. Abhishek Malik, Benjamin P. Moster, Christian Obermeier, “Exoplanet Detection using Machine learning “
3. Shafi, “Exoplanet Detection using Artificial intelligence ”.
4. <https://manyworlds.space/tag/direct-imaging/>
5. <https://blog.planethunters.org/2019/05/17/the-exoplanet-hunt-the-radial-velocity-method/>
6. <https://nexsci.caltech.edu/workshop/2011/>
7. https://www.researchgate.net/figure/A-light-curve-showing-the-transit-method-of-detecting-exoplanets\_fig1\_312524607
8. https://exoplanets.nasa.gov/news/1688/discovery-alert-a-giant-planet-and-three-eclipses/
9. https://www.sas.com/en\_gb/insights/articles/analytics/machine-learning-algorithms.html#:~:text=There%20are%20four%20types%20of,%2Dsupervised%2C%20unsupervised%20and%20reinforcement.