HIGH SCHOOL EDITION
Journal of Student Research

Using Machine Learning to Determine the Most Important Features in Exoplanet Verification

Ved Srivathsa[1] and Rida Assaf[#]

[1]The International School Bangalore, India
[#]Advisor

ABSTRACT

Over a decade ago, NASA launched the Kepler Space Telescope in order to find earth-like planets revolving around sun-like stars in the hopes of finding habitable exoplanets. The Kepler pipeline picked up data for over 9000 astronomical bodies, out of which 52% were determined to be false positives, while the remaining 48% were candidates to be classified as exoplanets. The data collected from this mission can be used to assess and automatically classify Kepler Objects of Interest (KOIs) as exoplanets or false positives. Our goal in this work is to determine if some data features are more important than others in classifying an object as an exoplanet. To this end, we built 5 Machine Learning classification models (namely, logistic regression, support vector classifier, gradient boosting classifier, random forest classifier, and multilayer perceptron) and used 15 features to train and test them. We have included Machine Learning models that are explainable to help attain our goal, Our best predictor (random forests) achieved a prediction accuracy of 99% when evaluated with k-fold cross-validation. We evaluated the feature importances of our model and found that 5 of the features (Not Transit-like Flag, Centroid Offset Flag, Stellar Eclipse Flag, Ephemeris Match Indicate Contamination Flag, and Planetary Radius) out of the 15 selected ones make up roughly 75% of the overall feature importances. We hope that our findings can guide the selection of appropriate data to accurately predict exoplanet candidacy for future missions.

Introduction

Many theories exist surrounding the possibility of life on other planets. The hunt for knowledge in these areas has inspired many space missions in the pursuit of answering difficult scientific questions. For instance, on March 7th, 2009, NASA launched the Kepler Space Telescope to find Earth-sized planets in the goldilocks zone (habitable zone) orbiting other stars [1][2]. Kepler recorded data for more than 9000 such exoplanets. Of these, roughly 52% have been confirmed to be false positives, while the remaining 48% remain candidates. Data about the Kepler Objects of Interest (KOIs) collected in the mission are available on the NASA Exoplanet Archive as well as on Kaggle [3][4]. The 52% false-positive statistic is a testament to the knowledge gap that exists in accurately vetting out these exoplanets. With the substantial amount of data available, Machine Learning is a natural field to resort to in order to perform statistical analysis and answer some of the open questions in this field.

There have been many attempts by scientists to solve similar problems in the past. In the paper 'Exoplanet validation with machine learning: 50 new validated Kepler planets' [5], a similar dataset of Threshold Crossing Events (TCEs) [6] is used to identify TCEs that are not Astrophysical False-Positives (AFPs) or Non-Transiting Phenomena (NTPs) using a Gaussian Process Classifier, Random forests, Extra trees, and a Multilayer Perceptron.

Moreover, the paper 'Identifying Exoplanets with Deep Learning: A Five-planet Resonant Chain around Kepler-80 and an Eighth Planet around Kepler-90' [7], shows the use of a one-dimensional Convolutional

Neural Network (CNN) with max-pooling to determine whether a planet is a candidate or false positive based on light curves produced by the Kepler pipeline available on the Mikulski Archive for Space Telescopes [8].

Another paper, 'A Machine Learning Technique To Identify Transit Shaped Signals' [9], uses the Q1 - Q17 DR24 catalog [10] to remove non-transiting signals while retaining known planet candidates. It uses dimensionality reduction and the k-nearest neighbors model to condense data and create a metric that will enable future missions to easily find the best planetary candidates.

The paper 'Exoplanet Detection using Machine Learning' [11] makes use of Kepler light curves data by using the analysis library TFresh to extract features from the curves. It uses a gradient-boosting classifier to then classify exoplanets. Similarly, the paper 'Automatic Classification of Kepler Planetary Transit Candidates' [12] also uses light curve data and classifies exoplanets using a random forest algorithm.

In order to help better understand the use of photometry, the paper 'Kepler Mission Design, Realized Photometric Performance, and Early Science' [13], provides more extensive insight into the use of transit photometry methods to classify exoplanets.

Many scientists are still tackling the problem of classifying planets as candidates or false positives. Our goal in this work is different: we aim to evaluate the most important features required to classify exoplanets and the performance of different Machine Learning models on the unique chosen dataset to find the most effective method to classify KOIs.

Methods

Data

The master dataset we base our work on is from NASA's Exoplanet Archive [4], which is a comprehensive database consisting of data regarding various KOIs. We used a subset of the archive posted as a dataset on Kaggle [3]. The features in this dataset include project disposition columns (comprising of important flags such as the centroid offset and stellar eclipse flags), planet transit properties (including the orbital periods, transit epochs, etc.), and stellar parameters (such as the stellar effective temperatures and stellar surface gravities). After dropping all missing values, the data we use consists of 9201 instances of KOIs each represented by 15 features.

We used 5- and 10-fold cross-validation to assess our models' performance and shuffled the data before splitting it into batches.

Models

We trained several machine learning algorithms using the Python module scikit-learn (sklearn) [14].

All hyperparameters reported below were found by applying Sklearn's GridSearchCV algorithm with 10-fold cross-validation to maximize model performance.

The first model we trained was a regular Logistic Regression model [15]. This is a simple model that uses a sigmoid function to classify data into their respective groups. It uses a gradient descent algorithm to minimize the cost function and reduce the error in the model. We set the maximum number of iterations to 10,000 (to ensure that the model converges) and set the inverse of the regularization strength to 8.5. The remainder of the hyperparameters were left as their default values.

The second model we trained was an SVM (support vector machine) classifier. A support vector machine is an algorithm that attempts to find a hyperplane to separate data into two different sets. The position of the best possible hyperplane is determined by finding the hyperplane that has the greatest margin (distance) from the nearest data points from both sets [16]. We specified the kernel to be an rbf kernel which performed better compared to the sigmoid kernel. We also set the inverse of the regularization strength to 100. All other hyperparameters were set as the default values.

The third model we trained was a gradient-boosting classifier model. This algorithm sets target outcomes for the next model to minimize the error of the model. These target outcomes are calculated based on how much changing the prediction impacts the total error of the model [17]. We set the learning rate to 0.05 and left the remaining hyperparameters as their default values.

The fourth model we trained was a random forest classifier. A decision tree model is one in which an output is generated based on a series of comparisons of the inputs of the model against threshold values. A random forest essentially builds several decision trees and combines them to get a more accurate prediction [18]. We set the number of trees in the forest to 1000, and all remaining hyperparameters were left as their default values.

The fifth and final model we trained was a multilayer perceptron (MLP) classifier. This is a neural network model that consists of several hidden layers each with a specific number of neurons. Input neurons feed inputs into the network. Every neuron is connected to every neuron in the subsequent layer, and each connection is assigned a weight (which represents the strength of the connection between two neurons) [19]. Our model used a logistic activation function with a maximum number of iterations set to 2000. We also set the learning rate to 0.00005 and had a total number of 800 neurons each in 6 hidden layers. All other hyperparameters were left as their default values.

Results

Each of the models performed differently, showing us which algorithm was the most effective at classifying KOIs. Using K-fold cross-validation [20], data was split into 5 and 10 batches, and the average score from each of the splits was calculated. Table 1 shows the 5-fold and 10-fold average accuracies of each of the machine learning models.

Table 1. The average accuracy for 5 Machine Learning models using 5- and 10-fold cross-validation.

Machine Learning Model	5-fold CV accuracy	10-fold CV accuracy
Logistic Regression	78.4%	78.2%
Support Vector Classifier	66.8%	66.8%
Gradient Boosting Classifier	98.9%	99.0%
Random Forest Classifier	**99.0%**	**99.0%**
Multilayer Perceptron Classifier	98.4%	98.4%

Overall, the random forest classifier and gradient boosting classifier models had the best performances with roughly 99.0% accuracy. The advantage of a random forest classifier is that it is highly intuitive in understanding the importance of features used to train the algorithm, creating greater scope for explainable Artificial Intelligence [21]. In order to better understand how the models worked, we looked into the feature importance's for the random forest classifier to see which features were of the greatest significance when accurately classifying KOIs. Figure 1 shows the percentage of importance each feature had.

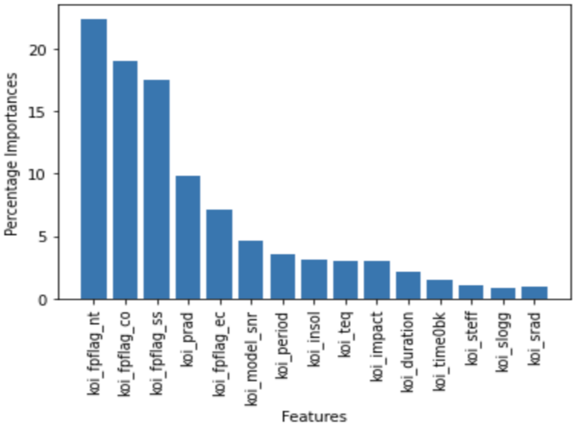

Figure 1. A breakdown of the feature importance's of the model expressed by the percentage importance of each feature

Conclusion

We experimented with 5 machine learning models and identified the most important features to collect in order to effectively predict whether or not a KOI is an exoplanet candidate, in the hopes that this can guide future data collection for similar missions.

While the Kepler space telescope was deactivated on November 15th, 2018, NASA's quest to find habitable exoplanets has not ended. On April 18th, 2018, NASA launched the Transiting Exoplanet Surveying Satellite (TESS) mission to search for exoplanets in an area 400 times larger than that surveyed by Kepler [23]. In the future, our models could be modified to classify and accurately vet out false positives for the TESS mission and could be used to extend our search for habitable exoplanets.

Acknowledgments

This research has made use of the NASA Exoplanet Archive, which is operated by the California Institute of Technology, under contract with the National Aeronautics and Space Administration under the Exoplanet Exploration Program.

References

1 - Goldilocks Zone. (2021, March 4). Exoplanet Exploration: Planets Beyond Our Solar System. https://exoplanets.nasa.gov/resources/323/goldilocks-zone/

2 - Koch, D. G., Borucki, W., Dunham, E., Geary, J., Gilliland, R., Jenkins, J., ... & Weiss, M. (2004, October). Overview and status of the Kepler Mission. In Optical, Infrared, and Millimeter Space Telescopes (Vol. 5487, pp. 1491-1500). International Society for Optics and Photonics.

3 - Kepler Exoplanet Search Results. (2017, October 10). [Dataset]. https://www.kaggle.com/nasa/kepler-exoplanet-search-results

4 - Kepler Objects of Interest. (2017–2018). [Dataset]. https://doi.org/10.26133/NEA4

5 - Armstrong, D. J., Gamper, J., & Damoulas, T. (2021). Exoplanet validation with machine learning: 50 new validated Kepler planets. Monthly Notices of the Royal Astronomical Society, 504(4), 5327-5344

6 - Q1-Q17 DR25 TCE. (2017–2018). [Dataset]. https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=tce

7 - Shallue, C. J., & Vanderburg, A. (2018). Identifying exoplanets with deep learning: A five-planet resonant chain around kepler-80 and an eighth planet around kepler-90. The Astronomical Journal, 155(2), 94.
8 - Home. (n.d.). MAST. https://archive.stsci.edu/
9 - Thompson, S. E., Mullally, F., Coughlin, J., Christiansen, J. L., Henze, C. E., Haas, M. R., & Burke, C. J. (2015). A machine learning technique to identify transit shaped signals. The Astrophysical Journal, 812(1), 46.
10 - Kepler Objects of Interest (KOI) Activity Tables. (n.d.). NASA. https://exoplanetarchive.ipac.caltech.edu/docs/Q1Q17-DR24-KOIcompanionV5.html
11 - Malik, A., Moster, B. P., & Obermeier, C. (2020). Exoplanet Detection using Machine Learning. arXiv preprint arXiv:2011.14135.
12 - McCauliff, S. D., Jenkins, J. M., Catanzarite, J., Burke, C. J., Coughlin, J. L., Twicken, J. D., ... & Cote, M. (2015). Automatic classification of Kepler planetary transit candidates. The Astrophysical Journal, 806(1), 6.
13 - Koch, D. G., Borucki, W. J., Basri, G., Batalha, N. M., Brown, T. M., Caldwell, D., ... & Wu, H. (2010). Kepler mission design, realized photometric performance, and early science. The Astrophysical Journal Letters, 713(2), L79.
14 - scikit-learn: machine learning in Python — scikit-learn 1.0 documentation. (n.d.). Scikit-Learn. https://scikit-learn.org/stable/
15 - Z. (2021, September 26). Logistic Regression Explained - Towards Data Science. Medium. https://towardsdatascience.com/logistic-regression-explained-9ee73cede081
16 - Support Vector Machines: A Simple Explanation. (n.d.). KDnuggets. https://www.kdnuggets.com/2016/07/support-vector-machines-simple-explanation.html
17 - Hoare, J. (2020, December 8). Gradient Boosting Explained - The Coolest Kid on The Machine Learning Block. Displayr. https://www.displayr.com/gradient-boosting-the-coolest-kid-on-the-machine-learning-block/
18 - Donges, N. (2021, September 17). A Complete Guide to the Random Forest Algorithm. Built In. https://builtin.com/data-science/random-forest-algorithm
19 - Multilayer Perceptron - an overview | ScienceDirect Topics. (n.d.). Multilayer Perceptron. https://www.sciencedirect.com/topics/veterinary-science-and-veterinary-medicine/multilayer-perceptron
20 -Brownlee, J. (2020, August 2). A Gentle Introduction to k-fold Cross-Validation. Machine Learning Mastery. https://machinelearningmastery.com/k-fold-cross-validation/
21 - Schmelzer, R. (2019, July 24). Understanding Explainable AI. Forbes. https://www.forbes.com/sites/cognitiveworld/2019/07/23/understanding-explainable-ai/?sh=4937bd697c9e
22 - Data columns in Kepler Objects of Interest Table. (n.d.). NASA Exoplanet Archive. https://exoplanetarchive.ipac.caltech.edu/docs/API_kepcandidate_columns.html
23 - NASA. (n.d.). TESS - Transiting Exoplanet Survey Satellite. https://www.nasa.gov/tess-transiting-exoplanet-survey-satellite/