Identifying Exoplanets from Kepler Light Curves: Milestone 1

Michael Calderin*
University of Florida CAP5771
(Dated: February 21, 2025)

1. INTRODUCTION

Starting in 2009 and continuing for 9.6 years, NASA's Kepler/K2 missions set out to hunt for planets outside of our solar system [1]. A large part of the identification process was to record the flux (brightness) from stars in a small patch of our galaxy and detect when there is a dip in the flux. The dips are typically signs of a planet crossing the star. For a planet, these transits are periodic and can be fit to help estimate parameters such as the planet's size, distance from its star, etc. These fitting models also give better quantification for the transit depth (the amount that the flux falls during the transit) and other transit-related features.

However, not all transits are planets. Some stars come in pairs and can also have transits. These are known as eclipsing binaries. There are other false positives such as interference from the light of other stars. For a transit to be confirmed as a planet, there is typically a pipeline that requires additional observations and can take years. There are several planetary candidates that to this day have not been confirmed to be planets. NASA uses Robovetter, a decision tree, to automate the classification process. It distinguishes between candidates and false positives but does not make predictions for true planets. Machine learning is a powerful tool that could potentially compensate for the weaknesses of Robovetter and improve the exoplanet identification process.

2. OBJECTIVE AND TECH STACK

The primary motive will be to classify transits based off the light curves acquired during the Kepler mission and additional contextual data. Depending on the model, features chosen may vary and this will be further solidified during feature selection in the second milestone. As an example, the flux and times for light curves of stars combined with context such as the coordinates of the star could be used. Ideally, the features should have strong predictive power but also be non-trivial and relatively simple to measure in practice. Features with strong predictive power that are difficult or time-consuming to measure would not be beneficial to NASA's pipeline.

To present findings, an interactive conversational agent will be used. Hopefully, this will be visually appealing instead of in a command-line but it will depend on model performance and time constraints. Several models will be developed with current interest in random forests, CNNs, and RNNs. SQLite will be used for the bulk of the storage, meaning the SQLite3 module. Pandas, NumPy, and SciPy will be used for data manipulation, and Matplotlib and Seaborn for visualizations. As for the agent, ChatGPT API, and Rasa are contenders. This is subject to change, especially since the SQL database is having time difficulties for queries due to the size of the data.

3. TIMELINE

February 24, 2025 - March 9, 2025

Data storage will be improved to allow for faster queries. LASSO will reduce the dimensionality so that relevant features can be analyzed more closely. PCA will also be taken into consideration.

March 10, 2025 - March 16, 2025

Features will be selected based on previous analysis. Samples will be split and training/hyper-tuning will begin for about three supervised classification models, iteratively evaluating performance metrics.

March 17, 2025 - March 21, 2025

Final attempts for improvements will be made along with an analysis of the strengths, weaknesses, and biases of each model. Some experimentation may also be useful to see if the average result of the models provides better performance compared to the individual models. Report and GitHub will be updated.

March 22, 2025 - March 30, 2025 Using the test set, run the best models based on previous performance. Draw insights and limitations.

March 31, 2025 - April 13, 2025 Research and develop the conversational agent. Possibly deploy the models as an API and/or SQL database to query from for more reliable responses.

April 14, 2025 - April 23, 2025

Work on the final presentation and submission.

*Electronic address: michaelcalderin@ufl.edu

4. DATA COLLECTION

NASA's exoplanet archive provides a "Cumulative Kepler Objects of Interest (KOI)" table in the form of a CSV which has summary information about each star's transits. This is where the potential exoplanets' dispositions are labeled as confirmed, candidate, or false positive. This was directly downloaded through their website [2] and saved as *KOI_cumulative.csv*. The light curve data was more complicated to fetch since there is 3 TB worth of light curves in their database. There was a section of the archive for bulk downloads that provided a script called *Kepler_KOI_wget.bat* [3]. It contains a *wget* command on each line that fetches light curve data for each star that is in the KOI table and would likely amount to more than 100 GB when fetched in its entirety. Each star's light curve is its own dataset.

In a Jupyter Notebook titled *data_collection.ipynb*, the bat file was processed line by line. Single quotation marks had to be converted to double quotations to be able to run the commands on Windows. The commands were run through Python using its *subprocess* module. The data was saved in an SQLite database titled *light_curves.db* and due to the size of the data, only relevant features were kept and the process was stopped after collecting data for the first 2680 stars which is about a quarter of the stars in the KOI table and roughly 20 GB. Since transits are typically periodic, each star has up to hundreds of transits in their light curves so there should be enough to train from if multiple transits per star are used. It is the current approach and might change if a more efficient storage structure is found. With this sample, SQL queries are already slow. In terms of downloading the data, it took two days to fetch these stars alone. It would have been ideal to use a random sample instead of going based on first come first serve, but due to computational constraints it would take too long to download a random sample at this time. For now, the potential for bias will be noted and tracked throughout the project.

5. DATA CONTENT AND PREPROCESSING

The data was analyzed in *data_processing.ipynb*. The KOI CSV was read in as a Pandas data frame but filtered so that it would only include stars that also appeared in the SQL light curve database. The KOI data had 3164 rows and 141 columns with features such as star ID, transit time, duration, etc. The SQL database had features such as star ID, time of measurement, flux, and quality of measurement. There were six columns and over 200 million rows.

With the exception of a few features, the features were mainly analyzed and not dropped at this stage. Those that were dropped were mainly due to being entirely null, constant, or in the interest of preserving the most useful data while having no null values. Basic information for each feature such as their null count, descriptive statistics, most frequent values, etc. were displayed. There were no duplicates. Some features that obviously needed data types conversions were handled. Pearson correlation coefficients were calculated for numerical features and chi-squared for categorical features. Histograms, frequency bar graphs, and box plots for each feature were saved to a folder. Outliers were detected but not removed at this stage. However, they were excluded for imputation purposes. Numerical features were scaled using the scikit-learn standard scaler. For more information, refer to the Jupyter Notebook which has markdown cells with details and insights.

6. EXPLORATORY INSIGHTS

Due to the large number of features, some key insights will be discussed but they are in no means exhaustive. To begin, Pearson correlations showed three predominant areas of high correlations (above 0.8): star/planet characteristics, equipment information, and errors. Meaning, features related to stellar or planetary data could likely be reduced to a few key features and the same goes for the other two categories. The chi-squared test is shown in Figure 1. Most dependent relationships are contextually obvious. For example, the disposition and false positive flags would be related because one of the classes of disposition is false positive and the flags are simply a more descriptive version of that. This gives strong indication that the categorical features could be condensed. Despite this, no features were dropped since that would be better suited for the feature selection and engineering phase. The main emphasis in this milestone is to understand the data and acquire enough evidence to justify feature selections in the next milestone.

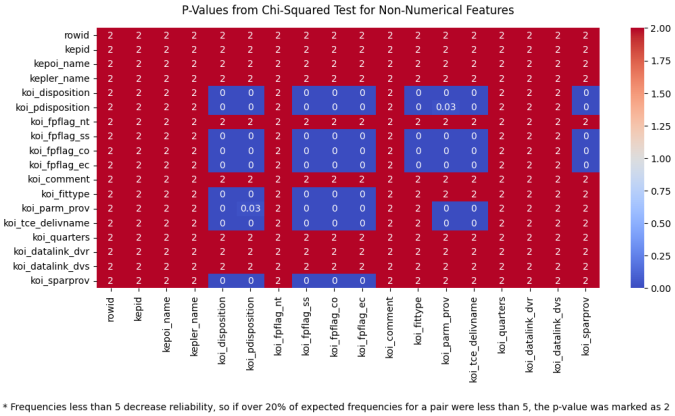

FIG. 1: The p-values after running the chi-squared test on non-numerical features. P-values of 2 are obviously nonphysical and indicate that the chi-squared would be inaccurate for that pair of features due to a small size in expected frequency which decreases reliability according to the scikit-learn documentation.

The target variable is "koi_disposition" and it was important to understand its distribution. It turns out there are about 700 candidates, 1000 confirmed planets, and 1400 false positives. The imbalance indicates that stratification might be useful for modeling. This should be monitored since the distribution might change if multiple transits per star are used during modeling.

In Figure 2, higher scores for candidates and confirmed planets indicate greater confidence in the classification while for false positives, lower scores indicate greater confidence. Across the board, there is high confidence in each disposition. Still, there is a notable imbalance, specifically for candidates, between median and mode. Likely, low-confidence outliers are skewing this category. This could be due to false positives having more obvious patterns and confirmed planets having more rigorous processing in the pipeline, while candidates have less predictable trends and are somewhat in the middle between false positives and confirmed.

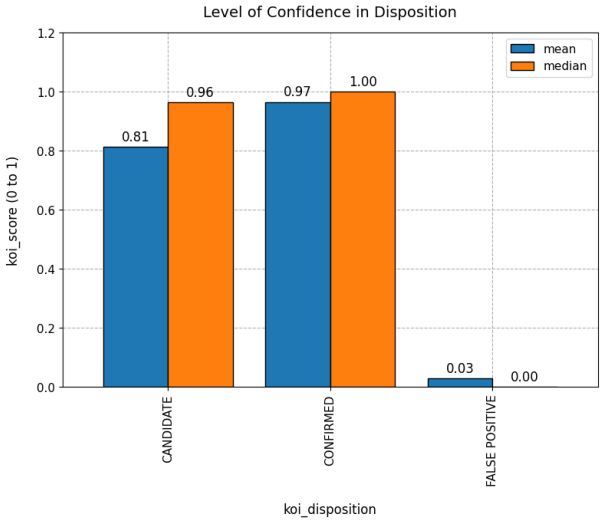

FIG. 2: The mean/median level of confidence in each disposition is displayed. These scores are generated by a Monte Carlo technique such that the score's value is equivalent to the fraction of iterations where NASA's automated classifier (Robovetter) outputs "CANDIDATE".

As shown in Figure 3, the current sample from the Kepler light curve data was the outer edges of the field of view. There are no points from the center. Given that Kepler only captured a small section of the sky and it was our own galaxy, the data is innately biased aside from sampling. It still would have been more representative of the data at hand to do randomized sampling, but the data is ultimately biased regardless and computational constraints prevented "fair" representation. There are also drawbacks to randomized sampling such as not getting enough data from neighboring stars which could lead to a lack of recognition of light interference type false positives. The problem at hand has many complex factors at play so optimized sampling would be a study of its own.

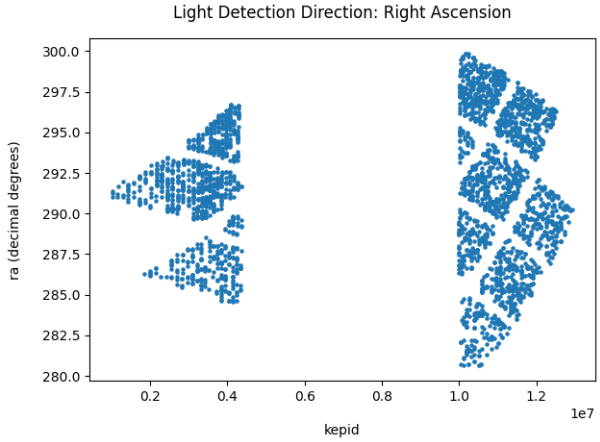

FIG. 3: "kepid" is a unique identifier for each star and is plotted against right ascension. The accompanying coordinate to right ascension, declination, varied from roughly 36 to 52 decimal degrees and was also missing stars in the middle. It provided no new information in terms of sample selection that this right ascension visualization did not encapsulate.

Figure 4 shows that the types of false positives are also imbalanced. Due to the size of the data, it is difficult to find all imbalances but clearly they are present so special attention is needed for this issue, especially as features are narrowed down in the feature selection phase.

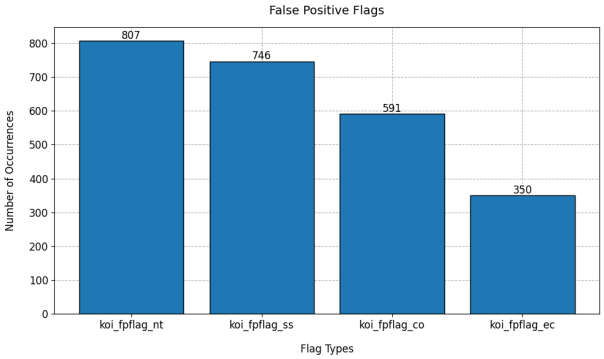

FIG. 4: This is the distribution of false positive flags. "nt" is not transit-like, "ss" is stellar eclipse, "co" is centroid offset (detecting light from a different, nearby star), and "ec" is ephemeris contamination (flux contamination or electrical crosstalk).

In terms of outliers, a democratic method was employed between z-score, inter-quartile range, and median absolute deviation for each feature. Generally, using two-thirds agreement was considering too much of the data as outliers. This is not only inconvenient since training typically requires as much data as possible, but also disconnected from visual insights. Upon inspection, many of the "outliers" are generally part of the cluster of data. Unanimous outlier detection seems to be a better fit. Going forward, values will be considered outliers if there

is unanimous agreement. These outliers will be tracked throughout the project but not discarded quite yet. Discarding even one outlier would mean a large amount of light curve data is thrown out. It would also be strange to remove outliers since transits themselves are rare events compared to the number of data points in a light curve. For more specifics on summary statistics, distributions, etc., refer to the Jupyter Notebook which is documented step-by-step.

---

[1] NASA, *Kepler by the numbers*, https://science.nasa.gov/resource/nasas-kepler-mission-by-the-numbers/ (2018).

[2] NASA, *Kepler objects of interest*, https://exoplanetarchive.ipac.caltech.edu/docs/data.html, cumulative KOI data.

[3] NASA, *Bulk data download*, https://exoplanetarchive.ipac.caltech.edu/bulk_data_download/, kepler KOI time series.