Evaluating Various ML Models to Efficiently Uncover Exoplanets

INTRODUCTION

- Exoplanets are planets located outside of our solar system, and several of them have the potential to support life
- The first two exoplanets found in 1992, Poltergeist and Phobetor, were found orbiting the pulsar PSR B-1257+12
- The NASA mission Kepler/K2 discovered several exoplanets in 2016 using various methods, the most prominent one being the transit method
- Other methods besides transit include radial velocity, gravitational microlensing, direct imaging, and astrometry
- The transit method has aided in the discovery of 4187+ new planets; however, the process of validating exoplanets with transit data is extremely time-consuming and laborious
- Scientists must use 2 different methods or telescopes to validate the existence of an exoplanet, and this often takes months or years
- By expediting the process of exoplanet validation, scientists will be closer to finding more habitable Earth-like exoplanets for future human exploration

TRANSIT METHOD

- A transit occurs when an astronomical body passes in front of its star and an observing telescope
- The star's flux (light intensity) will decrease by a measurable amount because the body is blocking some of the star's light from reaching the telescope

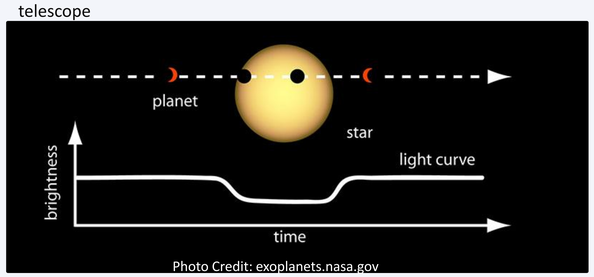
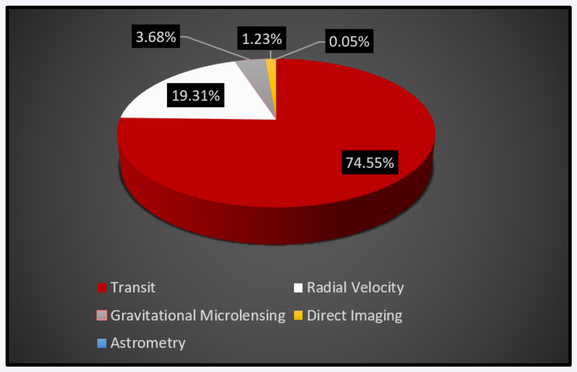


Photo Credit: exoplanets.nasa.gov

RESEARCH OBJECTIVES

- As several exoplanet validation methods take a significant amount of time, efficiency is crucial to narrow down and differentiate exoplanets from other astronomical bodies
- Finding exoplanets quickly could greatly drive human interest in settling habitable exoplanets in the future and discovering life beyond Earth
- The purpose of this study is to train various machine-learning models for accurate and rapid detection of exoplanets based on data obtained using the transit method
- By classifying whether a star is an exoplanet or not, scientists can further use the information from these machine-learning models to predict the characteristics of each planet to determine their usefulness and habitability

Percentage of Methods Used in Exoplanet Discovery



MATERIALS AND METHODS

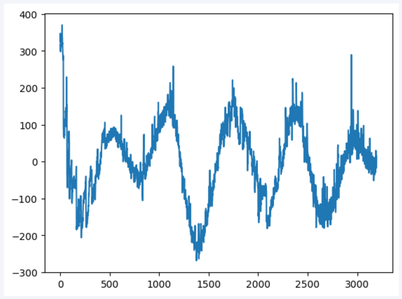
- The experiment was conducted in Google Colaboratory, a Jupyter Notebook virtual environment that provides free access to computing resources
- Python libraries such as Pandas, NumPy, Matplotlib, SkLearn, and TensorFlow were utilized for this study

DATASET

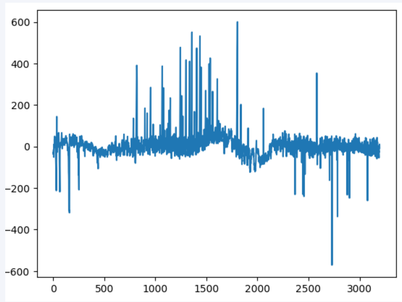
- The dataset used for this experiment was a publicly available Exoplanet Hunting in Deep Space dataset, compiled from the Kepler Space Telescope Mission
- This dataset contains 5087 stars with a celestial object either orbiting them or passing near the star
- Each star has 3197 flux (light intensity) values measured over a specified interval through the transit method
- Each light intensity value was measured in intervals and recorded in the dataset
- Based on these light intensity values, a star is categorized as either having an exoplanet orbiting it (labeled as 1) or not having an exoplanet orbiting it (labeled 0)
- Examples of stars with orbiting exoplanets made up <1% of the dataset, which would have caused bias in the models, so preprocessing with SMOTE was performed to equalize the number of examples for stars with orbiting exoplanets and stars without orbiting exoplanets
- The data was split using a 90/10 ratio into train and test sets

EXPLORATORY DATA ANALYSIS

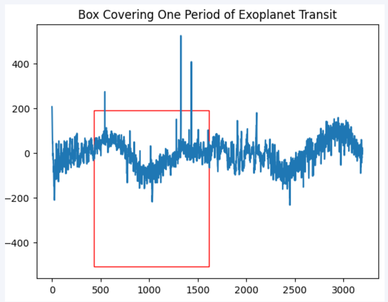
Exoplanet Light Intensity Curve



Non-Exoplanet Light Intensity Curve



One Period (Cycle) of Light Intensity During Exoplanet Transit

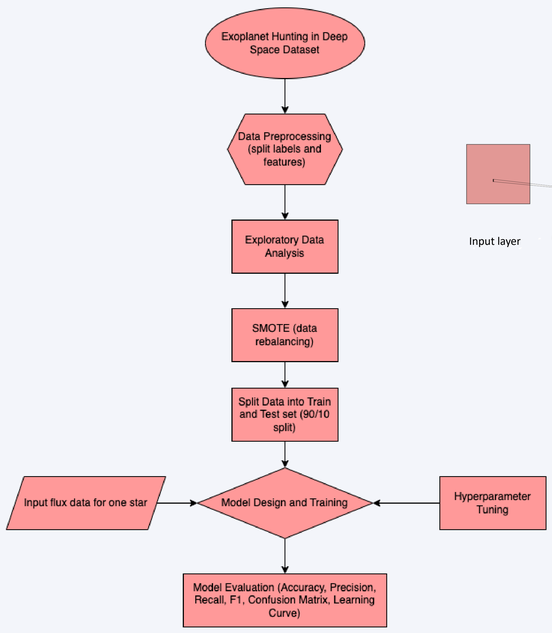


X-axis: Measurement Number, Y-axis: Flux (Light Intensity) Measurement

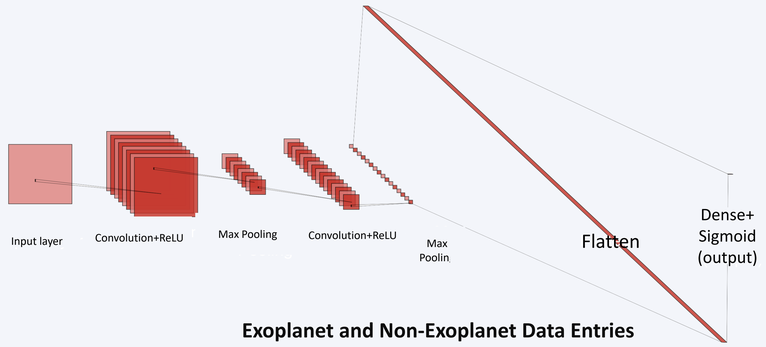
PROCEDURE

- Preprocessing was done to prepare the data for input into the machine learning models, by splitting the features and labels, running SMOTE on the data, and dividing the data into train and test sets
- Six machine learning models were trained and evaluated, including Convolutional Neural Networks, K-Nearest Neighbors, Multi-Layer Perceptron, Logistic Regression, Decision Tree, and a Neural Network
- Hyperparameter tuning was performed using GridSearchCV to improve the accuracy of the model after initial training and evaluation
- We measured each model's effectiveness using several metrics such as accuracy, precision, recall, F1 score, ROC-AUC, confusion matrices, and neural network learning curves

Workflow



CNN Architecture and Layers



Exoplanet and Non-Exoplanet Data Entries

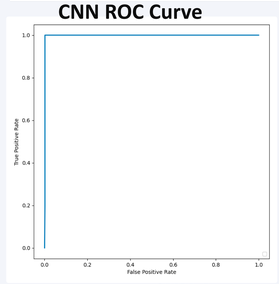
LABEL	FLUX.1	FLUX.2	FLUX.3	FLUX.4	FLUX.5	FLUX.6	FLUX.7	FLUX.8	FLUX.9	...
1	93.85	83.81	20.10	-26.98	-39.56	-124.71	-135.18	-96.27	-79.89	...
1	-38.88	-33.83	-58.54	-40.09	-79.31	-72.81	-86.55	-85.33	-83.97	...
1	532.64	535.92	513.73	496.92	456.45	466.00	464.50	486.39	436.56	...
1	326.52	347.39	302.35	298.13	317.74	312.70	322.33	311.31	312.42	...
1	-1107.21	-1112.59	-1118.95	-1095.10	-1057.55	-1034.48	-998.34	-1022.71	-989.57	...
...	...	...	...	...	...	...	...	...	...	...
0	-91.91	-92.97	-78.76	-97.33	-68.00	-68.24	-75.48	-49.25	-30.92	...
0	989.75	891.01	908.53	851.83	755.11	615.78	595.77	458.87	492.84	...
0	273.39	278.00	261.73	236.99	280.73	264.90	252.92	254.88	237.60	...
0	3.82	2.09	-3.29	-2.88	1.66	-0.75	3.85	-0.03	3.28	...
0	323.28	306.36	293.16	287.67	249.89	218.30	188.86	178.93	118.93	...

RESULTS

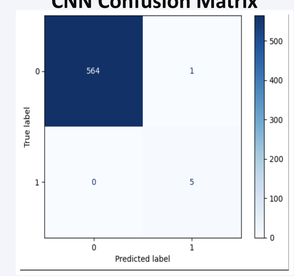
- The classification results were depicted on a confusion matrix
- Precision measures the total number of correctly identified exoplanets from the total amount of examples identified as exoplanets
- Recall measures the total number of correctly identified exoplanets from the total number of exoplanet samples in the test set
- The F1 score is the harmonic mean of precision and recall
- AUC is the Area under the ROC curve, which shows various classification thresholds that show a model's tradeoff between true positives and false positives
- An AUC closer to 1 shows the model can maximize true positives while minimizing false positives
- Accuracy and Loss Curves were used to track the neural network's performance over each training iteration to combat overfitting and underfitting
- These models maximized testing accuracy while minimizing testing loss

MODEL	Accuracy	Precision	Recall	F1 Score	AUC
K Nearest Neighbors	89.64%	0.05	0.6	0.09	0.84
Logistic Regression	98.59%	0.33	0.6	0.42	0.77
Decision Tree	98.94%	0.4	0.4	0.4	0.69
Convolutional Neural Network	99.82%	0.83	1	0.9	0.99
Neural Network	98.77%	0.37	0.6	0.46	0.74
Multi-Layer Perceptron	98.59%	0.33	0.6	0.42	0.92

CNN ROC Curve



CNN Confusion Matrix



CNN Accuracy and Loss Curves

CONCLUSION AND IMPACT

- After training and evaluating my models, the CNN had the greatest accuracy of 99.82%, as well as high scores in other metrics
- With the data that will be available from several space telescopes currently active in observing planets far beyond our galaxy, this CNN offers a user-friendly, reliable, and inexpensive way of sifting through this data in just a few seconds
- By speeding up detection with this CNN, scientists can use this tool to find exoplanets, so they can be studied further for their important characteristics
- This model can easily be accessible in both professional and non-professional settings and applied by astronomers, amateur scientists, and space enthusiasts alike to contribute to the discovery of more exoplanets

FUTURE STEPS

- I plan to incorporate data from other currently active telescopes and satellites, such as TESS, to re-train and test my models
- I would like to add data about the characteristics of the exoplanets so that this tool can be used to detect exoplanets as well as identify key features which will determine their habitability

Graphs/Charts created by student researcher