Draft version February 16, 2025
Typeset using LaTeX **modern** style in AASTeX631

Identifying Exoplanet Candidates with Neural Networks and Machine Learning

Vladimir Bautista[1] and Dimna Prado[1]

[1]*San Diego State University, 5500 Campanile Drive, San Diego, CA 92182-1221, USA*

ABSTRACT

In this project, we developed a machine learning model to classify light curve data, determining whether it may potentially indicate the presence of an exoplanet. By leveraging stochastic gradient descent (Ruder 2016) and the Adam (Kingma & Ba 2014) optimizer, we trained and optimized our model to analyze patterns in the light curves. The model achieved a high accuracy of 93.41% when tested on a synthetic dataset closely resembling the formatting of the training data. However, performance declined to 58.33% on raw light curve data from the NASA Exoplanet Archive and further to 50.00% on pre-processed light curves from the TESS Pipeline. These results emphasize the sensitivity of the model's success to the formatting of input images, showing the importance of consistency in data preparation for effective classification in exoplanet detection.

1. INTRODUCTION

1.1. *Detecting New Worlds Beyond Our Solar System*

Prior to 1990, our knowledge of planets in the universe extended only to those in our solar system. Then in 1995, Swiss astronomers Michel Mayor and Didier Queloz discovered the first exoplanet – 51 Pegasi b, (Mayor & Queloz 1995) orbiting a Sun-like star. Exoplanets are planets located outside our solar system, orbiting distant stars. This opened up a whole new field in astronomy. Since then, thousands of exoplanets have been discovered using various detection methods, such as the transit method and the radial velocity method. The current tally for exoplanets detected as of this writing is 5811 (NASA Exoplanet Archive). New exoplanets are frequently being discovered.

One of the most widely used techniques for detecting exoplanets is through light curve analysis, particularly using the transit method. This technique involves observing the periodic dimming of a star's light caused by a planet passing in front of it. Astronomers continuously monitor the brightness of a star over time using either ground-based or space-based telescopes, producing a light curve. A light curve is a plot showing the star's brightness as a function of time. An example light curve

is shown in Figure 1. When an exoplanet crosses in front of its host star (relative to the observer's line of sight), it blocks a small fraction of the star's light, causing a temporary dip in brightness. This dip appears as a characteristic "U-shaped" or "V-shaped" feature in the light curve, depending on the planet's size and the star's limb darkening. Some key parameters can be extracted by analyzing the light curve. The depth of the dip indicates the size of the planet relative to the star. A deeper dip suggests a larger planet. The duration of the transit reflects the time it takes for the planet to cross the star, providing information about the planet's orbital distance and speed. By observing multiple transits, we can determine the orbital period of the exoplanet. Also, deviations in the timing of transits (i.e., Transit Timing Variations) can indicate the presence of additional planets in the system, influencing the orbit of the transiting planet.

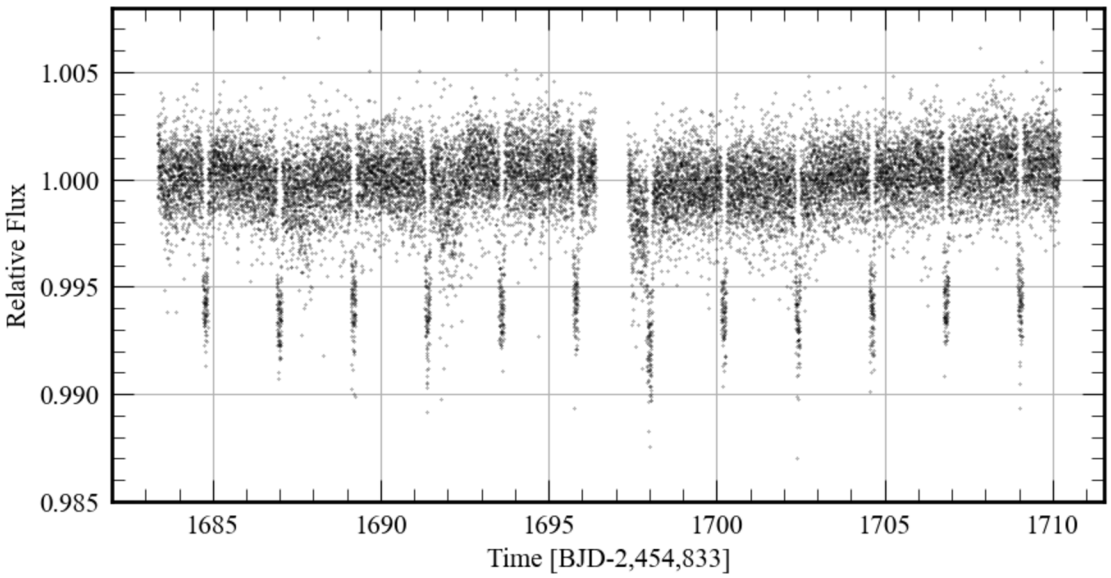

Figure 1. Light curve of the exoplanet HAT-P-7b observed by the *TESS* mission. The plot displays the star's relative brightness as a function of time, highlighting periodic dips corresponding to transits of the exoplanet across the star's disk. HAT-P-7b is a hot Jupiter, a class of gas giant exoplanets that are inferred to be physically similar to Jupiter but have short orbital periods ($P < 10$ days). The small amplitude of the brightness dips, within 1% of the total stellar brightness, underscores the need for highly precise photometric measurements to detect transiting exoplanets.

1.2. *Machine Learning Meets Astronomy*

Stochastic Gradient Descent (SGD) (Ruder 2016) is a widely used optimization algorithm in machine learning that efficiently updates model parameters to minimize the loss function. Unlike traditional gradient descent, which computes gradients using the entire dataset, SGD updates the parameters iteratively using small, random batches of the data. During each step, the parameters were updated using

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla L(\theta_t), \tag{1}$$

where θ_t represents the parameters at iteration t, η is the learning rate, and $\nabla L(\theta_t)$ is the gradient of the loss function L with respect to θ_t. This stochastic approach makes SGD notably effective for large datasets, as it reduces computational costs while introducing randomness that helps escape local minima and improve generalization (Theodoridis 2015). Momentum (Qian 1999) γ could be incorporated to Eq. 1 by modifying the formula to include a velocity term, v_t, which accumulates past gradients. The updated formula becomes

$$\begin{aligned} v_{t+1} &= \gamma v_t + \eta \nabla L(\theta_t) \\ \theta_{t+1} &= \theta_t - v_{t+1} \end{aligned} \tag{2}$$

where v_t is the velocity at iteration t, which combines the current gradient and a fraction of the previous velocity, and γ is the momentum coefficient, controlling the influence of the past velocity. The velocity term, v_t, replaces the direct gradient, allowing momentum to smooth out fluctuations and accelerate convergence by maintaining consistent movement in the direction of the optimal solution. Momentum accelerates convergence by smoothing out the optimization path. It achieves this by incorporating a fraction of the previous update into the current step, allowing the model to build velocity in consistent directions of the loss surface while reducing oscillations in less stable regions. This helps the optimizer move more efficiently toward the optimal solution, especially in scenarios where gradients vary significantly.

In this project, SGD plays a critical role in training a binary classification model to identify whether a system contains an exoplanet based on light curve data. By analyzing patterns in the data, such as transit dips or noise characteristics, SGD helps the model learn to differentiate between systems that are likely to host an exoplanet and those that are not. Leveraging the flexibility and efficiency of SGD, we can handle diverse noise patterns in synthetic and real light curves, enabling the model to converge quickly and adaptively optimize its performance. By using labeled datasets from missions like *Kepler* and *Transiting Exoplanet Survey Satellite (TESS)*, SGD trains a classification model by iteratively adjusting its parameters to minimize the error between predicted and actual labels. SGD excels at processing large and noisy datasets efficiently by using small, random batches of data to compute parameter updates, allowing the model to learn critical patterns, such as transit signals, while generalizing well to unseen data. Its stochastic nature helps the model avoid overfitting to irrelevant noise, making it particularly effective for distinguishing exoplanetary signals from non-exoplanetary variability. Once a star system passes the classifier as a possible exoplanet host, the next step involves confirming and characterizing the exoplanet candidate through additional observations and analyses.

Convolutional Neural Networks (CNNs) (LeCun and Bengio 1995) are a type of neural network designed to work with grid-like data, such as images. In light curve

classification approach, CNNs help identify patterns in the light curve images, such as dips that might indicate an exoplanet. Through layers of convolution, the network extracts features from the data while filtering out unnecessary noise. The network learns features from data through convolutional layers, where filters are applied to identify patterns such as edges, shapes, and textures. The primary objective is to optimize the CNN's parameters to minimize the error defined by a loss function, thereby improving the network's predictive accuracy. Each CNN applied the discrete convolution operation within its layers, which can be described as

$$f(x_i) = \sum_j k(j) \cdot g(x_i - j), \tag{3}$$

where $f(x_i)$ represents the output feature at position i, $k(j)$ is the filter kernel, and $g(x_i - j)$ is the input data. This operation allowed the networks to extract spatial features efficiently, reducing computational complexity compared to traditional fully connected layers. For continuous data, the convolution can be mathematically expressed as

$$f(x) = \int_{\mathbb{R}^d} k(s) \cdot g(x - s)\, ds, \tag{4}$$

capturing smooth transitions in the data.

We present our data in Section §2, outline the methods used to build our model in Section §3, and provide the results in Section §4. In Section §5, we discuss the model's performance during training and testing, explore the limitations of our approach, and propose potential directions for future work.

2. DATA

To create a robust training and validation dataset, we generated synthetic light curve data that mimics the transit signals of exoplanets while incorporating realistic noise profiles. We use synthetic data to maintain control over key parameters influencing the shape of the light curves, such as transit depth, duration, and noise types. This approach enabled us to create diverse and well-labeled datasets, reducing the ambiguities and uncertainties commonly associated with real-world data. Using real data for training and validation comes with several challenges. Real light curve data often contains noise and artifacts from instruments or environmental factors, which can make it difficult to isolate clean transit signals. Additionally, labels in real-world datasets can sometimes be incomplete or uncertain, especially for low signal-to-noise cases. There is also often an imbalance in the data, with far fewer examples of exoplanet signals compared to non-transiting systems. These factors can complicate the training process, making synthetic data a more practical choice for this project.

Each light curve (Figure 2) represents the brightness variations of a star over time, with periodic "U-shaped" dips to simulate the transit of an exoplanet. These dips are characterized by the parameters that define the planet's physical and orbital

properties. To emulate the observational challenges of real-world data, we added different types of noise—Gaussian, white, and red noise—to the light curves. Gaussian noise introduces random fluctuations that follow a normal distribution, representing thermal or electronic noise in detectors. It creates uncorrelated variations across the light curve, mimicking common short-term noise sources. White noise also introduces random variations but has equal power across all frequencies, representing uniform background noise that can mask transit signals. Finally, red noise, which is frequency-dependent, simulates long-term correlated variations often observed in astrophysical data, such as stellar activity or instrumental drift.

To simulate false transits in star systems without exoplanets, we generated synthetic light curves featuring occasional random dips and added noise (Figure 3). The dips were introduced based on a set probability at each time point, with each dip taking on a parabolic "U-shape" to resemble a transit. The depths and durations of the dips were randomized within specified ranges to introduce variability, capturing the diversity of potential false-positive signals that might occur in observational data. To ensure the training data focuses solely on the light curve patterns, we exclude text headers, axes labels, and other annotations from the images. These elements do not provide any meaningful information for the classification task and would instead contribute unnecessary noise, potentially hindering the model's ability to learn the key features of the light curves. We utilized a total of 4,449 synthetic images, split into 70% for training, 15% for validation, and 15% for testing. The odd number of images results from the server's capacity to generate large data batches without crashing. This dataset size is sufficient to meet the objectives of this study.

The real data used in the testing phase was sourced from the NASA Exoplanet Archive, a comprehensive repository that hosts a vast collection of confirmed exoplanet data. This archive serves as a gateway to additional resources, such as the Mikulski Archive for Space Telescopes (MAST), which provides access to raw observational data. MAST includes datasets from major space missions like Hubble, James Webb, TESS, and the upcoming Roman Space Telescope. For this project, we specifically extracted 12 raw light curve data from MAST to serve as a set of test data to see how our model performs on realistic and unprocessed light curve data (Figure 4). Each dataset we used included a data validation report summary, which contains images of processed light curves along with key analyses produced by the TESS Science Processing Operations Center (SPOC) Pipeline (Jenkins et al. 2016) at NASA Ames Research Center (Figure 5). These processed light curve images undergo rigorous corrections and optimizations to remove instrumental artifacts and enhance signal clarity, making them highly reliable for analysis. We employed these 12 processed light curve images as an additional batch of test data to evaluate how well our model performs on well-processed, clean signals, with highly different formatting. This allowed us to assess the model's robustness and accuracy when applied to data that closely resembles the outputs from professional processing pipelines.

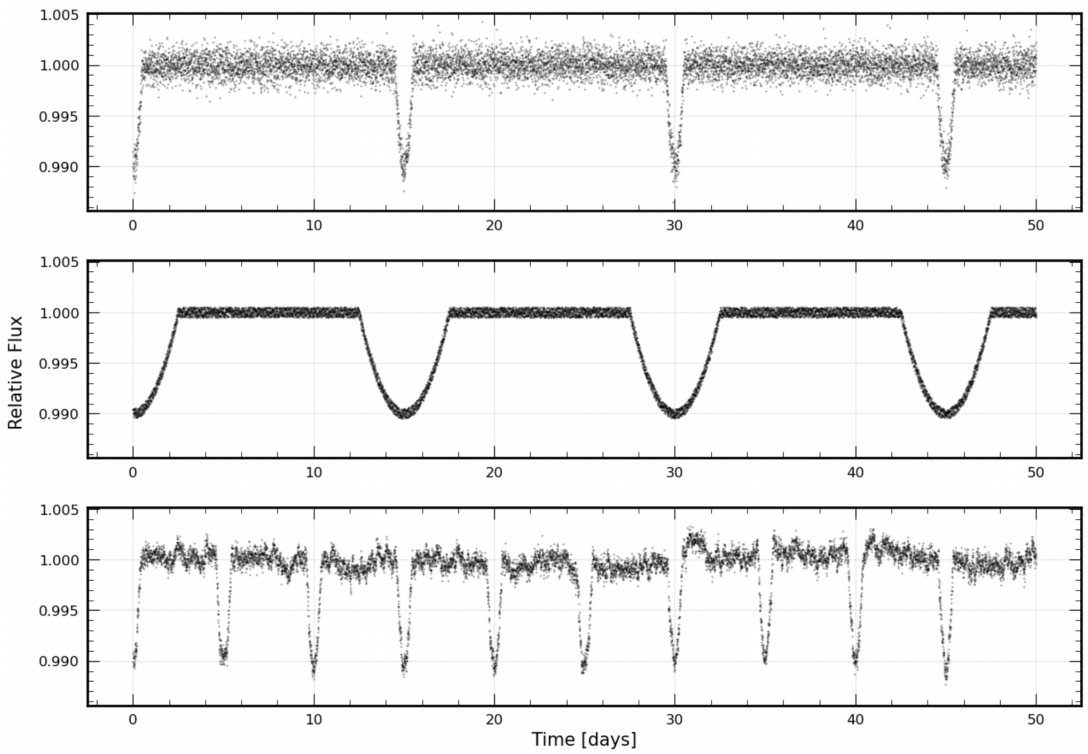

Figure 2. Synthetic light curves generated with different types of noise. The *top panel* illustrates a light curve with Gaussian noise, characterized by normally distributed fluctuations around the baseline flux. The *middle panel* shows a light curve with white noise, featuring uniformly distributed random variations. The *bottom panel* depicts a light curve with red noise, exhibiting correlated, long-term variability that mimics real-world astrophysical noise sources. Each synthetic light curve demonstrates the characteristic transit "U-shape" dips caused by an exoplanet.

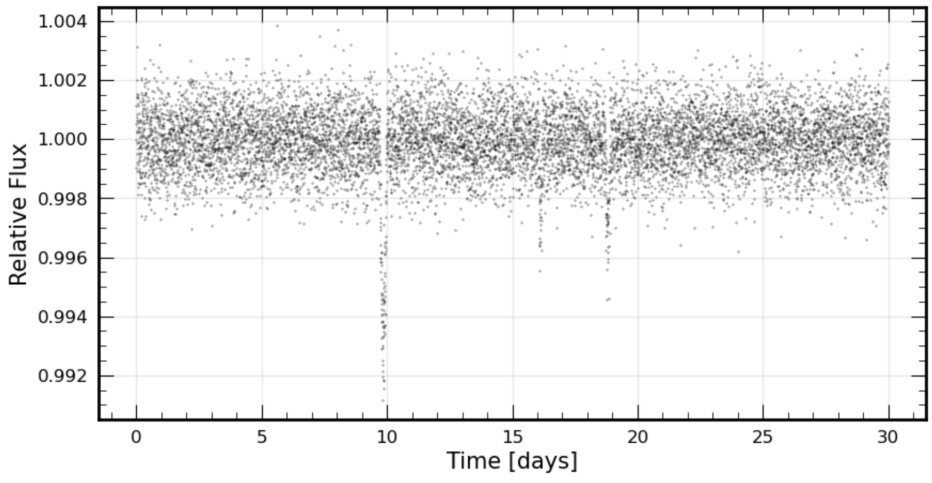

Figure 3. Synthetic light curve generated with Gaussian noise, simulating a system with no exoplanet. The light curve features false transits caused by random noise fluctuations, resulting in dips that do not correspond to periodic planetary transits.

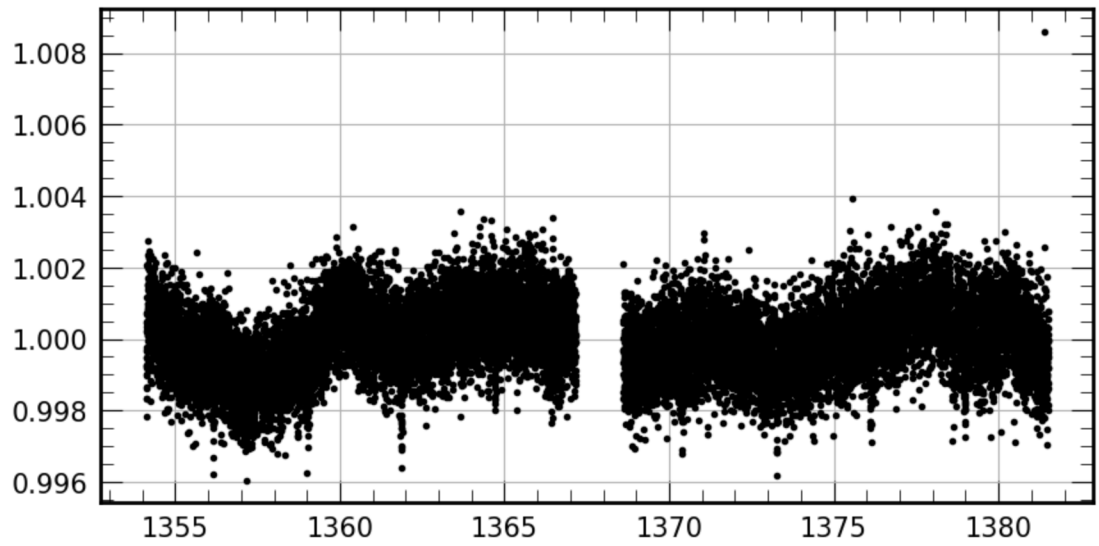

Figure 4. Example of raw light curve data of the system GJ 238 extracted from the MAST Archive. The plot reveals significant noise and data scatter, along with gaps indicating missing observations. Such raw data require extensive preprocessing to mitigate noise and extract reliable transit signals for exoplanet detection.

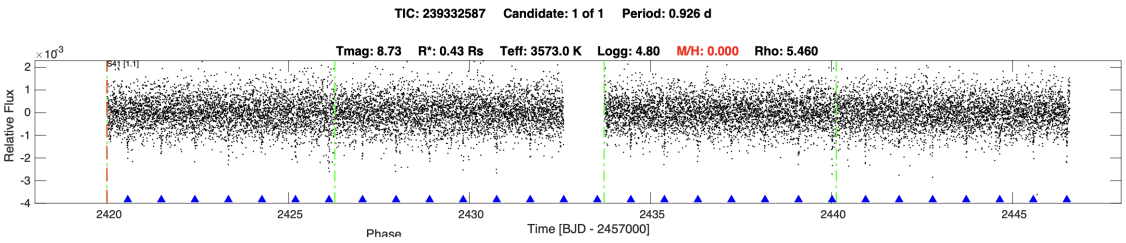

Figure 5. Processed light curve for the system TIC 239332587, generated by the TESS SPOC Pipeline. The system contains a confirmed exoplanet with an orbital period of 0.926 days. The plot shows the relative flux as a function of time (BJD - 2457000) and phase, with clear periodic dips indicating planetary transits. Key stellar parameters such as the stellar radius ($R^* = 0.43\,R_\odot$), effective temperature ($T_{\rm eff} = 3573\,{\rm K}$), and surface gravity (Logg = 4.80) are provided for context.

3. METHODS

All images were converted to grayscale, reducing the color channels while preserving the essential intensity patterns, which allowed the network to focus on structural features critical for classification. The images were then resized to a uniform dimension of 128×128 pixels, ensuring compatibility across the dataset. Due to the inherent temporal sequence of light curves, we avoided augmenting the data by flipping or cropping the images to prevent disrupting their order. Following this, the images were transformed into tensors. Neural networks require numerical input in a structured format to perform computations. Tensors, which are multi-dimensional arrays, are the standard input format for deep learning frameworks like PyTorch and TensorFlow. By transforming images into tensors, we enabled the neural network to process the light curve data efficiently. The dataset was divided into three subsets: 70% for training, 15% for validation, and 15% for testing.

Our machine learning architecture is built as a sequential convolutional neural network optimized for light curve classification. The model consists of four convolutional blocks, each designed to progressively extract and refine features from the input images. The first block looks for patterns in the input images using 32 filters, each focusing on a small 3x3 area of the image at a time. After detecting these patterns, we use a Rectified Linear Unit (ReLU) (Agarap 2018) activation function to keep only the positive values, making the model more efficient at learning. Next, max pooling simplifies the data by keeping only the most important features while reducing the image size, making the model faster and less complex. Finally, we apply dropout, which randomly ignores some of the model's connections during training, helping prevent it from memorizing the training data and improving its ability to work with new data. Subsequent blocks expand the feature extraction capacity by increasing the number of filters to 64, 128, and 256, while retaining similar layers of ReLU activation, max pooling, and dropout in the earlier blocks. The final convolutional block uses a large pooling size to further compress spatial dimensions before flattening the output into a one-dimensional vector, linking feature extraction to classification. A fully connected layer is dynamically sized based on the inferred output dimensions. These layers analyze the flattened features to make predictions on whether an exoplanet is present. The final layer applies a sigmoid function

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

to output probabilities for binary classification.

Binary Cross-Entropy Loss (BCELoss) was used to optimize the four CNN blocks for binary classification tasks. The function follows the formula

$$\text{BCELoss} = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right], \tag{6}$$

where y_i is the true label, $\hat{y}_i$ is the predicted probability, and N is the batch size. This loss function evaluates how well our model's predictions align with the actual labels, penalizing incorrect predictions more heavily as the predicted probability deviates from the true label, which provided a consistent measure of model performance throughout training and validation. In addition, we utilized *Adam* (Kingma & Ba 2014), an algorithm for first-order gradient-based optimization of stochastic objective functions, with a learning rate of 1×10^{-3} to help adjusts the learning rates for each parameter to help the model converge faster and more effectively. Its ability to handle noisy gradients in the light curve data ensures more stable updates to the model parameters.

During the model training, batch sizes of 32 input data and labels were processed. Predictions were generated, and the BCELoss function calculated the error between the predictions and true labels. Using backpropagation, the gradients were computed, and the Adam optimizer updated the model's parameters to minimize the loss. Backpropagation calculates how much each parameter in the model contributed to the prediction error by propagating the error backwards through the network. These gradients indicate the direction and magnitude of change needed for each parameter to reduce the error. The Adam optimizer then uses these gradients to adjust the parameters efficiently. This process helps the model learn the underlying patterns in the data and generalize to unseen data.

During validation phase, predictions were made without updating weights, providing an independent measure of the model's performance on unseen data. Loss and accuracy metrics were computed for validation data to monitor overfitting or underfitting trends. Both loss and accuracy metrics were normalized for each epoch to ensure that these metrics are calculated as averages across all validation samples, regardless of batch size. This approach helps avoid misinterpretations caused by variations in dataset size or batch processing.

Figure 6 illustrates the training and validation performance of the model over 30 epochs. The training loss and validation loss start at similar high values and remain nearly constant for the first 10 epochs, indicating no significant learning during the initial phase. However, around epoch 10, a sharp decrease in both training and validation loss occurs, converging to lower values by epoch 20 and stabilizing thereafter. This behavior suggests the model effectively learned to minimize the loss, reaching a point of stability. The training accuracy and validation accuracy also exhibit a significant rise around epoch 10, increasing rapidly to above 90% by epoch 15 and stabilizing with minimal fluctuations. The close alignment of training and validation accuracy, as well as the loss values, implies that the model generalizes well without signs of overfitting or underfitting. Overall, the results indicate that the model successfully learned to classify the light curves with high precision while maintaining consistency between the training and validation sets.

Introducing momentum into the training process caused the model to struggle with convergence, as evident by the fluctuations in both training and validation loss across the epochs seen in Figure 7. Unlike the steady decline seen without momentum, the loss values exhibit significant variability, preventing the model from stabilizing. This behavior arises because momentum amplifies gradient updates, which, while beneficial for speeding up training in some cases, can cause the model to overshoot the optimal solution. As a result, the training accuracy oscillates without steady improvement, and the validation accuracy remains stagnant, indicating that the model fails to learn effectively under these conditions. We experimented with various combinations of learning rates η and momentum γ, testing learning rates of 0.001, 0.0001, and 0.00005, alongside momentum values ranging from 0.1 to 0.9. The model's inability to successfully learn when momentum was introduced could be attributed to the improper combinations of the learning rate and momentum values.

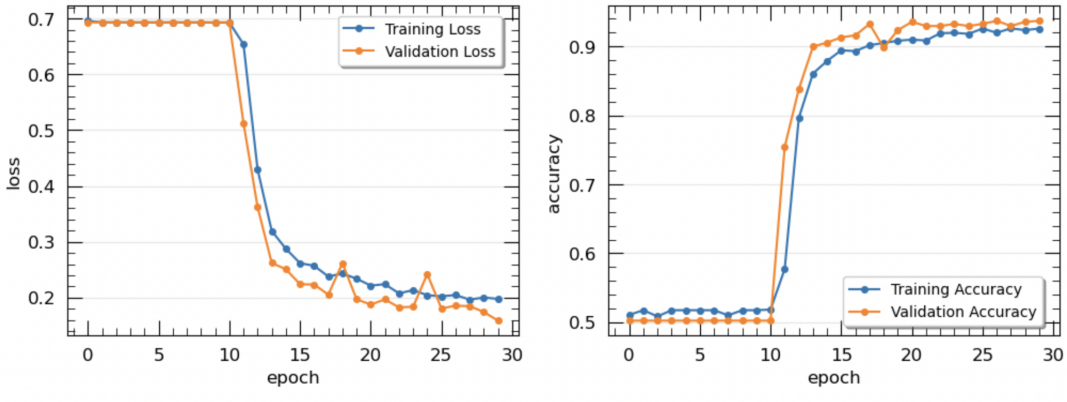

Figure 6. Training and validation performance of the model with a learning rate of 0.001 and no momentum. The left panel shows the loss values for both training and validation sets, indicating a consistent decrease over epochs. The right panel shows the accuracy for both sets, with a sharp rise in performance around epoch 10, followed by convergence. This demonstrates the model's ability to effectively learn patterns in the data without overfitting or underfitting.

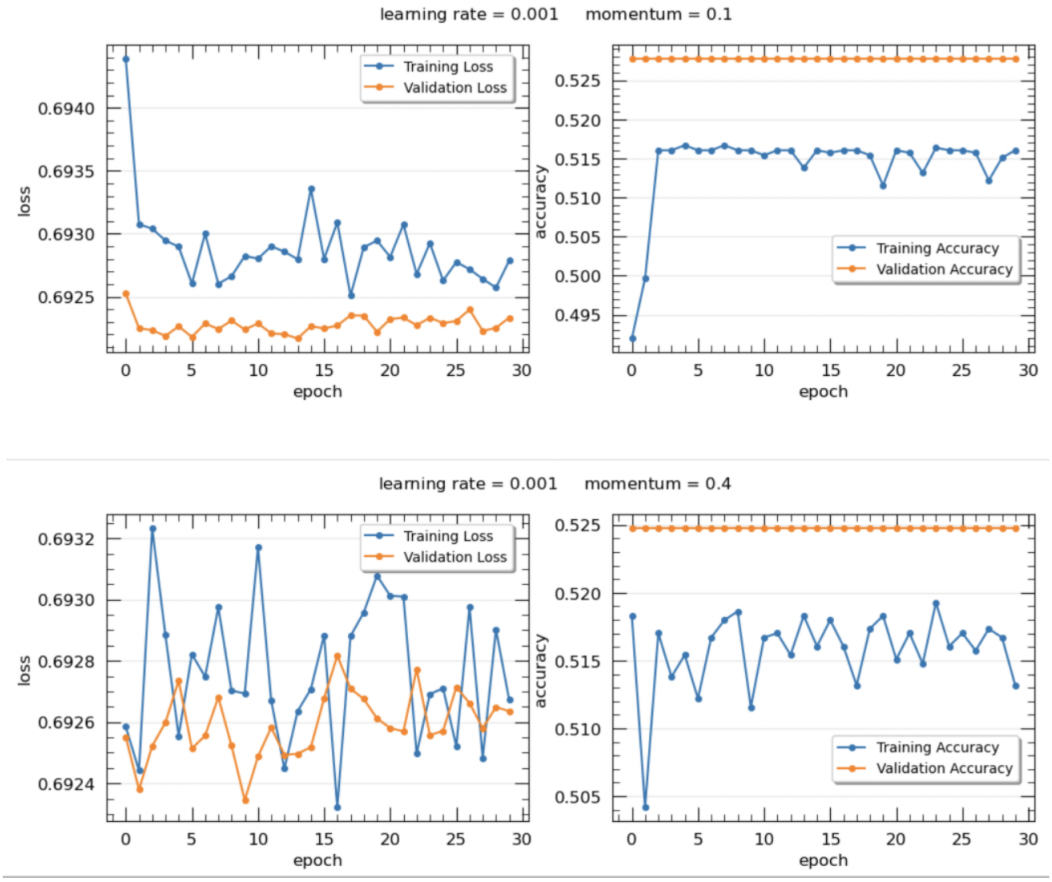

Figure 7. Training and validation loss (*left*) and accuracy (*right*) for models trained with learning rate 0.001 and different momentum values: 0.1 (*top*) and 0.4 (*bottom*). The addition of momentum introduced instability, with higher momentum causing more pronounced oscillations in training loss and accuracy, indicating challenges in convergence due to gradient overshooting. Validation metrics remained relatively stable, suggesting the model's difficulty in learning effectively with higher momentum.

4. RESULTS

The model achieved a high accuracy of 93.41% on the first batch of 668 test samples that are closely matching the training data's formatting. Figure 8 highlights its strong performance, with most predictions correctly identifying exoplanet or non-exoplanet cases. The consistent formatting likely contributed to the success, though a few misclassifications suggest room for improvement in handling borderline cases. On the second batch of test data consisting of 12 raw light curve data from the NASA Exoplanet Archive (Figure 9), our model achieved a final accuracy of 58.33%. The lower performance can be attributed to the raw data's inherent challenges, such as gaps in observations, high levels of noise, and inconsistencies in formatting compared to the training data. These factors likely disrupted the model's ability to recognize patterns effectively. On the third test dataset, consisting of pre-processed images generated by the TESS SPOC Pipeline, the model achieved an accuracy of 50.00%,

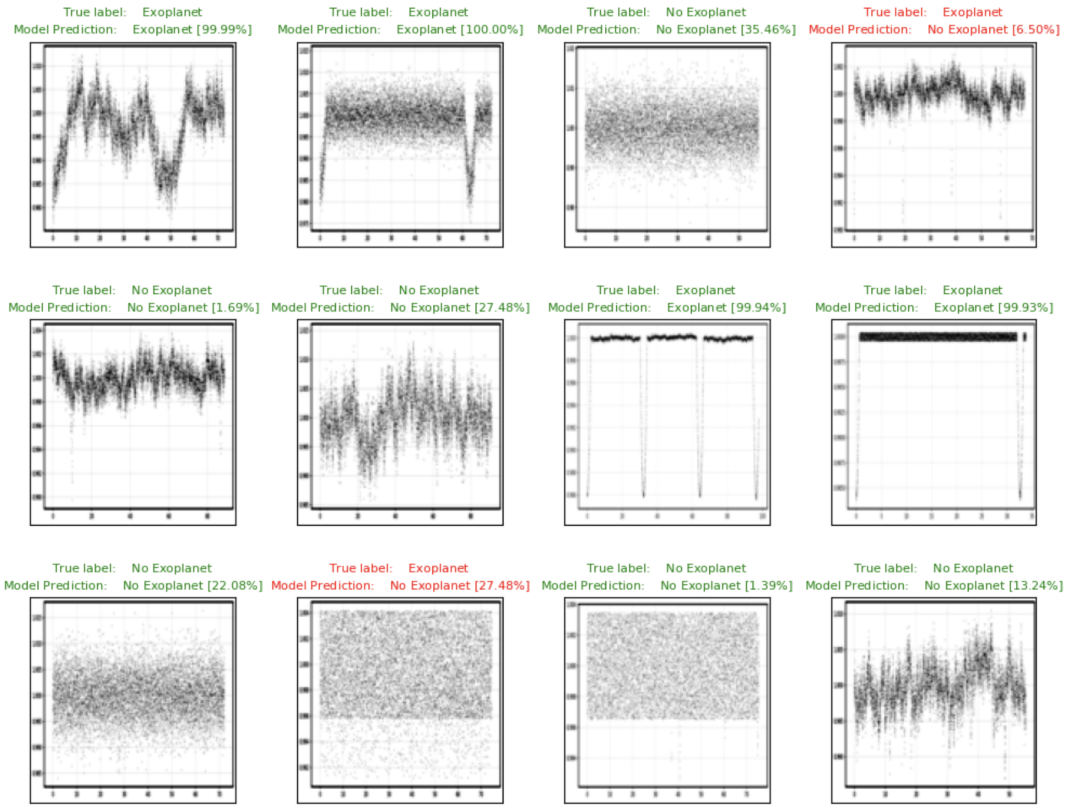

Figure 8. Examples of the model's predictions on the first batch of test data, formatted similarly to the training set. True labels and model predictions are shown above each light curve. The model demonstrates strong performance, with most predictions matching the true labels and high confidence scores. Misclassifications are marked in red, highlighting occasional errors.

correctly identifying 6 out of 12 light curve samples (Figure 10). This dataset differs significantly from the training data and the first two test sets due to its distinct formatting, as the pre-processed images include annotations, axes, and other features not present in the training data. These differences likely contributed to the model's reduced performance, as it struggled to adapt to the unfamiliar data representation.

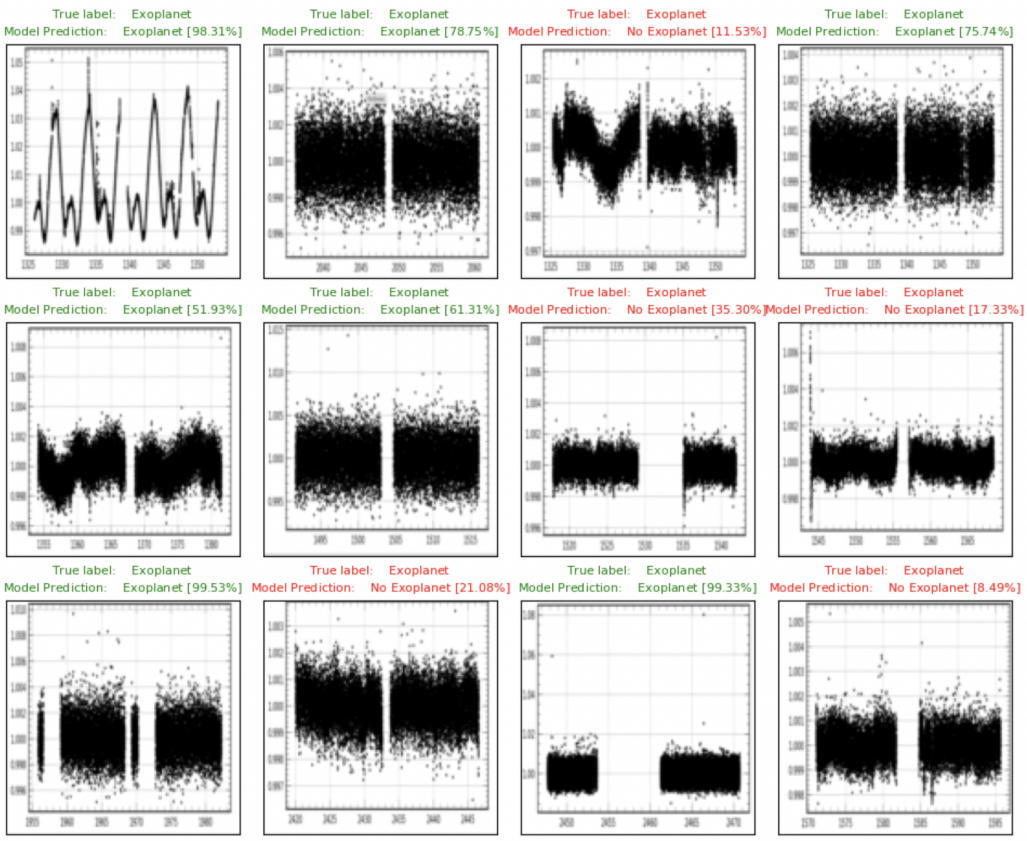

Figure 9. Model predictions on the second test dataset, consisting of raw light curve data from the NASA Exoplanet Archive. This dataset features inherent challenges such as observational gaps, noise, and formatting inconsistencies. The results highlight the model's varying performance, with correct predictions in some cases but significant misclassifications in others, reflecting the difficulty of handling raw, unprocessed data.

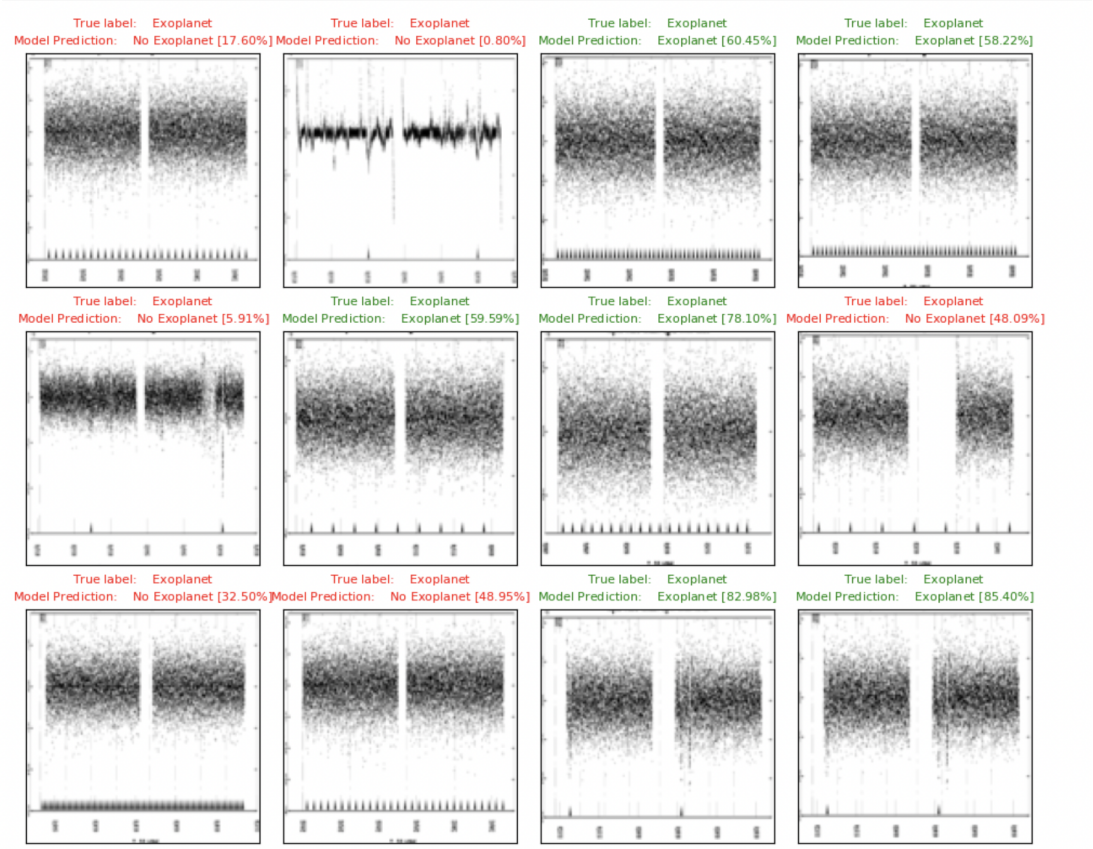

Figure 10. Sample predictions from the model tested on preprocessed light curve data generated by the TESS SPOC Pipeline. Unlike the first two batches of test data, this dataset features significant formatting differences, including annotated axes and labels, which were not present in the training data. These variations in formatting posed a challenge for the model, contributing to its reduced accuracy of 50% on this dataset.

5. DISCUSSION

The training results with a learning rate of 1×10^{-3} and no momentum indicate steady progress in learning (Figure 6). Both training and validation losses decrease significantly, particularly around epoch 10, where the model learns rapidly. The training and validation accuracies increase consistently, leveling off at around 90%, which suggests that the model is performing well on both seen and unseen data. The similarity in trends for training and validation metrics indicates that the model is generalizing effectively, with no clear signs of overfitting or instability. When momentum was introduced to the training process with values between 0.1 to 0.9, it disrupted the model's ability to converge effectively. Momentum aims to accelerate the learning process by incorporating a fraction of the previous gradient's direction into the current gradient update, which can help traverse plateaus in the loss surface. However, in this case, the momentum caused the gradient updates to overshoot, leading to fluctuations in both training loss and accuracy, as evident in Figure 7. These oscillations prevented the model from stabilizing around the optimal solution, resulting in a failure to achieve consistent improvement in performance.

Our model achieved strong performance on the synthetic test data, which closely resembled the training data in both formatting and structure. However, its accuracy dropped when tested on raw data from the NASA Exoplanet Archive. This decrease can be attributed to the unprocessed nature of the raw data, which often includes noise, gaps in observations, and irregularities that were not present in the synthetic training set. These inconsistencies make it harder for the model to generalize effectively, as it was trained on data with consistent formatting. The model's performance further dropped when evaluated on data processed by the TESS SPOC Pipeline. While this dataset is more refined, the formatting and structure differ significantly from the synthetic training data. These differences introduce features that the model has not been trained to handle, leading to difficulty in recognizing patterns it would otherwise detect in familiar data.

5.1. *Limits of Applicability to Complex Datasets*

One of the key limitations of our project lies in the sensitivity of the model's success to the formatting of the test data. While our model performed well on synthetic data, its accuracy dropped significantly when tested on raw or differently formatted datasets. This demonstrates the model's reliance on the consistency of the data format it was trained on. Additionally, representing light curve data as images introduces further challenges, as the model's performance can be influenced by how the data is distributed along the time axis. Variations in time-axis scaling or spacing can affect how features, such as dips or noise, are perceived by the model. These factors showcases the limitations of applying the model to diverse real-world datasets and suggest that further preprocessing or normalization may be necessary for broader applicability.

Real-world astronomical data, such as light curves, often contain significant noise, gaps, and artifacts that make them challenging to use directly for analysis. They require extensive preprocessing, including cleaning, detrending, and normalization, to make them suitable for machine learning models. Additionally, acquiring real data can be time-consuming, as it involves accessing archives, downloading large datasets, and understanding their unique formats and characteristics, which vary depending on the source and instrument.

This approach is specifically tailored for analyzing exoplanets detectable via the transit method, as it relies solely on light curve data showing periodic dips in brightness when a planet passes in front of its host star. As a result, it does not account for exoplanets that could be detected using other methods, such as the radial velocity method, microlensing, and timing variations, inevitably leaving out a significant number of exoplanets. However, this is particularly useful as a first step in exoplanet detection. By analyzing light curve data, it provides an efficient way to identify potential candidates from large datasets, such as those collected by missions like Kepler and TESS. Using machine learning, the model can process thousands of light curves quickly, helping to narrow down promising signals for further analysis. This reduces the workload for astronomers by filtering out systems that are unlikely to host exoplanets, allowing follow-up efforts to focus on the most likely candidates.

5.2. *Future Works*

Future work for this study could focus on improving its adaptability to diverse datasets and extending its application beyond light curve analysis. One area of improvement is developing preprocessing techniques or augmentations to better align the model's training data with the varying formats of real light curves, such as those processed by different pipelines or containing gaps and irregularities. Another potential direction is incorporating multi-method detection strategies, where this approach serves as an initial filter before integrating data from other techniques like radial velocity or microlensing for more comprehensive exoplanet identification. Additionally, the model could be enhanced to handle multi-planet systems by recognizing overlapping transit signals or detecting non-periodic events like eccentric orbits. Finally, expanding the training dataset with more realistic noise models and leveraging larger, high-quality archives, such as future datasets from the James Webb Space Telescope or Roman Space Telescope, could further improve the robustness and accuracy of this method in real-world applications.

APPENDIX

.1. *Fostering Inclusivity Through Open Access*

Projects that rely on proprietary tools, specific archives, or advanced computational techniques can create challenges for researchers who may not have access to these resources. Proprietary tools often require costly licenses, and restricted archives may limit participation to certain institutions. Advanced computational methods may also require specialized hardware or expertise that not all researchers have the opportunity to use or develop. These barriers can make it harder for some researchers to contribute fully, particularly those from underfunded institutions or regions. Ensuring that methods, code, and results are openly accessible helps address these challenges by lowering entry barriers and making it easier for a wider range of individuals to participate. Open access promotes collaboration and brings diverse perspectives to the field, ultimately benefiting the scientific community as a whole. The data used in this project, sourced from the NASA Exoplanet Archive, MAST, and the TESS SPOC Pipeline, are all open access and publicly available. These repositories are designed to ensure transparency and inclusivity by providing researchers, educators, and the general public with unrestricted access to high-quality astronomical data. By making these datasets openly accessible, these organizations promote collaboration across diverse institutions and enable contributions from individuals and groups regardless of geographic or economic barriers. This commitment to open access fosters a more inclusive environment for scientific discovery, allowing a wider range of perspectives to advance our understanding of exoplanets and the broader universe.

Astronomy also offers incredible opportunities for learning and exploration, and public websites like those hosted by NASA play a vital role in making this field accessible to everyone. These open-access platforms provide a wealth of educational resources, including data archives, interactive tools, and detailed information about missions and discoveries. By offering free, high-quality content, they foster curiosity and learning in astronomy for students, educators, and enthusiasts alike. Similarly, the machine learning community benefits from open-access platforms such as TensorFlow, PyTorch, and Kaggle, which provide free tools, tutorials, and datasets to empower individuals from diverse backgrounds to develop and apply machine learning techniques. Together, these resources lower barriers to entry, enabling a wider audience to engage with cutting-edge science and technology, ultimately promoting inclusivity and innovation in both fields.

.2. *Challenges with Land Use in Astronomy Projects*

Challenges with land use in astronomy arise when proposed sites for ground-based observatories overlap with lands that hold cultural, spiritual, or historical significance for indigenous communities. Locations like Mauna Kea in Hawaii and Cerro

Armazones in Chile are often chosen for their ideal observing conditions, such as high altitude and minimal light pollution. However, these sites can also be deeply significant to local communities, leading to tensions between the pursuit of scientific progress and the preservation of cultural traditions. In some cases, these tensions stem from inadequate consultation with or lack of consent from the communities affected by these projects. Indigenous groups may feel that their perspectives and concerns are not adequately considered during the planning and construction phases. Additionally, these projects can sometimes disrupt local ecosystems or traditional practices tied to the land.

The tension surrounding the use of Indigenous lands such as Maunakea in Hawaii, arises from the intersection of scientific progress and Indigenous rights. This conflict showcases the failure to adequately consult or obtain consent from Native Hawaiian communities, who view Maunakea as a sacred site central to their cultural and spiritual heritage. The construction of telescopes on this land, including the controversial Thirty Meter Telescope, has been met with resistance, including protests and legal challenges, as it symbolizes the ongoing disenfranchisement of Indigenous people and their sovereignty. The root of the issue lies in the historical marginalization of Native Hawaiians, where decisions regarding land use have often prioritized scientific advancement over cultural preservation and self-determination (Kahanamoku 2020). For the scientific community, this serves as a stark reminder of the importance of equitable practices, where Indigenous voices must be central to decision-making processes. The situation with Maunakea underscores a broader challenge in reconciling the goals of modern science with the rights and values of Indigenous communities, urging astronomers and policymakers to approach such projects with greater sensitivity and respect for cultural significance.

This project exclusively uses data collected by space telescopes such as those accessed through the NASA Exoplanet Archive, MAST, and the TESS SPOC Pipeline. Space telescopes operate from orbit, beyond the constraints of geographic or terrestrial boundaries, which inherently removes the potential for land use conflicts often associated with ground-based observatories. By avoiding the need for physical infrastructure on culturally or historically significant lands, space-based data collection bypasses the ethical and social complexities that arise from disputes over land use. Additionally, the data from these telescopes are made openly accessible, fostering inclusivity and enabling researchers from around the world to engage with exoplanet research without encountering the barriers posed by restricted or proprietary data sources. This reliance on space-based observatories ensures that the study is built on a foundation free from the controversies linked to land use, allowing the focus to remain on scientific exploration and innovation.

REFERENCES

Agarap, A. F. arXiv preprint arXiv:1803.08375 (2018).

Jenkins, Jon M., et al. Software and Cyberinfrastructure for Astronomy IV. Vol. 9913. SPIE, 2016.

Kahanamoku, Sara, et al. arXiv preprint arXiv:2001.00970 (2020).

Kingma, Diederik P. arXiv preprint arXiv:1412.6980 (2014).

LeCun, Yann, and Yoshua Bengio. The handbook of brain theory and neural networks 3361.10 (1995): 1995.

Mayor, Michel, and Didier Queloz. Nature. 378.6555 (1995): 355-359.

Qian, Ning. Neural networks 12.1 (1999): 145-151.

Ricker, George R., et al. Journal of Astronomical Telescopes, Instruments, and Systems 1.1 (2015): 014003-014003.

Ruder, Sebastian. arXiv preprint arXiv:1609.04747 (2016).

Theodoridis, Sergios. Academic press, 2015.