Classifying Exoplanet Types from the NASA Exoplanet Archive

Data Science Project

By

Tanya Sharma (EP22BTECH11028)
Harshini Dongarwar (MS22BTECH11013)

Introduction

The discovery and classification of exoplanets—planets that orbit stars outside our solar system—have revolutionized our understanding of planetary systems and their formation.

In recent years, the volume and diversity of exoplanet data have grown significantly, making it increasingly important to apply data-driven techniques for analysis and discovery. Traditional methods of astronomical data interpretation are often labor-intensive and limited by human bias whereas machine learning offers scalable and unbiased alternatives for pattern recognition and classification tasks.

With the increasing availability of observational data from missions such as Kepler, TESS and others, large databases like the NASA Exoplanet Archive have become invaluable resources for planetary science and machine learning applications.

By analyzing a curated subset of the NASA Exoplanet Archive, this project explores how different planetary and stellar features contribute to the likelihood of an exoplanet being discovered via a specific method such as transit, radial velocity or direct imaging. Understanding these associations not only improves detection strategies but can also guide future observational campaigns.

The use of supervised learning algorithms enables the model to learn from labeled historical data and generalize this knowledge to classify newly observed systems. This framework provides a foundation for predictive modeling in astronomical research, where labeling new data manually is often impractical or infeasible.

The goal is to develop predictive models that can accurately infer the discovery method of an exoplanet using features such as mass, radius, orbital period, equilibrium temperature and stellar properties.

This classification task not only highlights the correlations between planetary properties and detection techniques but also serves as a demonstration of how machine learning can assist astronomers in handling large-scale astronomical datasets.

Furthermore, this work contributes to the broader goal of **automating scientific discovery in astronomy**, paving the way for more efficient exploration of the universe through intelligent systems that can analyze complex, high-dimensional datasets with minimal human intervention.

Feature Selection and Dataset Overview

➢ For the purpose of this project, we selected a subset of features from the full exoplanet dataset to focus on variables relevant to planetary and stellar characteristics. The selected features include:

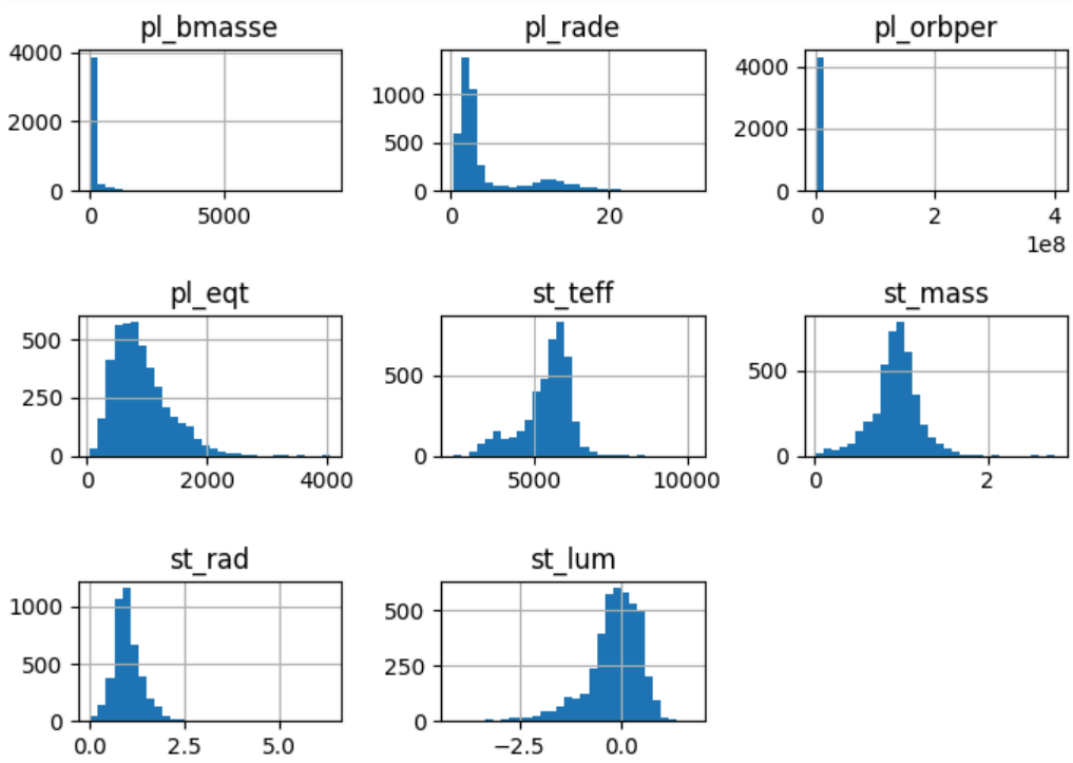
- pl_bmasse: Planetary mass (in Earth masses)
- pl_rade: Planetary radius (in Earth radii)
- pl_orbper: Orbital period (in days)
- pl_eqt: Equilibrium temperature of the planet (in Kelvin)
- st_teff: Effective temperature of the host star (in Kelvin)
- st_mass: Stellar mass (in Solar masses)
- st_rad: Stellar radius (in Solar radii)
- st_lum: Stellar luminosity

➢ The resulting dataset consists of **4,310 entries** and **8 numerical features**, all of type float64.

	Name	Count	Non-Null	Dtype
0	pl_bmasse	4310	non-null	float64
1	pl_rade	4310	non-null	float64
2	pl_orbper	4310	non-null	float64
3	pl_eqt	4310	non-null	float64
4	st_teff	4310	non-null	float64
5	st_mass	4310	non-null	float64
6	st_rad	4310	non-null	float64
7	st_lum	4310	non-null	float64

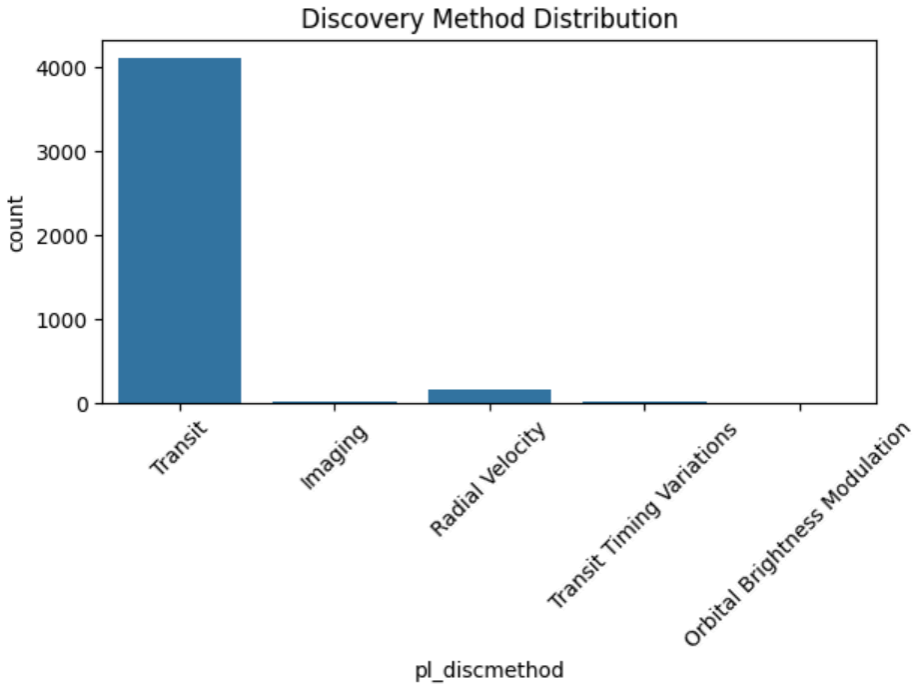
Exploratory Data Analysis: Feature Distributions

➢ To understand the structure of the selected features, histograms were plotted for each variable using pandas. This visual representation helps assess the **distribution**, **skewness** and **presence of outliers** in the data.



1. **pl_bmasse**: Highly right-skewed distribution, indicating most exoplanets have relatively low masses while a few have extremely high masses.
2. **pl_rade**: Right-skewed distribution, indicating most exoplanets have relatively small radii.
3. **pl_orbper**: Highly right-skewed distribution, indicating most exoplanets have relatively short orbital periods.
4. **pl_eqt**: Right skewed with most planets having moderate equilibrium temperatures.
5. **st_teff**: Shows normal distribution around 5500-6000 K.
6. **st_mass**: Normal distribution near 1 solar mass.
7. **st_rad**: Right skewed distribution with larger outliers.
8. **st_lum**: Left skewed distribution for the majority of planets.

Discovery Method Distribution



➢ This bar plot displays the distribution of exoplanets based on the method used for their discovery. It is shown that a few methods dominate the dataset particularly the **Transit** and **Radial Velocity** methods. This imbalance in discovery methods impacts the dataset and model performance significantly. For instance:

- **Transit** and **Radial Velocity** methods are more likely to detect large or close-in planets, introducing **bias** in the observed physical features.

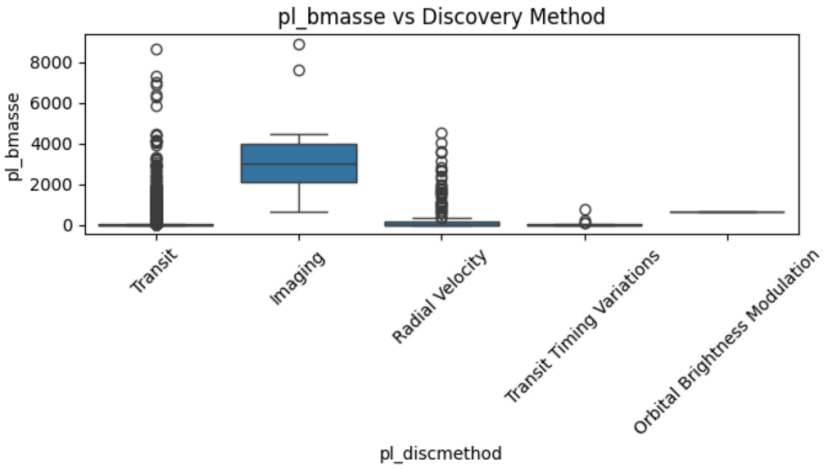
- Models trained on this dataset may perform better on planets discovered through these dominant methods and may **underperform on underrepresented methods** due to lack of sufficient training examples.

Therefore, understanding this distribution is crucial for interpreting the results of any model trained on this data especially in terms of **class imbalance** and **generalizability** to other discovery techniques.

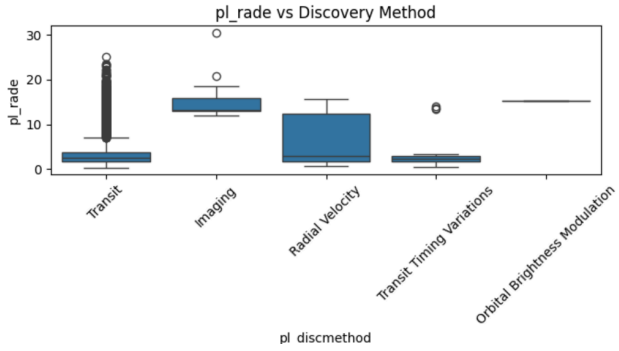
Plots between features & discover method

- To better understand the relationship between planetary properties and the discovery methods, we plotted boxplots for all key features.
- These plots reveal how the distribution of each feature varies with the detection technique. For example, the **Transit** method generally finds smaller or **close-in planets** (lower mass, radius and orbital period) while methods like **Radial Velocity** are more likely to detect massive or **distant planets**.
- Each method shows unique biases due to its observational constraints with some distributions showing tight clustering and others revealing a wide spread or significant outliers.
- These visualizations are essential as they highlight how each feature may influence the modeling and interpretation of the dataset. By analyzing the feature-wise distributions across methods, we can anticipate potential **biases**, understand **feature importance** and design **robust models** that generalize beyond method-specific patterns.

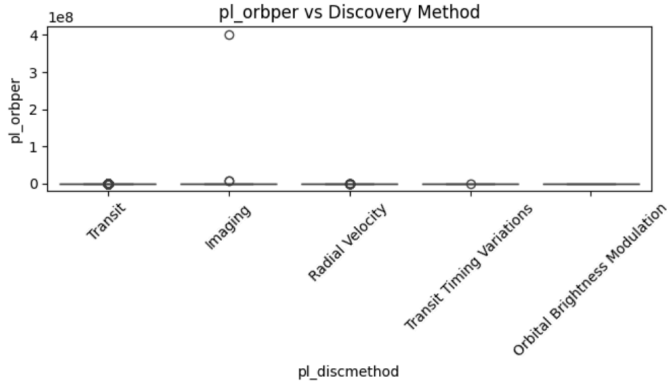
Plot between pl_bmasse & Discovery Method



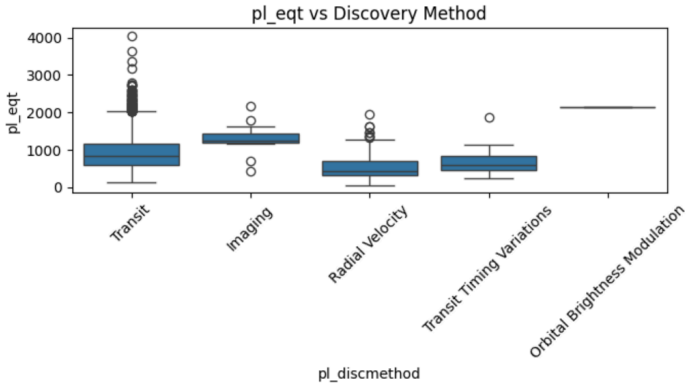
Plot between pl_rade & Discovery Method



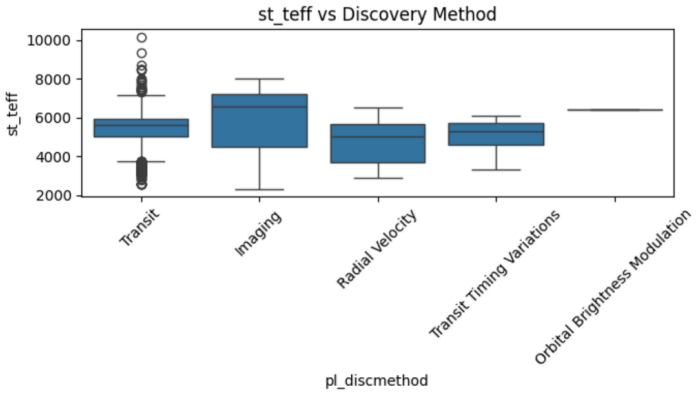
Plot between pl_orbper & Discovery Method



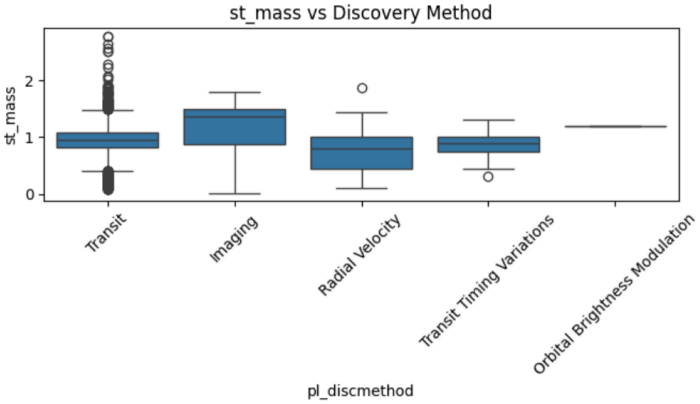
Plot between pl_eqt & Discovery Method



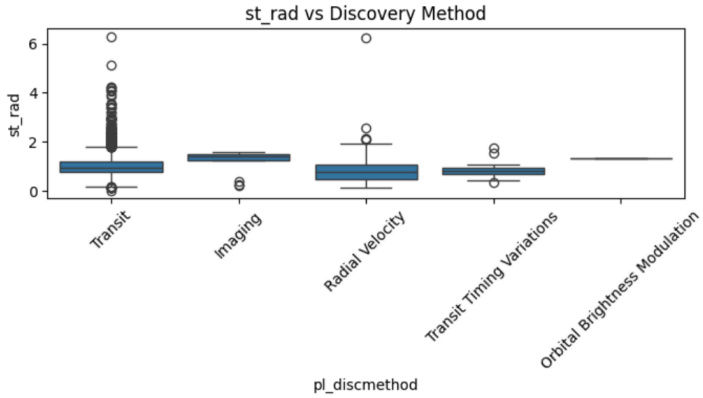
Plot between st_teff & Discovery Method



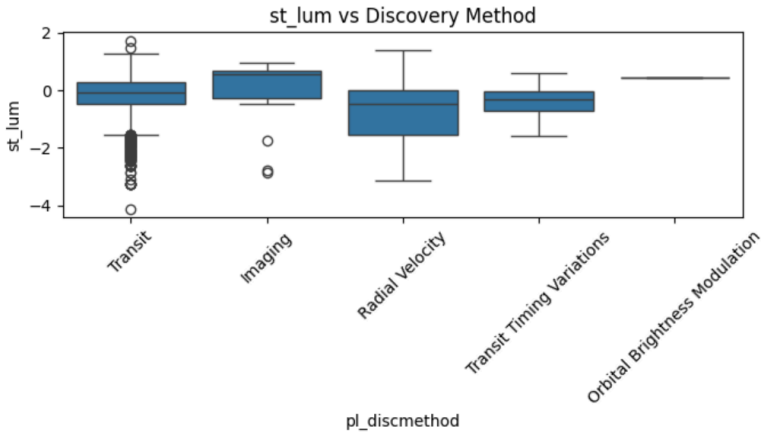
Plot between st_mass & Discovery Method



Plot between st_rad & Discovery Method

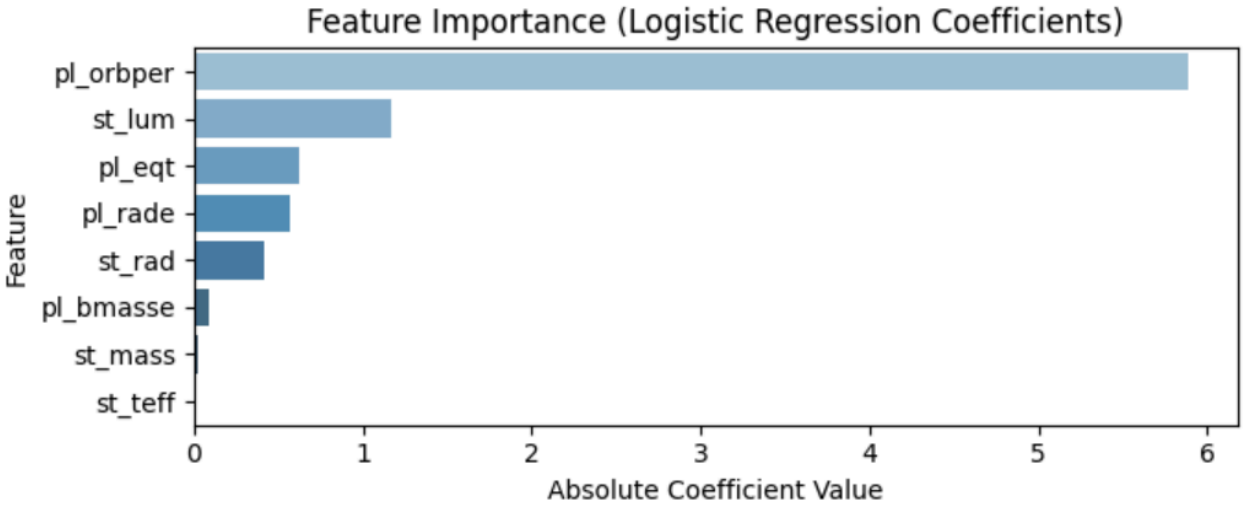


Plot between st_rad & Discovery Method



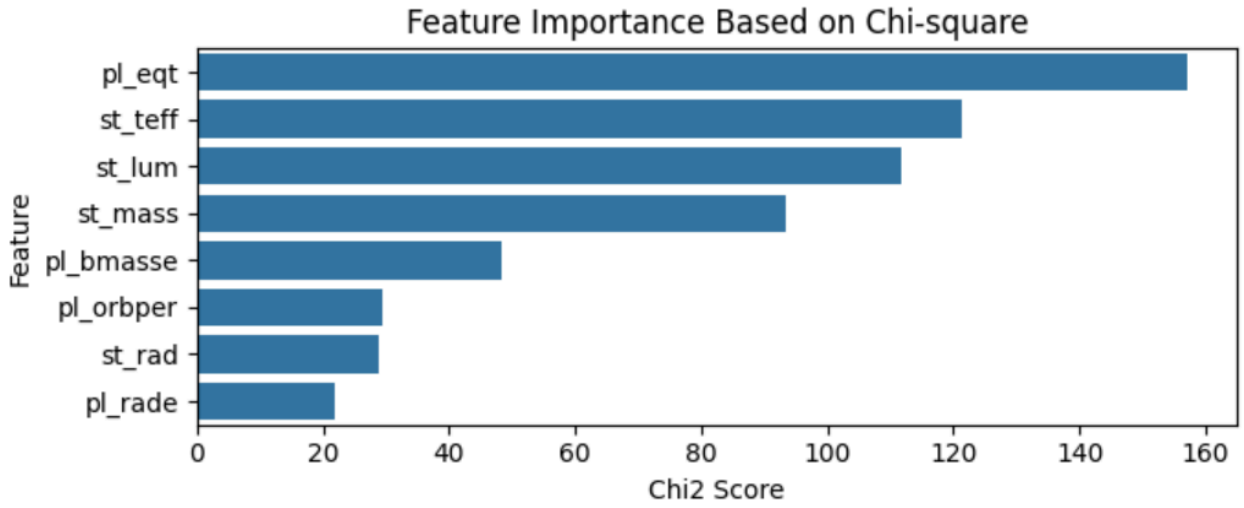
Feature Importance using Logistic Regression

- ➢ Logistic Regression is not only a classification algorithm but also provides a way to interpret feature importance in linear models.
- ➢ In this analysis, we use Logistic Regression to estimate the influence of each numerical feature on the target variable, i.e., the discovery method of exoplanets.
- ➢ By fitting a logistic model to the data, the coefficients (weights) assigned to each feature indicate their relative importance: a higher absolute value of a coefficient suggests a stronger relationship with the prediction.
- ➢ Before fitting the model, feature scaling (such as standardization) is typically applied to ensure comparability across variables.
- ➢ This approach allows us to interpret which planetary or stellar properties most significantly contribute to distinguishing between different detection methods.



Feature Selection using Chi-Square Test

- The Chi-Square (χ^2) test is a statistical method used to assess the independence between categorical variables.
- In the context of feature selection, we use the Chi-Square test to evaluate the dependency between individual numerical features and the categorical target variable, which in this case is the discovery method.
- The test measures how the observed distribution of feature values across different classes deviates from the expected distribution under the assumption of independence.
- Features that produce higher Chi-Square statistics are considered more relevant, as they show greater disparity in distribution across classes.



Why is there difference between the feature importance of Logistic regression and Chi square Test?

Chi-square test and logistic regression are having different feature importance because they assess feature relevance using fundamentally different approaches.

- ➢ The **Chi-square test** is a univariate statistical test that evaluates the independence between each feature and the target variable. It works well for categorical target variables and is sensitive to the distribution of values in discrete bins.
- ➢ In contrast, **logistic regression** considers the **multivariate relationships** between all features and the target simultaneously. It assigns weights based on how each feature contributes to maximizing the separation between classes in a logistic model.

Thus, the difference in rankings is not a contradiction but a reflection of the distinct ways these methods evaluate feature influence. Combining insights from both can lead to more robust feature selection.

Model Training and Classification Algorithms

- ➢ To classify exoplanets based on their discovery methods, we proceed to the model training phase using three distinct machine learning algorithms: Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM).
- ➢ Each of these models represents a different class of learning paradigms—ensemble methods, instance-based learning and margin-based classifiers respectively.
- ➢ The models are trained on a labeled dataset where the planetary and stellar features serve as input variables and the discovery method serves as the target label.
- ➢ The objective is to evaluate and compare the predictive performance of each algorithm in classifying the detection technique based on physical attributes of exoplanets and their host stars. Evaluation metrics such as accuracy, precision, recall and F1-score are used to assess the effectiveness of each model on the test dataset.

Comparison between K-nearest neighbours,Random Forest and Support Vector Machine

- We have decided to train these 3 models to check how accurately they can predict if the planets were detected through Transit or Radial Velocity method.
- We got the data from the Nasa Exoplanet Archive.
- The features which we used were:
    - pl_bmasse: Estimated mass of the planet in Earth masses.
    - pl_rade:Radius of the planet expressed in Earth radii.
    - pl_orbper:Time the planet takes to complete one full orbit around its star, in days.
    - pl_eqt:Estimated temperature of the planet assuming it is a blackbody and has no atmosphere, in Kelvin.
    - st_teff:The surface temperature of the host star, in Kelvin.
    - st_mass:Mass of the star in units of the Sun's mass.
    - st_rad:Radius of the star in units of the Sun's radius
    - st_lum:Logarithmic luminosity (brightness) of the star compared to the Sun.
- We then resampled the data because we found 4117 entries for transit and 159 for Radial velocity.
- We oversampled Radial velocity so that we have 2000 entries of RV.
- The test size was kept as 0.2.

PCA analysis on Models:

- To improve model efficiency and explore the impact of dimensionality reduction on classification performance, we applied Principal Component Analysis (PCA) to our models.
- For each number of components, PCA was applied to both the training and test sets to transform the feature space.
- Each model was then trained on the PCA-transformed training data.
- For each configuration, we recorded the accuracy, explained variance, and the confusion matrix to evaluate model performance.
- To evaluate the impact of dimensionality reduction, we compared the performance of each model trained on the original dataset (without PCA) versus the same model trained on datasets transformed using Principal Component Analysis (PCA) with varying numbers of components.

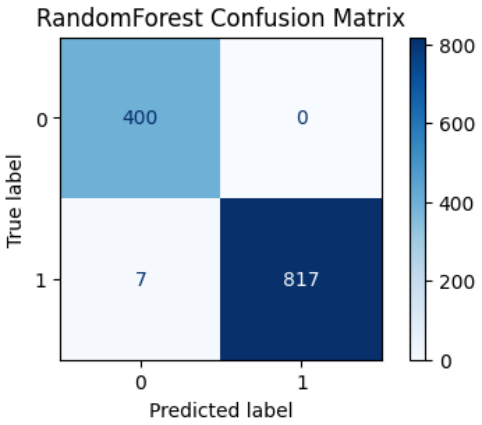
RESULTS WITHOUT PCA

The following results were observed testing our models(test_size=0.2):

RANDOM FOREST

- Overall Accuracy:0.9943

Class	Precision	Recall	F1-Score
Radial Velocity	0.98	0.99	0.99
Transit	1.00	0.99	1
Macro Avg	0.99	1.00	0.99
Weighted Avg	0.99	0.99	0.99

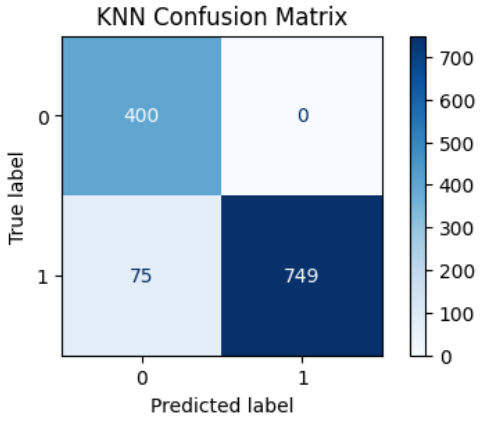


- The model is extremely reliable at identifying both discovery methods.
- It never mistakes a Radial-Velocity planet for a Transit planet and only 0.85% (7/824) of Transit planets are mislabeled as Radial-Velocity.
- Both macro‑ and weighted‑averages are ~0.99, showing balanced performance despite the class‑imbalance (824 vs. 400).

KNN

➢ Overall Accuracy:0.9387

Class	Precision	Recall	F1-Score
Radial Velocity	0.84	1	0.91
Transit	1.00	0.91	0.95
Macro Avg	0.92	0.95	0.93
Weighted Avg	0.95	0.94	0.94



➢ Perfect recall on Radial-Velocity: KNN never misses a true RV planet, but at the cost of misclassifying many Transit planets as RV (lower precision on RV).

➢ High precision on Transit: Every time KNN predicts “Transit,” it’s correct—but it fails to catch ~9% of true Transits (lower recall).

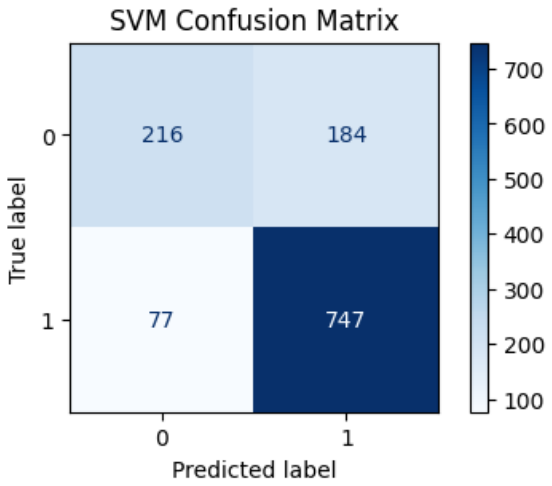
➢ Overall lower F1-scores (0.91 RV, 0.95 Transit) and accuracy (0.9387) indicate that KNN is less capable of separating these two discovery methods than Random Forest.

➢ This suggests that KNN struggles in this feature space—possibly because Transit and RV classes overlap in the original seven-dimensional space in a way that distance-based classification can’t cleanly separate without more sophisticated decision boundaries.

SVM

➢ Overall Accuracy:0.7868

Class	Precision	Recall	F1-Score
Radial Velocity	0.74	0.54	0.62
Transit	0.80	0.91	0.85
Macro Avg	0.77	0.72	0.74
Weighted Avg	0.78	0.79	0.78



➢ The poor recall on the minority class (Radial-Velocity) suggests the SVM's linear decision boundary cannot capture the separation between the two methods in feature space.

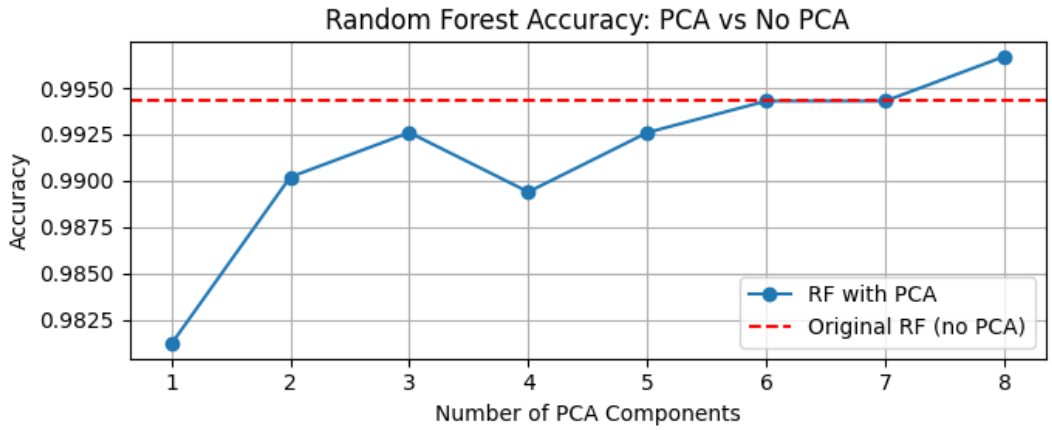
➢ It's over-favoring the majority class (Transit), producing too many RV→T errors.

➢ In contrast, Random Forest and even a simple distance‑based KNN handle the feature interactions better.

RESULTS WITH PCA

Random Forest

# Components	Accuracy	Explained Variance
1	0.9812	0.6763
2	0.9902	0.8893
3	0.9926	0.9672
4	0.9894	1.0000
5	0.9926	1.0000
6	0.9943	1.0000
7	0.9943	1.0000
8	0.9967	1.0000

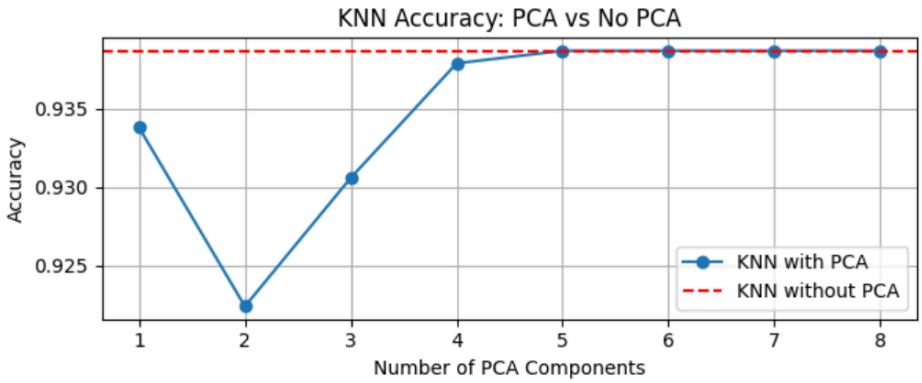


- The first three principal components already capture 96.7 % of the total variance in the original eight-dimensional feature space.

- By the fourth component, 100 % of the variance is retained, indicating that the remaining features lie within the subspace spanned by those four components.

K Nearest Neighbours(KNN)

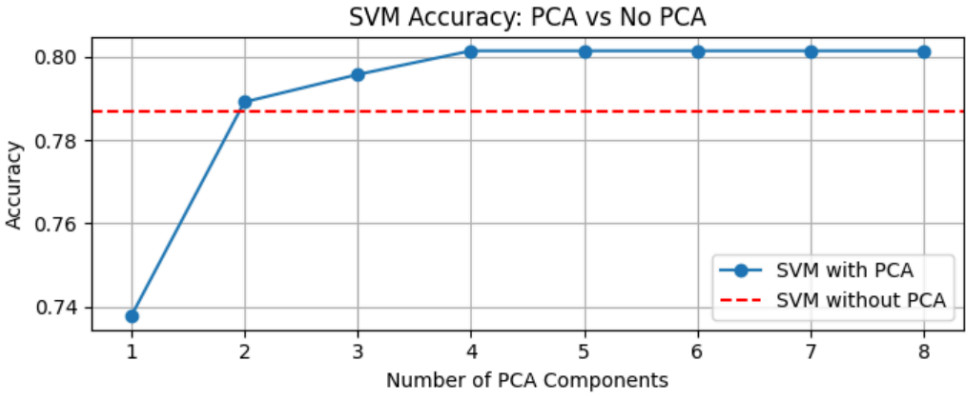
PCA Components	Accuracy	Explained Variance
1	0.9338	0.6763
2	0.9224	0.8893
3	0.9306	0.9672
4	0.9379	1.0000
5	0.9387	1.0000
6	0.9387	1.0000
7	0.9387	1.0000
8	0.9387	1.0000



- High accuracy (~93–94%) is achieved with just 1–4 PCA components.
- Explained variance reaches 100% by 4 components, so more components don't improve performance.

Support Vector Machine (SVM)

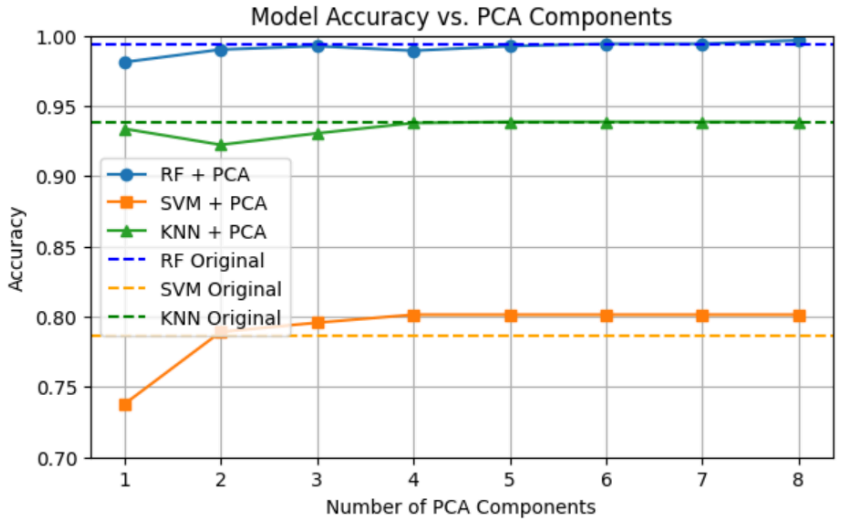
PCA Components	Accuracy	Explained Variance
1	0.7377	0.6763
2	0.7892	0.8893
3	0.7958	0.9672
4	0.8015	1.0000
5	0.8015	1.0000
6	0.8015	1.0000
7	0.8015	1.0000
8	0.8015	1.0000



- Accuracy improves from 0.7377 (1 component) to 0.8015 (4 components).
- After 4 components, adding more features does not improve accuracy — it plateaus.

CONCLUSION

- In this project, we explored the classification of exoplanet discovery methods using machine learning techniques applied to data from the NASA Exoplanet Archive. Starting with comprehensive data cleaning and exploratory analysis, we examined the distributions, correlations and relationships between planetary and stellar features. Statistical tests and visualizations helped us understand the impact of each feature on the classification task guiding our feature selection process.
- To evaluate the predictive performance, we trained and tested three different classification algorithms—Random Forest (RF), Support Vector Machine (SVM) and K-Nearest Neighbors (KNN)—on both the original feature space and the reduced-dimensional space obtained using Principal Component Analysis (PCA). These models were selected to capture both linear and non-linear patterns in the data and provide a comparative understanding of how well each method handles the classification task.
- Finally, to assess and compare the models comprehensively, we present a combined performance plot showing the accuracy of each algorithm with and without PCA. This visualization highlights the trade-offs between model complexity, dimensionality reduction and classification performance, offering insight into which combinations yield the most robust and accurate predictions. This analysis reinforces the role of machine learning in helping astronomical discovery and emphasizes the importance of thoughtful preprocessing, feature selection and model evaluation in scientific data analysis.



Code: https://github.com/tanyash05/DataScienceProject