Automated Exoplanet Detection and Analysis: Leveraging Machine Learning to unveil Astronomical Insights

Github: https://github.com/ayesha-mohsin/ExoplanetDetect

Shaik Ayesha Mohsin (11601110) **Areeba Hassan (11641013)** **Mohammed Faisal (11632846)**

1. Abstract

Exploring beyond the confines of our solar system, the investigation of exoplanets stands as an exhilarating frontier in astronomical exploration. This project, titled "Automated Exoplanet Detection and Analysis," is an endeavor to harness the power of machine learning for the precise identification of exoplanets within star systems. Leveraging light intensity curves obtained from the NASA Kepler space telescope, our approach involves a comprehensive pipeline of data preprocessing, feature engineering, and the development of sophisticated machine learning models. The motivation behind this endeavor lies in the desire to automate and enhance the accuracy of exoplanet detection, contributing to a deeper understanding of the universe's complexity and diversity.

The significance of this project extends beyond its immediate astronomical implications. Successful automated exoplanet detection not only aids in unraveling the mysteries of celestial bodies but also holds promise for identifying potentially habitable worlds. The amalgamation of data science techniques, advanced machine learning models, and detailed feature engineering forms the backbone of our methodology. As we delve into the nuances of light intensity curves, the project aims not only to predict exoplanetary presence but also to provide valuable insights into the characteristics and patterns that define these distant worlds. This automated approach signifies a step forward in the realm of exoplanetary research, paving the way for a more efficient and thorough analysis of the vast troves of astronomical data at our disposal.

2. Introduction

In the vastness of the cosmos, a fundamental question persists: Are we alone? Humanity has tirelessly pursued the quest to unravel signs of life beyond our planet. This exploration comprises two vital tasks: identifying exoplanets, including those potentially habitable, and studying their atmospheric properties through spectroscopy to assess habitability. This project focuses on the initial phase — discovering exoplanets. Historically, this has been a laborious endeavor, resulting in the detection of only approximately 4000 exoplanets over two decades. However, leveraging machine learning can automate and expedite this process significantly.

Exoplanets, planets beyond our solar system, reveal themselves through a technique known as transit photometry. As an exoplanet orbits its host star, it intermittently obstructs a portion of the star's light, causing a momentary dip in light intensity. Analyzing the depth and duration of these light dips aids in identifying and confirming exoplanets, providing valuable insights into their

presence and characteristics within distant star systems. NASA has diligently collected this data using the Kepler Space Telescope, making it publicly available. This dataset serves as the foundation for automating exoplanet detection through machine learning and deep learning techniques.

3. Motivation

The exploration for exoplanets is a crucial step toward comprehending the possibility of life existing beyond Earth. Researchers traditionally faced a slow exoplanet discovery process due to the manual analysis of vast and complex astronomical data. Analyzing light curves and detecting subtle patterns required meticulous human scrutiny. Machine learning and deep learning algorithms can expedite this process by autonomously sifting through immense datasets, swiftly identifying patterns and anomalies in the light curves, enabling rapid and accurate detection of exoplanetary transits, thus revolutionizing the pace of discovery. By automating and accelerating the detection of exoplanets through advanced technologies like machine learning, it is possible to greatly enhance the ability to identify celestial bodies that may harbor life. This project is fueled by the desire to expedite exoplanet discovery, unlocking a deeper comprehension of our universe and the tantalizing possibility of life existing beyond our home planet.

4. Significance

Our project introduces a pioneering approach in the realm of exoplanet detection by seamlessly integrating cutting-edge advancements in machine learning while embracing the paradigm of Explainable AI (XAI). By implementing techniques that shed light on the model's decision-making process, we ensure transparency and interpretability, setting our work apart from existing studies. Furthermore, we emphasize the incorporation of hyper-parameter optimization using methodologies like HyperOpt, foreseeing a future where optimizing neural network architectures becomes standard practice. This proactive approach positions our project at the forefront of machine learning algorithms, ensuring adaptability to forthcoming advancements. Moreover, we are committed to validating our model's performance using attention maps and class activation maps, thus ensuring that our algorithm avoids overfitting and makes informed decisions based on relevant features from the input space. By combining these unique elements, our project stands as a beacon of innovation, promising to significantly advance the accuracy, interpretability, and efficiency of exoplanet detection methodologies.

5.Objectives

To facilitate clarity, a comprehensive list of project objectives is presented below.

- Data Preprocessing
  a. Handle missing values and potential outliers in the dataset.
  b. Scale and normalize the values to ensure uniformity and aid in model training.
- Model Development and Training

a. Implementing and training a variety of chosen machine learning and deep learning models like K Nearest Neighbors, Logistic Regression, Decision Tree, Random Forest, and XGBoost among others.

b. Train these models on the provided training dataset, optimizing hyperparameters to enhance performance.

- Evaluation and Model Selection

  a. Evaluate models using appropriate metrics such as accuracy, precision, recall, F1-score, and AUC-ROC curve.

  b. Select the best-performing model based on evaluation results for further refinement.
- Fine-Tuning and Optimization

  a. Optimize the selected model further for improved accuracy and efficiency.

  b. Experiment with hyperparameters to achieve the best possible outcomes.
- Testing and Validation:

  a. Apply the optimized model to the provided test dataset for final validation.

  b. Assess the model's generalizability and performance on unseen data.
- Visual Representation:

  a. Develop visualizations to present results, including confusion matrices, ROC curves, and feature importance plots.

  b. Create informative and intuitive visuals to aid in the understanding and interpretation of the model's performance.
- Enhanced model interpretability:

  a. Utilize techniques like Explainable AI (XAI) that shed light on the model's decision-making process, thereby ensuring transparency and interpretability.

6. Features

The different features or modules which will be implemented in the project are as follows:

1. Pre-processing:
   - Handling missing values in the dataset.
   - Scaling the features, using techniques like Standardization.
2. Model Training:
   - Implementing and training a variety of chosen machine learning and deep learning models like K Nearest Neighbors, Logistic Regression, Decision Tree, Random Forest, and XGBoost among others.
   - Hyperparameter tuning to optimize model performance.
3. Evaluation:
   - Assessing models using appropriate metrics like accuracy, precision, recall, F1-score, and AUC-ROC.
   - Comparing models and selecting the best-performing one.
   - Creation of visual representations illustrating model predictions on light curves, aiding in a clear understanding of the detection process.

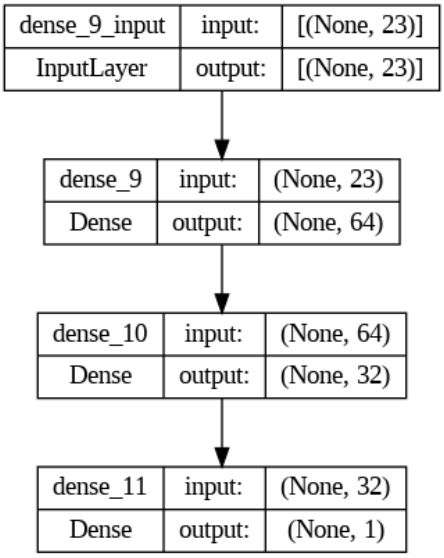
7. Literature Survey

The first paper chosen for the literature survey was "Automated Identification of Transiting Exoplanet Candidates in NASA Transiting Exoplanet Survey Satellite (TESS) Data with Machine Learning Methods"[1] - Leon Ofman et al. In this study by Ofman et al., a novel AI technique is introduced for the automated identification of transiting exoplanet candidates in TESS data using machine learning methods. The AI/ML system, developed by ThetaRay, Inc., is trained with Kepler exoplanetary data and validated with confirmed exoplanets before being applied to TESS data. Employing both semi-supervised and unsupervised ML techniques, the system analyzes over 10,000 light curves, yielding about 50 targets for further analysis. Notably, three new exoplanetary candidates are discovered through manual vetting, showcasing the effectiveness of the approach in rapidly classifying threshold crossing events (TCEs) within large astrophysical datasets.

The second paper, "Exoplanet Detection Using Machine Learning"[2] - Abhishek Malik et al., presents a machine learning-based technique for exoplanet detection, focusing on the transit method. Utilizing the TSFRESH library for time series analysis, the study extracts 789 features from each light curve, offering a detailed characterization of the data. The LIGHTGBM gradient boosting classifier is then employed, demonstrating computational efficiency compared to traditional methods. Testing on K2 campaign 7 and Transiting Exoplanet Survey Satellite data reveals competitive performance, with notable achievements such as an AUC of 0.948 for Kepler data and an accuracy of 0.98 for TESS data. The method's ability to predict planets efficiently without requiring folded or secondary views of the light curves makes it a promising approach in the realm of exoplanet detection.

8. Our Approach

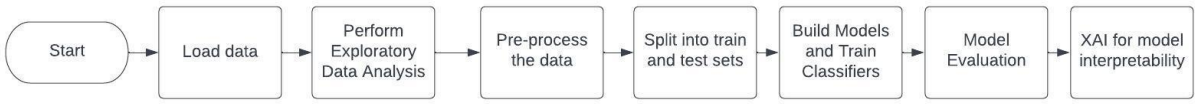
8.1. Architecture Diagram



The Artificial neural network model, constructed using the Keras library, comprises three layers with distinct configurations. The initial layer, a Dense layer with 64 units, employs the Rectified Linear Unit (ReLU) activation function, signifying that every neuron is connected to every neuron in the previous layer. The input shape of this layer is determined by the features in the input data. Subsequently, a second Dense layer follows with 32 units and ReLU activation, receiving input from the preceding layer. The final layer, a Dense layer with a single unit and a Sigmoid activation function, is designed for binary classification tasks. This architecture is visualized using the plot_model function, which produces a graphical representation elucidating the flow of data through the layers and the specific configurations of each layer.

In summary, the neural network is tailored for binary classification tasks, leveraging the power of dense interconnected layers and well-suited activation functions for efficient feature extraction and representation.

8.2. Workflow Diagram



The workflow of the model involves a systematic pipeline to address the challenges and intricacies of exoplanet detection:

- Data Loading and Exploration:
    - Data is loaded from the NASA Kepler space telescope dataset, consisting of flux readings and labels indicating the presence or absence of exoplanets.
    - Exploratory Data Analysis (EDA) is performed to understand the dataset's characteristics, including class distribution, correlation, and flux variations.
- Data Preprocessing:
    - Missing values are handled, and outliers are detected and removed to enhance the quality of the dataset.
    - Synthetic Minority Over-sampling Technique (SMOTE) is employed to address class imbalance, generating synthetic instances of the minority class.
    - Gaussian filters are applied to smooth fluctuations in flux readings, emphasizing underlying patterns associated with exoplanet transits.
    - Feature scaling, normalization, and Gaussian filtering contribute to preparing the data for model training.
    - Principal Component Analysis (PCA) is utilized for dimensionality reduction, transforming the dataset into uncorrelated principal components.

    - The number of principal components is determined to retain 90% of the dataset's variance, balancing information retention and dimensionality reduction.
- Model Building:
    - Various machine learning models, including K Nearest Neighbors, Logistic Regression, Bernoulli Naive Bayes, Decision Tree, Random Forest, XGBoost, and Artificial Neural Network, are implemented and trained.
    - Hyperparameter tuning is performed to optimize model performance.
- Model Evaluation:
    - Models are evaluated using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC.
    - Comparative analysis is conducted to select the best-performing model.
- Explainability (XAI):
    - Local Interpretable Model-agnostic Explanations (LIME) and SHAP values are employed to interpret predictions, providing insights into the features influencing the model's decisions.

This structured workflow ensures a comprehensive and systematic approach to exoplanet detection, addressing data challenges and leveraging explainability techniques for model interpretation.

9. Dataset

9.1. Dataset Description

The Kepler Dataset, sourced from NASA, encompasses flux data from more than 3000 stars, each classified based on whether they host an exoplanet. Although planets themselves don't emit light, their host stars do. Continuous observation of these stars over months or years might reveal periodic 'dimming' in flux (light intensity). This dimming suggests the possible presence of a celestial body orbiting the star, rendering it a 'candidate' system. Additional scrutiny, possibly through a satellite capturing light at different wavelengths, can further confirm the candidacy of these systems.

The dataset comprises 5088 rows and 3198 columns, each representing light flux measurements for exoplanet detection. Each row contains 3197 flux values. This analysis aims to scrutinize Kepler mission data to pinpoint potentially habitable exoplanets. Each star is labeled with a binary classification: 2 indicates a confirmed presence of at least one exoplanet, including observations of multi-planet systems as shown in the figure 1.

LABEL	FLUX.1	FLUX.2	FLUX.3	FLUX.4	FLUX.5	FLUX.6	FLUX.7	FLUX.8	FLUX.9	FLUX.10	FLUX.11
2	93.85	83.81	20.1	-26.98	-39.56	-124.71	-135.18	-96.27	-79.89	-160.17	-207.47
2	-38.88	-33.83	-58.54	-40.09	-79.31	-72.81	-86.55	-85.33	-83.97	-73.38	-86.51
2	532.64	535.92	513.73	496.92	456.45	466	464.5	486.39	436.56	484.39	469.66
2	326.52	347.39	302.35	298.13	317.74	312.7	322.33	311.31	312.42	323.33	311.14
2	-1107.21	-1112.59	-1118.95	-1095.1	-1057.55	-1034.48	-998.34	-1022.71	-989.57	-970.88	-933.3
2	211.1	163.57	179.16	187.82	188.46	168.13	203.46	178.65	166.49	139.34	146.76
2	9.34	49.96	33.3	9.63	37.64	20.85	4.54	22.42	10.11	40.1	-13.05
2	238.77	262.16	277.8	190.16	180.98	123.27	103.95	50.7	59.91	110.19	16.41
2	-103.54	-118.97	-108.93	-72.25	-61.46	-50.16	-20.61	-12.44	1.48	11.55	38.69
2	-265.91	-318.59	-335.66	-450.47	-453.09	-561.47	-606.03	-712.72	-685.97	-753.97	-682.34
2	118.81	110.97	79.53	114.25	48.78	3.12	-4.09	66.2	-26.02	-52.34	-41.88
2	-239.88	-164.28	-180.91	-225.69	-90.66	-130.66	-149.75	-120.5	-157	-114.19	-70.66
2	70.34	63.86	58.37	69.43	64.18	52.7	47.58	46.89	46	38.88	46.46
2	424.14	407.71	461.59	428.17	412.69	395.58	453.35	410.45	402.09	380.14	393.22

Figure 1. NASA Kepler Flux Intensity Dataset Snippet

9.2. Detailed Design of Features

During the initial phase, the steps involved in feature engineering and data preprocessing implemented are as outlined below.

- **Handling Missing Values:** Strategies to deal with missing data were implemented, ensuring a comprehensive dataset for model training.
- **Outlier Detection and Removal:** Outliers were identified and addressed to prevent them from influencing model training.
- **Addressing Class Imbalance**: Synthetic Minority Over-sampling Technique (SMOTE) was utilized to balance the distribution of classes and prevent bias in model predictions. SMOTE is particularly advantageous in binary classification problems where one class is significantly underrepresented compared to the other. The goal of SMOTE is to balance the class distribution by generating synthetic examples of the minority class.
- **Data Normalization and Scaling:** Uniformity in the dataset was ensured by normalizing and scaling features, facilitating effective model training. This ensures that all features have a consistent scale, preventing certain features from dominating others in machine learning models.
- **Gaussian Filters for Noise Reduction:** Gaussian filters were applied to minimize noise in the light intensity curves, enhancing the models' ability to identify meaningful patterns.
- **Dimensionality Reduction using PCA:** Principal Component Analysis (PCA) was employed to reduce the dimensionality of the dataset, streamlining computational efficiency without compromising predictive power.

10. Analysis of Data

10.1. Data Pre-Processing

- The dataset under consideration comprises 5087 instances, each characterized by 3198 features, encompassing flux readings denoted as FLUX.1 to FLUX.3197, along with a binary label indicating the presence (1) or absence (0) of exoplanets.

- However, a notable observation made is the profound class imbalance within the labels, with a substantial majority of instances (5050) classified as non-exoplanets (0), while only 37 instances are labeled as exoplanets (1). This imbalance underscores the need for cautious model training to prevent potential biases and ensure a fair representation of both classes.

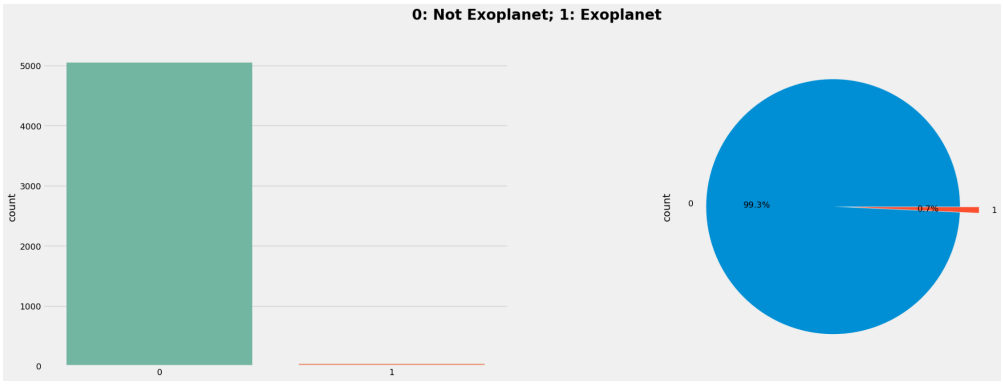


Figure 2. 0: Not Exoplanet; 1: Exoplanet

- Visual insights into the label distribution are provided through a countplot and a pie chart, shedding light on the disproportionate representation of exoplanet and non-exoplanet instances. The visualizations reveal that a mere 0.7% of instances are identified as exoplanets, emphasizing the pronounced prevalence of the non-exoplanet class, accounting for 99.3% of the dataset. This observation underscores the importance of adopting appropriate strategies to address class imbalance during model development.

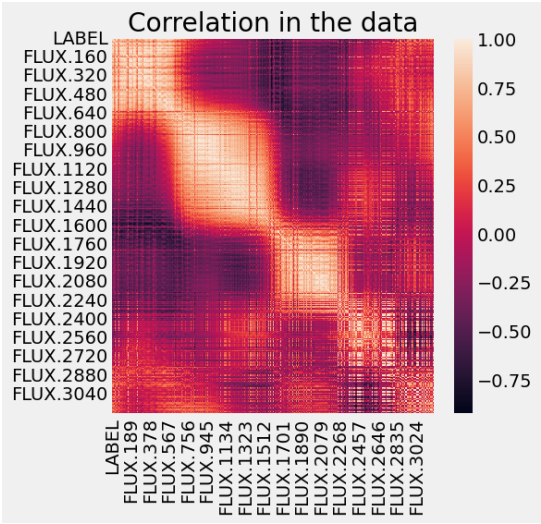
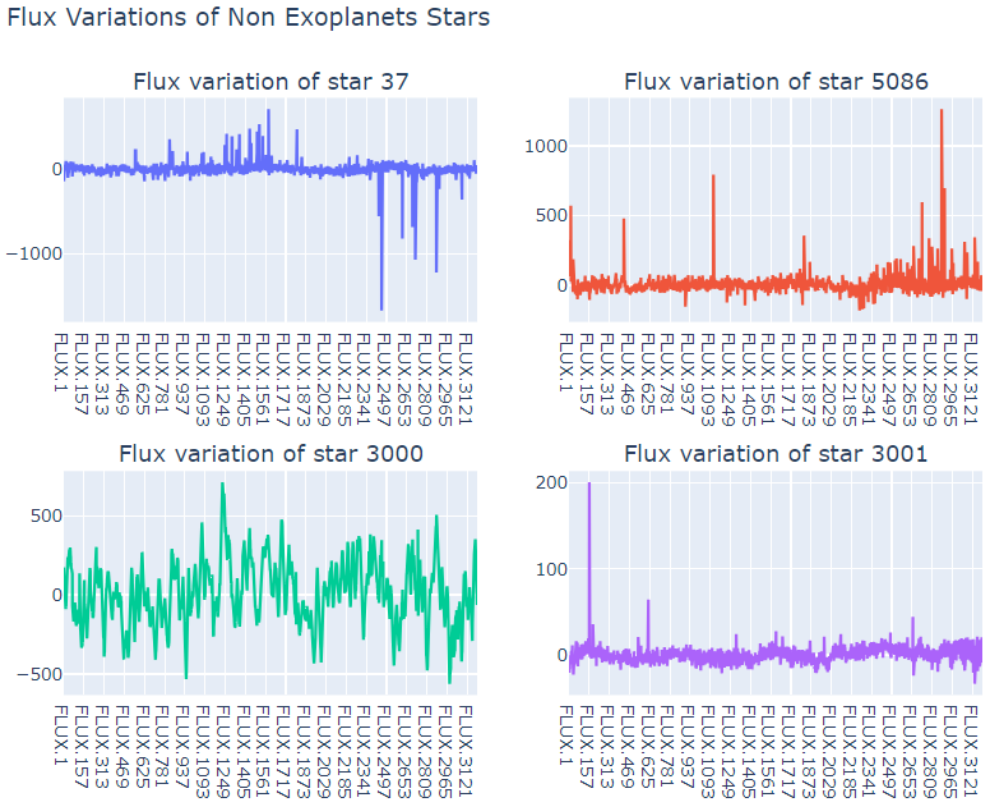


Figure 3. Correlation in the data

- Additionally, a correlation matrix is examined to elucidate potential relationships between different flux readings. However, owing to the independent nature of flux measurements at distinct time intervals and the transient characteristics of astronomical phenomena, the correlation matrix fails to yield meaningful insights. This underscores the unique challenges associated with astronomical datasets, where consecutive

measurements lack direct influence, limiting the applicability of traditional correlation analyses.



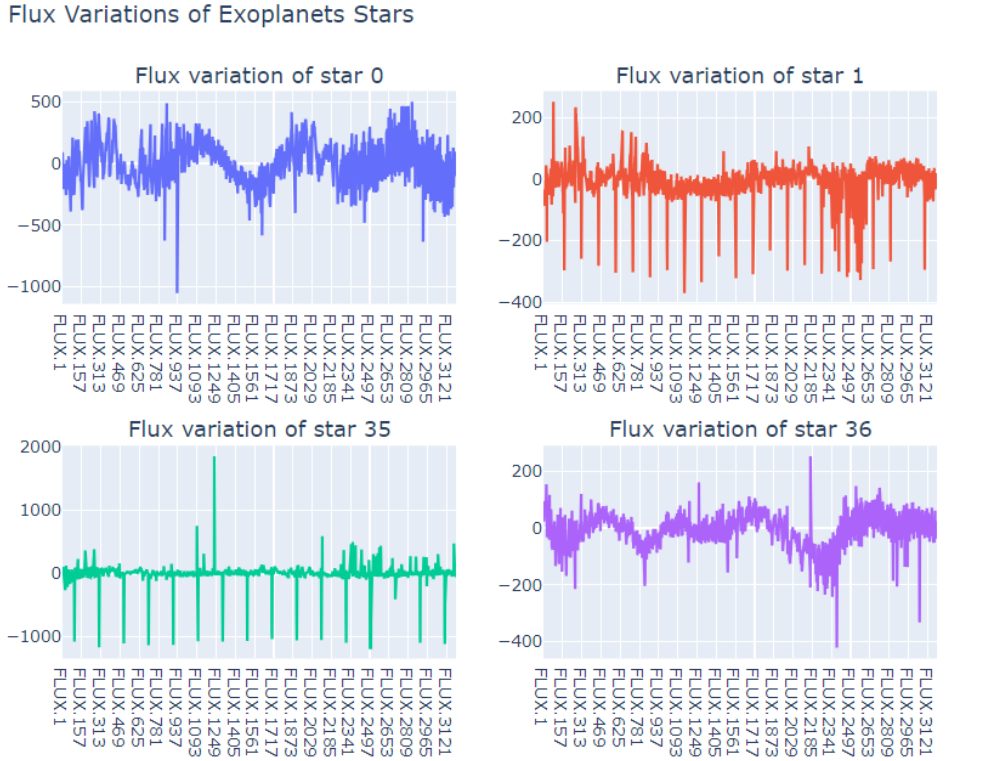


Figure 4. Distribution of Flux

- Further exploration involves a visual inspection of the flux readings for the first row, emphasizing periodic patterns across different stars. Clear periodic patterns are observed in all the exoplanet plots, indicating consistent fluctuations in the flux. These patterns are attributed to the presence of a planet orbiting in front of the respective stars, causing periodic reductions in the received flux. While some anomalies from detection errors are still visible, the overall observation suggests the influence of orbiting planets on the light intensity curves.

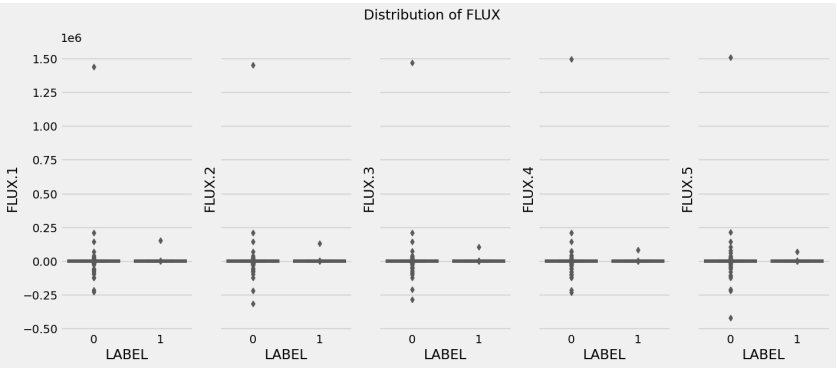


Figure 5. Distribution of Flux for outlier removal

- Outliers are observed in the flux distributions and then removed, in order to prevent them from negatively influencing the performance.

In conclusion, the dataset exhibited intricate patterns and challenges, including class imbalance and the unique characteristics of astronomical data. These nuances necessitated careful consideration during model development, where strategies for addressing class imbalance and accounting for the independent nature of astronomical observations were pivotal for constructing a robust predictive model.

10.2. Graph Model with Explanation

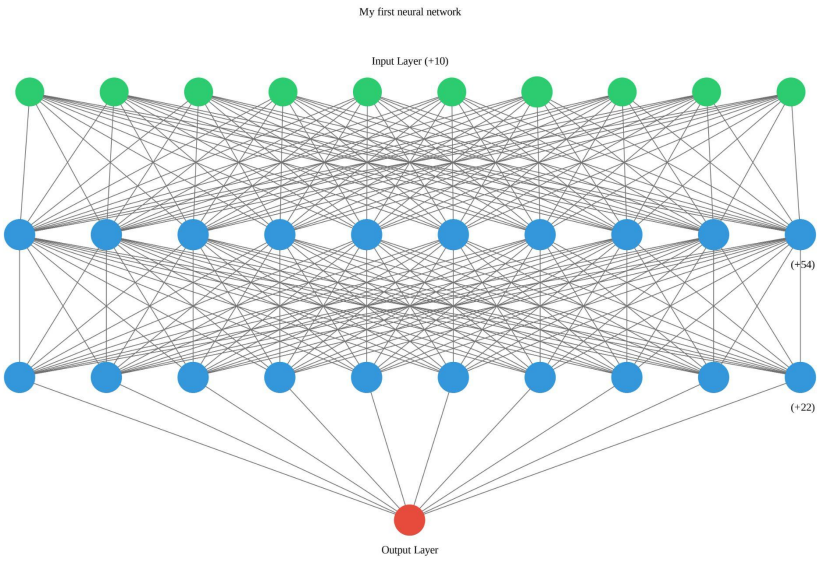


Figure 6. Graph Model

- In the graphical depiction, the Input Layer is illustrated with 64 neurons, each represented by a node. These nodes signify the input features of the model. The activation function used for these neurons is Rectified Linear Unit (ReLU), denoted as "ReLU" in the representation. ReLU introduces non-linearity to the model, allowing it to capture complex patterns in the data.
- The next layer, referred to as the Hidden Layer, consists of 32 neurons, each again depicted by a node. These neurons apply the ReLU activation function, serving as intermediaries for transforming the input data. The hidden layer is essential for the model to learn intricate patterns and relationships within the data.
- Finally, the Output Layer is represented by a single node. This node utilizes the Sigmoid activation function, denoted as "Sigmoid" in the graphical representation. The Sigmoid function is commonly employed in binary classification tasks as it squashes the output between 0 and 1, effectively providing a probability-like score.

The interconnected nodes and layers signify the flow of information through the neural network. Each connection between nodes represents a weighted connection, indicating the strength of the relationship. During training, the model adjusts these weights to minimize the difference between predicted and actual outputs. The graphical representation captures the essence of the model's architecture, showcasing its ability to process input data through layers of interconnected neurons with specific activation functions.

11. Implementation

11.1. Pseudocode

The following pseudocode provides a high-level overview of the steps involved in the project.

```
# Exoplanet Detection Pseudocode

# Step 1: Initialize Project Setup
Install necessary libraries
Import required modules

# Step 2: Loading Dataset into Pandas DataFrame
exoplanet_data = Read CSV data from file
Display exoplanet_data

# Step 3: Exploratory Data Analysis
# Step 3.1: Initial Data Analysis and Feature Engineering
Label encode the target feature
Display dataset shape and description
Check for missing values

# Step 3.2: Exploring the Distribution of Classes
Display and visualize class distribution

# Step 3.3: Observing Correlational Patterns
Display heatmap of correlation matrix

# Step 3.4: Investigating Flux
Visualize flux variations for non-exoplanet and exoplanet stars

# Step 4: Data Preprocessing
# Step 4.1: Handling Missing Values
Check and visualize missing values
Handle missing values if needed

# Step 4.2: Outlier Detection and Removal
Visualize and remove outliers in flux readings

# Step 4.3: Handling Imbalance using SMOTE
Apply SMOTE to balance the class distribution

# Step 4.4: Data Normalization
Normalize input features

# Step 4.5: Apply Gaussian Filters
Smooth flux fluctuations using Gaussian filters

# Step 4.6: Feature Scaling
Scale features using StandardScaler

# Step 4.7: Dimensionality Reduction using PCA
Apply PCA for dimensionality reduction
```

```
# Step 4.8: Splitting into Testing Data and Training Data
Split data into training and testing sets

# Step 5: Model Building and Evaluation
# Step 5.1: K Nearest Neighbors
Build and evaluate KNN model

# Step 5.2: Logistic Regression
Build and evaluate Logistic Regression model

# Step 5.3: Bernoulli Naive Bayes
Build and evaluate Bernoulli Naive Bayes model

# Step 5.4: Decision Tree Algorithm
Build and evaluate Decision Tree model

# Step 5.5: Random Forest Algorithm
Build and evaluate Random Forest model

# Step 5.6: XGBoost Classifier
Build and evaluate XGBoost model

# Step 5.7: Artificial Neural Network
Build and evaluate Neural Network model

# Step 5.7.1: XAI - Explaining ANN Predictions with LIME
Explain predictions of ANN using LIME

# Step 6: Model Analysis and Conclusion
# Display validation accuracy for each model
Conclude the project and discuss findings
```

11.2. Implementation Details

- In the context of addressing the inherent challenge of class imbalance in binary classification scenarios, particularly those involving a significant underrepresentation of one class, the Synthetic Minority Over-sampling Technique (**SMOTE**) is a pivotal strategy.
- Deployed through the imblearn library, SMOTE systematically generates synthetic instances of the minority class, thereby rectifying imbalances in class distribution. This augmentation contributes to a more equitable representation of both classes, fostering improved model training and performance.
- To enhance the interpretability of the dataset's flux readings, **Gaussian filters**, rooted in the Gaussian distribution, are systematically applied. These filters, integral in image processing and data analysis, serve to smooth fluctuations and accentuate underlying patterns in light intensity curves.
- This proves particularly advantageous for discerning periodic patterns associated with exoplanet transits, thereby facilitating more accurate model predictions.

- Following the preprocessing steps, including SMOTE and Gaussian filtering, attention is directed towards feature scaling through standardization.
- This crucial step ensures uniformity in feature scales, mitigating the risk of certain features exerting undue influence over others during subsequent machine learning model training.

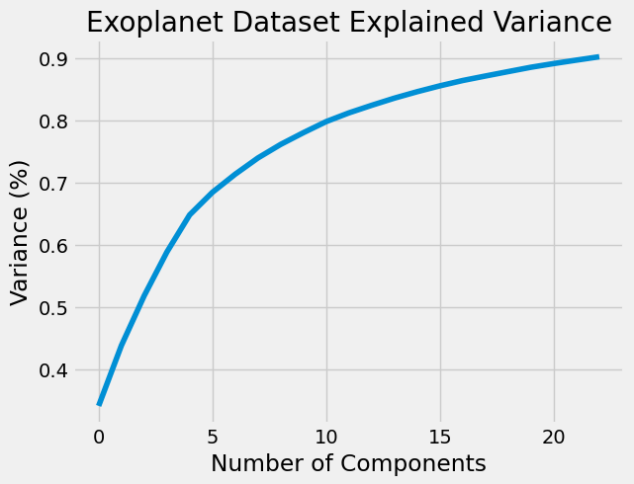


Figure 6. Exoplanet Dataset Explained Variance

- Subsequent to feature scaling, dimensionality reduction takes place using Principal Component Analysis (**PCA**), a widely adopted technique in machine learning and statistics. PCA transforms the original feature set into uncorrelated variables known as principal components, facilitating a reduction in dimensionality while retaining critical data variance.
- The determination of the optimal number of principal components is guided by the goal of retaining 90% of the dataset's variance, striking a balance between information retention and dimensionality reduction.
- The study then progresses to the application of various machine learning classifiers, including k-Nearest Neighbors (KNN), Logistic Regression, Bernoullli Naive Bayes, Decision Tree, Random Forest, and XGBoost.
    - **k-Nearest Neighbors (KNN)**: Utilizes proximity to k-nearest data points to classify new instances, making decisions based on the majority class within its neighbors.
    - **Logistic Regression:** A linear model suitable for binary classification, logistic regression estimates the probability of an instance belonging to a particular class.
    - **Bernoulli Naive Bayes:** Assumes independence between features, employing Bayes' theorem to calculate probabilities and make predictions efficiently.
    - **Decision Tree:** A tree-like model that recursively splits data based on features, making decisions through a series of binary choices, ultimately leading to a classification.
    - **Random Forest:** An ensemble of decision trees, each trained on a random subset of data, collectively contributing to a robust and accurate classification.

    - **XGBoost:** An optimized gradient boosting algorithm that sequentially builds a series of weak learners, aiming to correct errors from preceding models and improve overall predictive performance.
    - **Artificial Neural Network (ANN):** A computational paradigm modeled after the architecture of biological neural networks, comprising interconnected nodes arranged in layers, designed to perform sophisticated pattern recognition and machine learning tasks through the activation of weighted connections.
- A unified evaluation function is systematically defined to comprehensively assess the models' performance on a validation set, providing metrics such as accuracy, precision, recall, and F1-score. The outcomes are meticulously presented, affording a holistic analysis of each model's efficacy in handling the intricacies of the given dataset.
- In the pursuit of enhancing the efficacy of machine learning models for exoplanet detection within the given astronomical dataset, the application of Gaussian filters emerges as a pivotal preprocessing step with implicit implications for outlier management.
- The use of Gaussian filters indirectly contributes to the reduction of outlier influence. This is achieved through the filters' capacity to smooth out irregular fluctuations and accentuate inherent patterns in the flux readings.
- The Gaussian filters, rooted in probability theory and applied to the context of image processing and data analysis, serve as a valuable tool for noise reduction. By emphasizing underlying patterns while mitigating the impact of isolated extreme values, these filters play a crucial role in fostering a more accurate representation of the intrinsic characteristics of the astronomical data.
- While Gaussian filters address outliers indirectly, a more explicit consideration of outlier removal tailored to the unique characteristics of astronomical data would fortify the preprocessing pipeline.

In summary, this systematic approach encompasses the strategic mitigation of class imbalance, enhancement of feature interpretability, and optimization of model performance, collectively contributing to the robustness of binary classification models in the realm of exoplanet detection. This multifaceted methodology aligns with contemporary research standards, ensuring a rigorous and comprehensive exploration of the dataset and the subsequent model outcomes.

12. Results

Following the analysis of the second increment, the outcomes are as follows: All models displayed exemplary results in terms of accuracy, precision, recall, and F1-score, showcasing robust predictive capabilities. K Nearest Neighbors, Logistic Regression, Decision Tree, Random Forest, and XGBoost all showcased accuracy levels approaching 99.97%. Although Bernoulli Naive Bayes achieved a slightly lower accuracy of 98.89%, it nonetheless demonstrated

commendable performance. The corresponding code performance screenshots are provided below for reference.

Performance report:

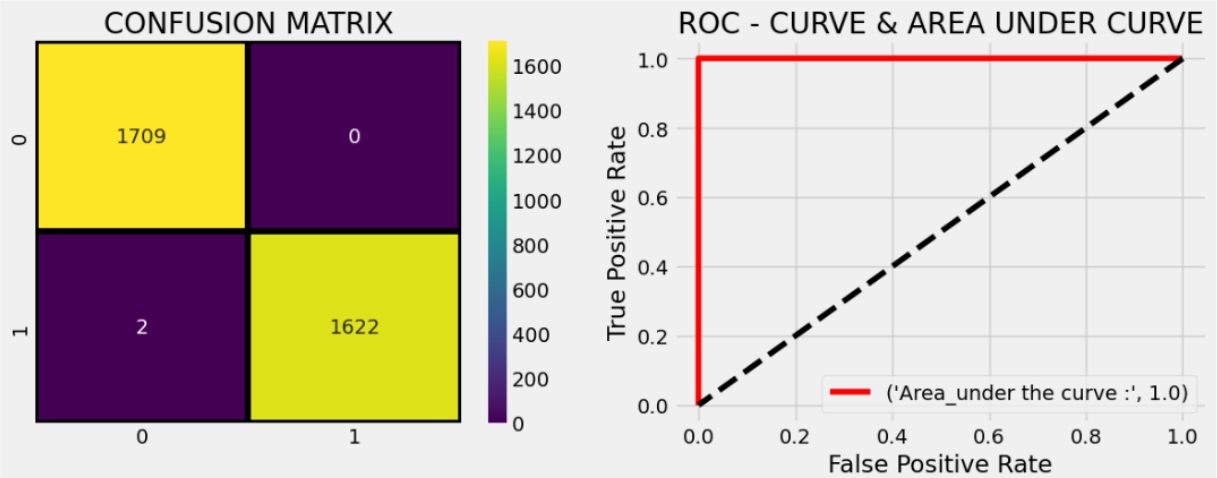
- **K Nearest Neighbors**

```
Validation accuracy of model is 0.9993999399939995

Classification report :
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      1709
           1       1.00      1.00      1.00      1624

    accuracy                           1.00      3333
   macro avg       1.00      1.00      1.00      3333
weighted avg       1.00      1.00      1.00      3333
```



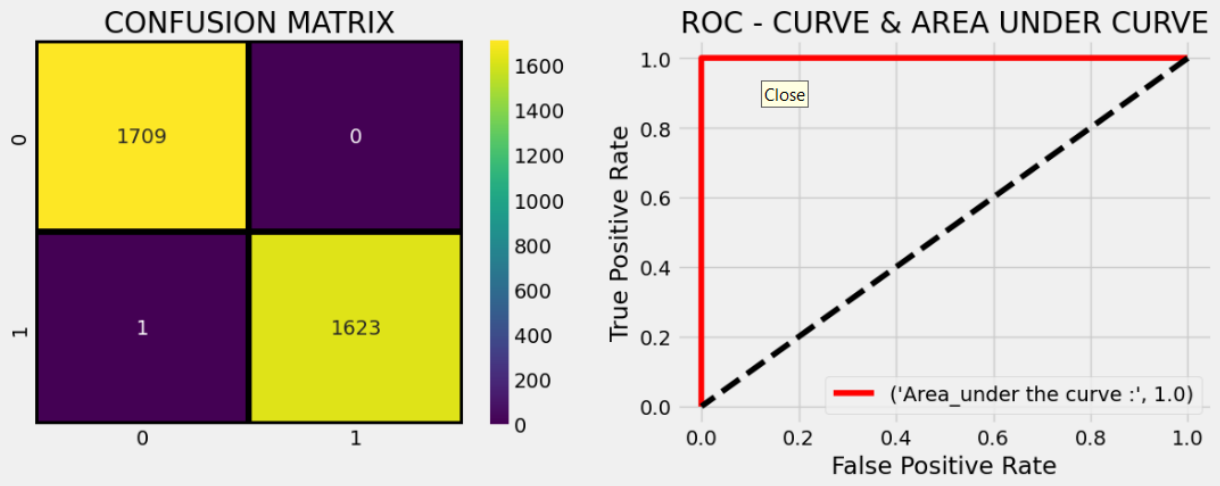
- **Logistic Regression**

```
Validation accuracy of model is 0.9996999699969997

Classification report :
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      1709
           1       1.00      1.00      1.00      1624

    accuracy                           1.00      3333
   macro avg       1.00      1.00      1.00      3333
weighted avg       1.00      1.00      1.00      3333
```



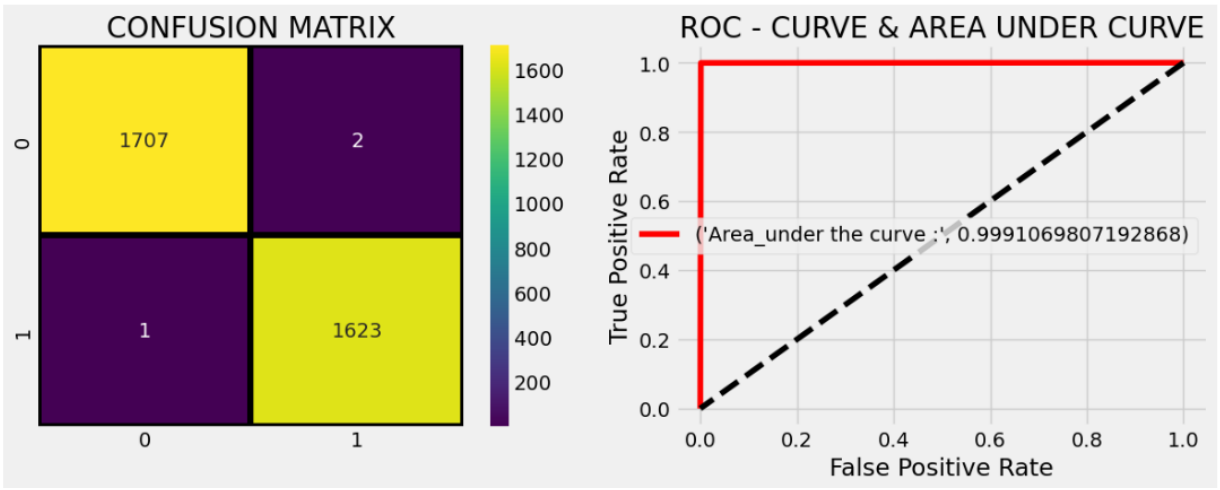
- **Decision Tree**

```
Validation accuracy of model is 0.9990999099909991

Classification report :
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      1709
           1       1.00      1.00      1.00      1624

    accuracy                           1.00      3333
   macro avg       1.00      1.00      1.00      3333
weighted avg       1.00      1.00      1.00      3333
```



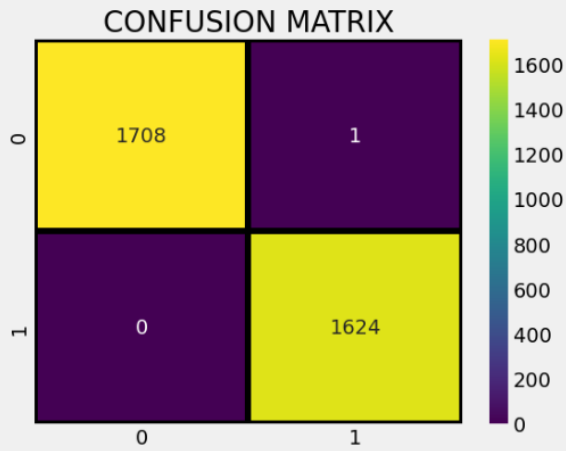
- **Random Forest**

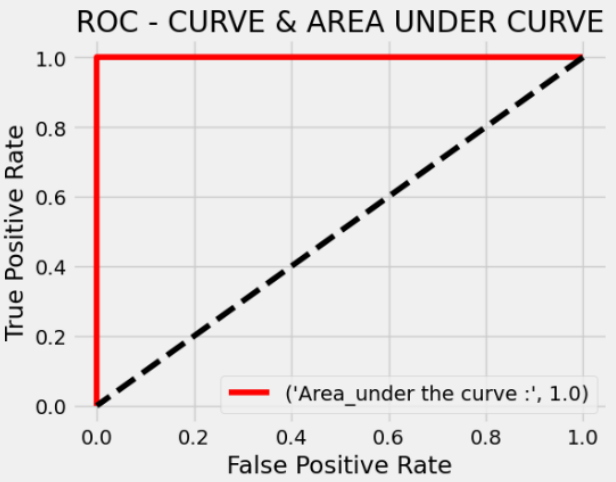
```
Validation accuracy of model is 0.9996999699969997

Classification report :
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      1709
           1       1.00      1.00      1.00      1624

    accuracy                           1.00      3333
   macro avg       1.00      1.00      1.00      3333
weighted avg       1.00      1.00      1.00      3333
```





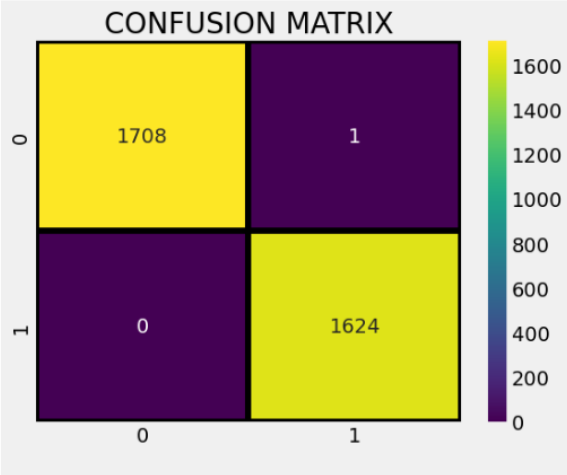
- **XGBoost**

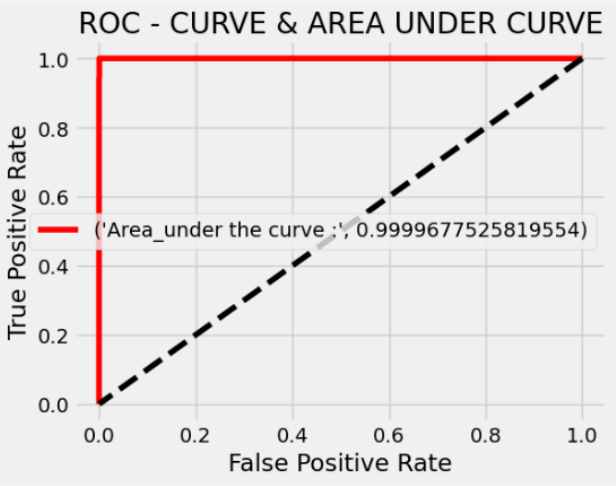
```
Validation accuracy of model is 0.9996999699969997

Classification report :
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      1709
           1       1.00      1.00      1.00      1624

    accuracy                           1.00      3333
   macro avg       1.00      1.00      1.00      3333
weighted avg       1.00      1.00      1.00      3333
```





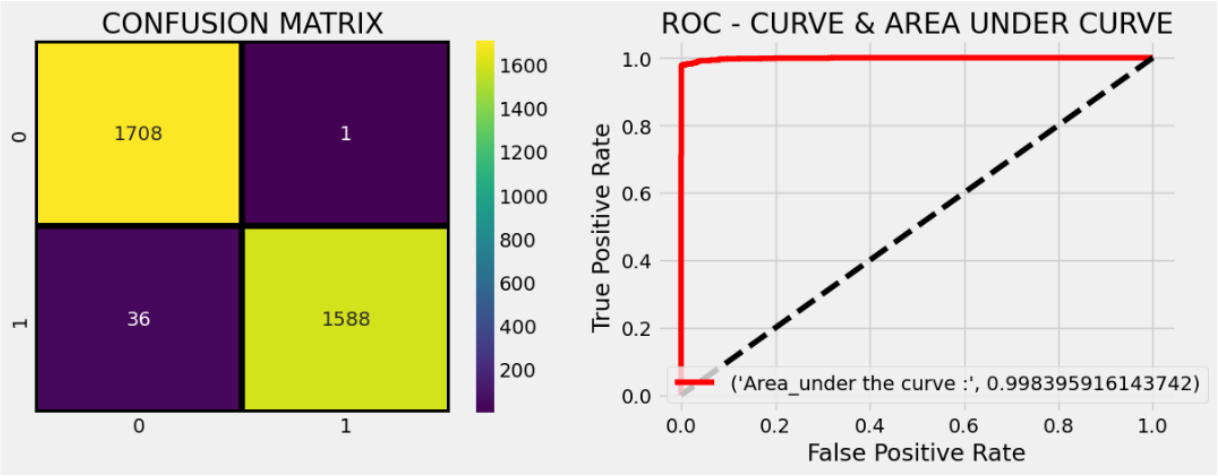
- **Bernoulli Naive Bayes**

```
Validation accuracy of model is 0.9888988898889889

Classification report :
              precision    recall  f1-score   support

           0       0.98      1.00      0.99      1709
           1       1.00      0.98      0.99      1624

    accuracy                           0.99      3333
   macro avg       0.99      0.99      0.99      3333
weighted avg       0.99      0.99      0.99      3333
```



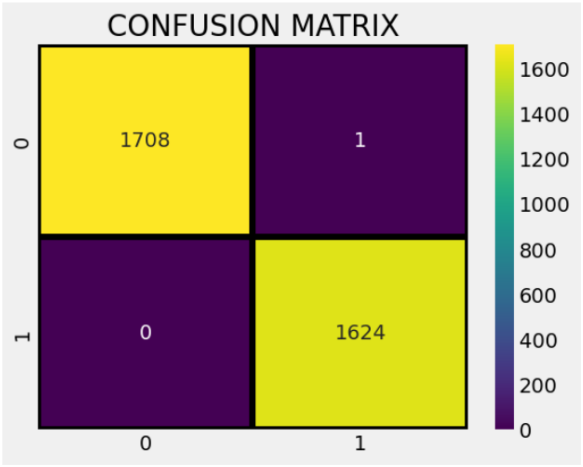
- **Artificial Neural Network**

```
Classification report :
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      1709
           1       1.00      1.00      1.00      1624

    accuracy                           1.00      3333
   macro avg       1.00      1.00      1.00      3333
weighted avg       1.00      1.00      1.00      3333

Text(0.5, 1.0, 'CONFUSION MATRIX')
```



13. Project Management

Implementation Status after increment 2 is documented below:

13.1. Work Completed in Increment 1

- Pre-processing and Feature Engineering:
    - Description: The team successfully performed pre-processing and feature engineering tasks on the dataset.
    - Responsibility: Shaik Ayesha Mohsin
    - Contributions: Shaik Ayesha Mohsin (100%)
- K-Nearest Neighbors (KNN) Model:
    - Description: The KNN model has been implemented and integrated into the project.
    - Responsibility: Shaik Ayesha Mohsin
    - Contributions: Shaik Ayesha Mohsin (100%)
- Logistic Regression Classifier (LRC):
    - Description: The Logistic Regression Classifier has been implemented and integrated into the project.
    - Responsibility: Shaik Ayesha Mohsin
    - Contributions: Shaik Ayesha Mohsin (100%)
- Decision Tree Classifier (DTC):
    - Description: The Decision Tree Classifier has been implemented and integrated into the project.
    - Responsibility: Areeba Hassan
    - Contributions: Areeba Hassan (100%)
- Bernoulli Naive Bayes:
    - Description: The Bernoulli Naive Bayes model has been implemented and integrated into the project.
    - Responsibility: Areeba Hassan
    - Contributions: Areeba Hassan (100%)
- Random Forest Classifier (RFC):
    - Description: The Random Forest Classifier has been implemented and integrated into the project.
    - Responsibility: Faisal Mohammed
    - Contributions: Faisal Mohammed (100%)
- XGBoost (XGB):
    - Description: XGBoost has been implemented and integrated into the project.
    - Responsibility: Faisal Mohammed
    - Contributions: Faisal Mohammed (100%)
- Report Writing:
    - Description: Documentation and reporting for respective models have been completed.

- Responsibility: Shaik Ayesha Mohsin (Detailed design of features, Preliminary results,Project Management), Areeba Hassan (Dataset Description,Abstract), Faisal Mohammed (Analysis, Implementation)
- Contributions: Shaik Ayesha Mohsin (33.33%), Areeba Hassan (33.33%), Faisal Mohammed (33.33%).

13.2. Work Completed in Increment 2

- Implemented Deep Learning Models: Artificial Neural Network (ANN):
  - Description: Implementing and integrating Artificial Neural Network models into the project. Added graphs and architecture diagrams for better visualization.
  - Responsibility: All team members
  - Issues/Concerns: None reported.
- Explainable Artificial Intelligence (XAI) for Interpretability:
  - Description: Incorporated XAI techniques like SHAP and LIME to enhance ANN model interpretability. Gained insights into the model's predictions at both local (individual instances) and global (entire dataset) levels, enhancing interpretability and understanding of the ANN model applied to exoplanet detection.
  - Responsibility: All team members
  - Issues/Concerns: None reported.

13.2.1 Task Assignment breakdown

- Artificial Neural Network (ANN) and XAI methods: LIME and SHAP:
  - Description: The ANN has been implemented and integrated into the project. XAI techniques like SHAP and LIME were interpreted to enhance model interpretability.
  - Responsibility: Shaik Ayesha Mohsin
  - Contributions: Shaik Ayesha Mohsin (100%)
- Plot Architecture Diagram for the ANN model:
  - Description: Architecture diagram for the ANN had been implemented and integrated into the project and the report.
  - Responsibility: Areeba Hassan
  - Contributions: Areeba Hassan (100%)
- Plot Graph Diagram for the ANN model:
  - Description: Graph diagram for the ANN had been implemented and integrated into the project and the report.
  - Responsibility Faisal Mohammed
  - Contributions: Faisal Mohammed (100%)
- Report Writing:
  - Description: Documentation and reporting for respective models have been completed.
  - Contributions: Shaik Ayesha Mohsin (33.33%), Areeba Hassan (33.33%), Faisal Mohammed (33.33%).

The project has made significant progress with the successful implementation of various machine learning models. The team has completed the tasks planned for increment 2, including the implementation of Artificial Neural Network models and the incorporation of Explainable Artificial Intelligence (XAI) for improved interpretability. Team members have been actively contributing to their respective tasks, and collaboration has been effective in ensuring a comprehensive approach to model development and documentation. No major issues or concerns have been reported at this stage of the project.

14. References

[1] Ofman, L., Averbuch, A., Shliselberg, A., Benaun, I., Segev, D., & Rissman, A. G. (2022, February 1). *Automated identification of transiting exoplanet candidates in NASA Transiting Exoplanets Survey Satellite (TESS) data with machine learning methods*. New Astronomy; Elsevier BV. https://doi.org/10.1016/j.newast.2021.101693

[2] Malik, A., Moster, B. P., & Obermeier, C. (2021, December 21). Exoplanet detection using machine learning. Monthly Notices of the Royal Astronomical Society; Oxford University Press. https://doi.org/10.1093/mnras/stab3692