EXOPLANET DETECTION BY TRANSIT METHOD

MSc Research Project
Data Analytics

Arnab Hati
Student ID: x22107321`

School of Computing
National College of Ireland

Supervisor: Christian Horn

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: ……. ………Arnab Hati……………………………………………………………………………………

Student ID: ………………x22107321……………………………………………………………………..…………

Programme: …………… Data Analytics…………………………… **Year:** ………2023………………..

Module: …………… MSc Research Project……………………………………….……………………

Supervisor: …………… Christian Horn ………………………………………………………….………………

Submission Due Date: ………………14/12/2023……………………………………………………………………….………

Project Title: ……… EXOPLANET DETECTION BY TRANSIT METHOD …………………….………

Word Count: ………6027……………………… **Page Count**……………………20…………………….……..

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……Arnab Hati………………………………………………………………………………

Date: ………14th of December 2023……………………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

EXOPLANET DETECTION BY TRANSIT METHOD

Arnab Hati
x22107321

Abstract

The detection of exoplanets is essential for understanding the diversity of the universe, habitability opportunities, and the possibility of alien life outside our solar system. Extraterrestrial life has been the focus of extensive research for decades. To detect exoplanets, many machine learning and deep learning approaches have provided important predictions. The transit method or observation is responsible for the variations in a star's spectrum caused by an orbiting planet's gravitational pull. To improve these predictions, this research concentrates on the implementation of machine learning and the deep learning model after the feature selection technique is applied. Two machine learning models (XGboost, Catboost) and three deep learning models (RNN, Variableational Encoder, GRU) were implemented. Once the best feature was selected to improve the overall performance, Catboost outperformed the other machine learning and deep learning models by 99.98%.

1 Introduction

1.1 Background and Motivation

An exoplanet is a planet that is found outside of our solar system. The process of finding an exoplanet is known as exoplanet detection. There have been significant advances in the field of exoplanet detection, which has resulted in the detection of thousands of exoplanets using detection methods. According to the theory that seven earth-sized planets orbit around a single star, this system is called Trappist -1 system. An exoplanet contributes a lot to the understanding of the planetary system (Fig. 1).

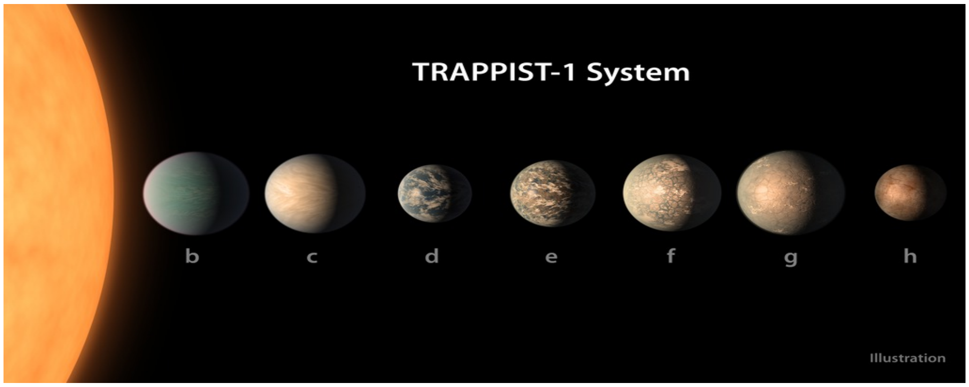


Figure 1: [1]Trappist-1 System

[1] https://www.spitzer.caltech.edu/image/ssc2018-04b1-trappist-1-planet-lineup-feb-2018

The discovery of the seven planets that orbit around the dwarf star, Trappist- 1, has made the study of exoplanets even more important. According to a NASA report (2022), most of the planets are discovered through indirect methods that measure the decrease in a star's brightness when an exoplanet passes directly in front of the star. This method is called the transit method, and it measures the variations in the star's spectrum caused by the planet's gravity.

Transit presents an "exoplanet" that slightly dims the star's light by passing the planet in front of the star. This dimming is observed in little curve graphs which show light-receiving operation over a long-lasting time. The transit can assist in addressing various exoplanet characteristics and highlighting the exoplanet's atmosphere Hence, the main objective of this study is to investigate multiple mechanisms for exoplanet detection and to evaluate model performance with the help of the Kepler dataset by looking at relevant research papers.

1.2 Research Problems

The transit method of exoplanet detection faces difficulties in optimizing model performance and selecting features from the Kepler data set. Current methods need to be evaluated. One of the most important challenges is the variability of accuracy between machine learning and deep learning models. The challenge of finding the most efficient feature selection technique is a major issue that hinders the development of a robust exoplanet detection model. Addressing this challenge is essential for understanding exoplanetary systems better and improving the precision of detection models, leading to progress in astrophysics and space research.

The purpose of this study is to resolve and analyze the following research query:
How does feature-based analysis affect exoplanet detection accuracy, with a focus on CatBoost and XGBoost? Compare Deep Learning models (RNNs, Variational Encoders, GRUs) for heterogeneous architectures.

1.3 Objective

This research aims to explore different factors related to exoplanet detection and compare the performance of different models on the Kepler dataset. To determine the impact of feature selection and improve the performance of machine learning and deep learning models in exoplanet detection.

2 Related Work

The study conducted by Al-Mamun *et al.* (2023) evaluated the historical track of 'exoplanet finding'. The exoplanet is considered a planet that secures its place outside the solar system. The paper highlighted the major role in recognizing people’s understanding of the ‘exoplanet systems’ that demonstrate planets that move around other stars. It is observable that all of the planets are rotating around the prime star: the sun. The system is difficult to view with a telescope straightly. The paper approached an innovative model called the ‘Life Convolutional Neutral Networks (LCNN)’ model which addresses exoplanet detection by involving the use of ‘The Kepler dataset’. In machine learning, especially in ‘Convolutional Neutral Networks’ the study depicted a crucial progression toward automated, structured, and accurate exoplanet detection with a land of exoplanet studies. The LCNN structure states an extraordinary function that can achieve a training ability of 76.92% along with a testing accuracy of 99.12 %. The different accuracy highlights successful models and agrees to identify a reliable exoplanet. The paper not only enhances the seize of exoplanetary history but also understands the metaphoric ability of machine learning in expanding the colossal exoplanet system.

Following Tu *et al*. (2022) it can be stated that Convolutional Neutral Networks have been organised on 15,638 superflares on solar-system stars which originated from the three years of ‘Transiting Exoplanet Survey Satellite’. The TESS defines a space investigation platform that is designed to observe exoplanets in orbit about 200,000 near stars with a specific interest in analyzing small planets. These three networks are utilized to place the visual observation which helps to search for superflares and eliminate the false-positive events in current periods. The paper analyzed ‘stellar light curves’ in observing super flare signals. ‘TESS-pixel level of data’ helps to identify the superflares.

According to the study by Sharma *et al*., (2023). it has highlighted that observation of planets can sustain life which occurs at a new level in ‘NASA’s Keplar goal’. The study's main goal is to introduce approximately 4000 planets in the solar system. Moreover, the task of evaluating data is considered quite difficult and laborious and has asked for more accurate methods in introducing new exoplanets by eliminating false positives and errors. The main focused goal is to involve the ‘machine learning’ algorithms to justify the stars along with exoplanets by collecting data from the Kepler satellite.

The literature by Malik et al. (2022) constructed a new ‘machine leading’ technique to identify exoplanets which is used in the transit process. This learning method helps to analyze various research areas. The paper aimed to focus on improving the ‘conventional algorithm’ which is incorporated in ‘astrophysics’ to observe the exoplanets. Extracting 789 features can gather information about the natural types of light curves. The study evaluated the method that anticipated a planet with an ‘AUC 0,948’ for ‘Keplar data’ and the percentage of 94.8 secured a higher rank for ‘true-plante signals’ than ‘non-planet signals’.

The goal of the paper by Salinas *et al*. (2023). is to analyze a large database of light curves by the ‘Transiting Exoplanet Survey Satellite (TESS)’. The ‘deep-learning’ method has been utilized to understand the transit signals of exoplanets directly. Though CNNs have some issues such as the requirement of many layers to hold dependencies on sequential data including ‘light-curves’ and ‘making networks’ which are not practical. A new architecture for ‘automation’ has been presented that is shaped to include the most important features of a ‘transit signal’ and ‘steller parameter’ through a ‘self-attention’ mechanism. Moreover, each element can be identified to differentiate a signal from a false positive.

According to Kumari. (2023) it has demonstrated that the existence of exoplanets has been found by ‘NASA’s Kepler Telescope’. The ‘computational data’ has depicted the identification of exoplanets from the signals, received by the ‘Keplar telescope’. In identifying the exoplanets the ‘residual networks’ of the ‘Keplar data’ have been used. The study also incorporated that ‘deep learning algorithms’ help to recognize the existence of exoplanets with less information. The CNN-oriented method is involved in addressing the categorization in a ‘low-data scenario’.

Chintarungruangchai *et al*. (2023) have analyzed that direct imaging can analyze several exoplanets which has a crucially important contribution to the origin of the planets. The method included in the study has adopted ‘Angular Differential Imaging (ADI)’ that drives a result with a large “Signal-to-Noice Ratio (SNR)’. This method requires an observational period from a ‘large telescope’ that is often over-consent. The study has likely revealed the plausibility of operating a ‘converter’ that is involved in expanding the SNR originated from an amount of ADI frameworks. Two-dimensional machine learning is present here to be tested. Besides that, the paper mainly focused on updating the ‘five-layer wide inference network’ using the ‘residual learning method’ and ‘batch normalization’ which can transform the observational data in the future by delivering the best result.

A Study cited by Priyadarshini and Puri (2021), mentioned that exoplanet detection is one of the most effective research studies. In the past, it has been seen that exoplanet has been detected by various conventional methods such as “transit method, direct imaging, radial velocity, astrometry” and many more. Therefore, the study with the help of machine algorithms able to detect machine learning. However, this study has incorporated another process to detect “exoplanet transit” through artificial intelligence. The performance of the learning is measured by various parameters such as accuracy, sensitivity and specificity.

After the analysis, the detection results had an accuracy of 99.62% which builds a sustainable importance in this field.

According to Iglesias Álvarez *et al*. (2023), "Machine Learning" can be considered as a solution regarding time reduction and computational cost reduction. These are required to evaluate a huge volume of light curves employing the transit method which is acquired from several different surveys to detect signals that are like transit. Detection of periodic dimming is involved in the technique in stellar light curves because of the existence of an orbiting exoplanet. The researchers created a trained 1D "Convolutional Neural Network" which is also tested with simulated light curves. This light curve imitates the outcome that is expected from the Kepler Space Telescope in the extended mission (K2). The light curve of the research considers several phenomena regarding variables of stellar including pulsation, flares, and rotations. These phenomena including stellar noise which are expected from data of K2, hampers the detection of transit signal as in real time data.

By Kaliraman *et al*. (2022), the "Box-fitting Least Squares (BLS)" technique, which is largely used in the discovery of exoplanets, created a huge amount of false positives that should be checked manually in the noisy data occurrence. An unbiased and automated technique for detecting exoplanets is crucial while mitigating false positive outcomes mimicking transiting signals of planets. An innovative mechanism based on a "convolutional neural network" to find exoplanets is introduced by employing the transit technique. SMOTE is employed to resample the information because the dataset is huge and imbalanced while the approach of expanding decay and techniques of early dropping are employed to mitigate overfitting the model. The model executes "Grid-SearchCV" for finetuning hyperparameters. Finally, the model is examined by employing "k-fold cross-validation" to establish a full model. In this study, specific performance criteria are used including precision, accuracy, recall, "f1 score", specificity and sensitivity. The research concluded after data analysis that the "convolutional neural network" created a maximum of 99.6% accuracy on the data testing.

According to Olmschenk *et al*. (2021) the "Transiting Exoplanet Survey Satellite (TESS)" assignment calculated starlight in ~75% of the sky across its primary mission of 2 years. This leads to numerous "30-minute cadence light curves" from TESS for analysis in the transiting exoplanet discovery. The researchers aim to serve an approach to search this huge dataset for transit signal which are both efficient computationally and delivers highly performant forecasting. This particular approach does not require as much effort of human search as it is supposed to. The researchers present a "convolutional neural network" which is trained to recognise signals of planetary transit and remove false positives. The network of this study requires no previous transit variable recognized through other frameworks for predicting a provided light curve. This network performs assumption on a "TESS 30-minute-cadence light curve" in ~5 ms on a "single GPU" which enables large-scale archival searches. The paper presents 181 new candidates for earth-like planets by the network presented in this study. The "neural network" model is provided additionally as an open source of codes that are available for public use and extension.

Two machine learning methods are studied in this research by Tiensuu *et al*. (2019), including "Support Vector Machine" and "Convolutional Neural Network" to select the best-performing model on a dataset that contains light intensity time series from extrasolar stars. The main complexity in the dataset is that there are a lot of exoplanet stars than exoplanet orbited stars. This is led to the presumed dataset which is enhanced by mirroring the star curves with an exoplanet which is orbiting and including them to the dataset in this context. Some techniques are done before incorporating the methods in the set of data in an attempt to further improve the outcomes. "Feature extraction" and "Fourier transform" are important measures of the time series for the SVM but preprocessing of the further alternatives is examined. The time series are smoothed and detrended for the CNN method, providing two inputs regarding the same light curves and all code is incorporated in "Python".

Jara-Maldonado *et al*. (2020) state that data scientists are encouraged by such Spatial Missions as the "Transiting Exoplanet Survey Satellite (TESS)" mission and the "Kepler" mission to explore datasets of light curves. These data analyses help to seek transit planets which aim to discover and validate exoplanets which refers to the planets that are discovered beyond the solar system. Exoplanet transits can be distinguished by the availability of light curves and radial velocity. Examination of these datasets manually is a job that necessitates huge quantities of effort and time and also tends to have errors. The implementation of "machine learning" models consequently has become more familiar in the research of the discovery and characterization of exoplanets. This study provides an analysis of the algorithms that are based on "machine learning" on the discovery of different transit exoplanets.

According to Osborn *et al*. (2020), an existing "neural network" model has shown the best performance on the TESS simulated data in this research, with an average of 97% or precision and accuracy of 92% on the planets in the two-class model.

3 Research Methodology

This project follows a basic approach of Knowledge Discovery in Databases (KDD) methodology, bearing in mind the goal of this study to determine how well the machine learning and the deep learning model can detect two different classes of exoplanets: false positives and confirmed from the Kepler's dataset. KDD methodology is a systematic approach to analyzing the raw data, extracting knowledge from the data, and building and fine-tuning a model with rigorous testing and performance assessment.

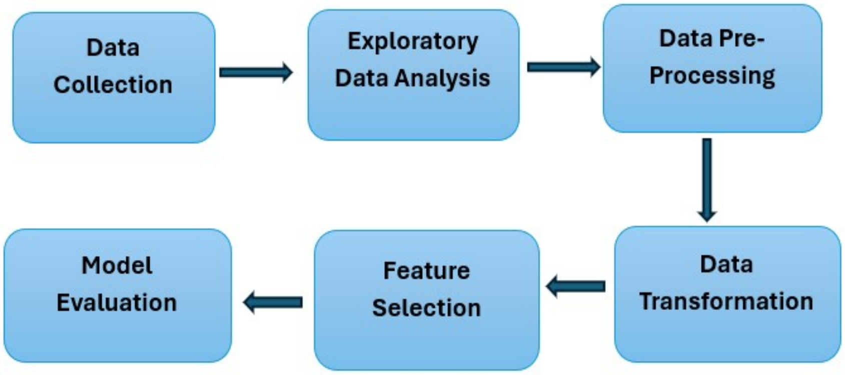
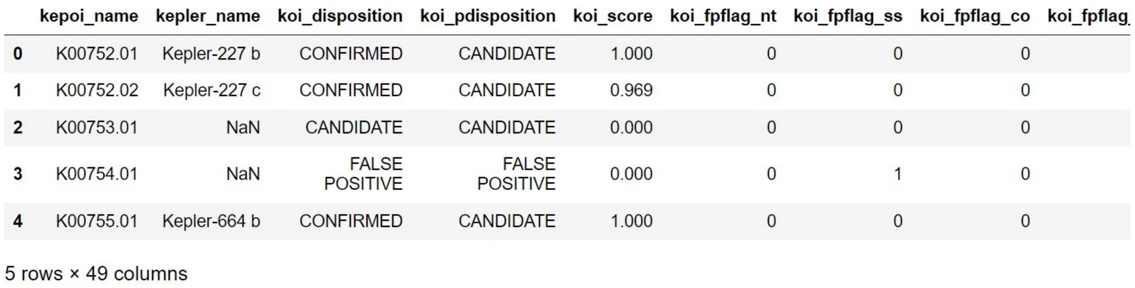


Figure 2: KDD Methodology for Exoplanet Detection

3.1 Data Collection and Description

The dataset used for this thesis is downloaded directly from the NASA Kepler website[2]. The dataset refers to the information collected by NASA's Kepler mission. This dataset contains 49 columns, and 9546 rows, and is in tabular format. The target variable for this project is koi-dposition, but other variables are also used as input variables.



	kepoi_name	kepler_name	koi_disposition	koi_pdisposition	koi_score	koi_fpflag_nt	koi_fpflag_ss	koi_fpflag_co	koi_fpflag
0	K00752.01	Kepler-227 b	CONFIRMED	CANDIDATE	1.000	0	0	0	
1	K00752.02	Kepler-227 c	CONFIRMED	CANDIDATE	0.969	0	0	0	
2	K00753.01	NaN	CANDIDATE	CANDIDATE	0.000	0	0	0	
3	K00754.01	NaN	FALSE POSITIVE	FALSE POSITIVE	0.000	0	1	0	
4	K00755.01	Kepler-664 b	CONFIRMED	CANDIDATE	1.000	0	0	0	

5 rows × 49 columns

Figure 3: Sample dataset

3.2 Exploratory Data Analysis

Exploratory data analysis (EDA) assists in understanding the characteristics of the data set. It visualizes and summarizes data. EDA reveals insights, evaluates data quality, and facilitates preprocessing. It finds missing values and correlations that help to select features. EDA helps to make modelling decisions, improves predictive accuracy, and provides meaningful interpretation. All in all, EDA is an important part of the data analysis process that helps to make informed decisions and sets the foundation for effective statistical

[2] https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=cumulative

modelling and machine learning. While analysing the dataset koi-disposition has three values i.e., false positive, confirmed, and candidate. These objects are classified as "candidate," meaning they are still in the process of being studied and have not yet been definitively identified as exoplanets. Osborn et al., (2020), found that an existing ‘neural network’ model performed best on simulated TESS, with an average for the planets in the 2-class model. In this project 2-class model is used i.e., FALSE POSITIVE and CONFIRMED.

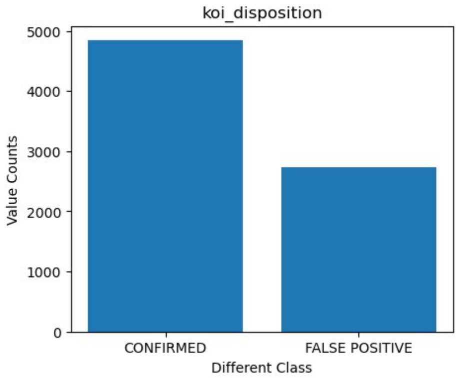


Figure 4: Frequencies of two class

Missing values in the data set can have a negative effect on the machine learning model. They can lead to inaccurate predictions, poor model performance, and distorted results. Managing missing values correctly by imputing or removing them is essential for maintaining model integrity and providing robust results in data-driven tasks. For each column, calculate the missing percentage. Drop columns with more than 80% of missing data. Remove rows with missing values. This improves the quality of the data and prepares the data for analysis or modelling. Evaluate and clean the dataset. These EDA and preprocessing actions ensure a clean dataset, which is essential for precise and meaningful downstream analysis or for machine learning applications.

Visualization improves the understanding of data by presenting complex data in a comprehensible form. Visualizations help to recognize correlations, anomalies, and distribution properties that you might miss in raw data. A list of column names containing numerical data types is subplot and iterated through numerical columns, producing histograms for each one relating to the ‘koi_disposition’ variable. The resultant plots show the distribution of numerical features, helping to visualize how they relate to Kepler objects disposition.

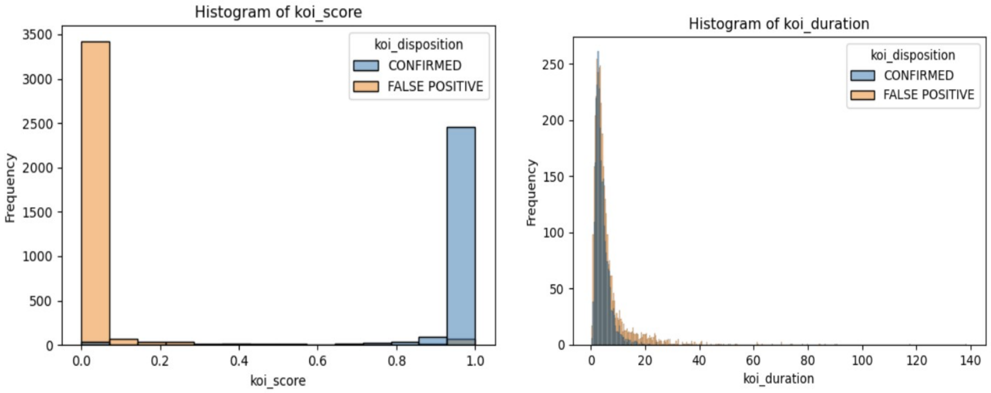


Figure 5: Histogram of different columns.

3.3 Data Pre-processing

In this project, it is the Anderson-Darling test that is applied to assess whether a sample has come from one particular distribution or a normal distribution. To verify that the sample is of a normal distribution population, Shapiro-Wilk tests are used. That test is to verify the invalid hypothesis that data came from a normal distribution. In both tests, all columns did not follow the Gaussian distribution. Various transformations, such as log, box-cox, exponential, etc. are used to compress or enlarge the data distribution. These transformations deal with skewness and make the distribution symmetric. The transformations aim at stabilizing variances, reducing outliers, and aligning the data with Gaussian distribution characteristics, which helps in statistical analysis and modeling assumptions. Many of the columns are normally distributed using various transformations as shown in the figure.

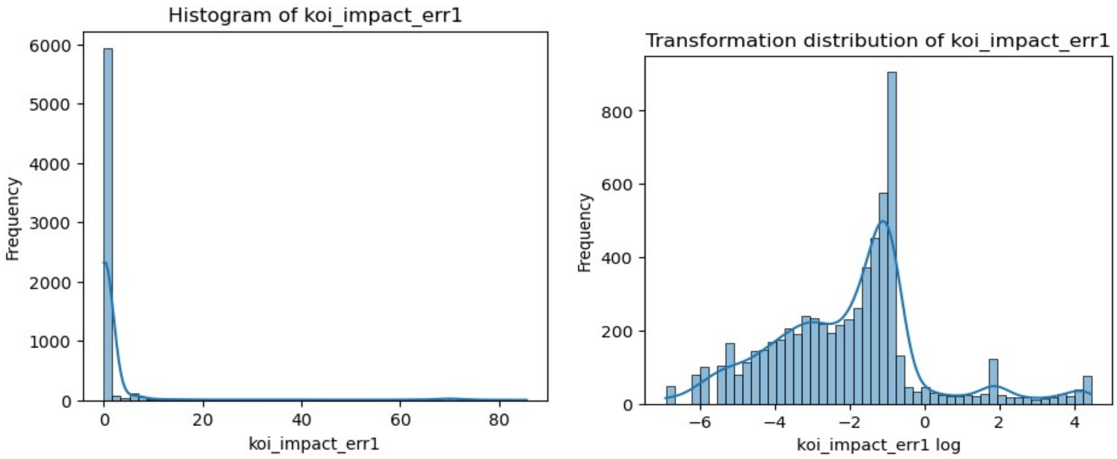
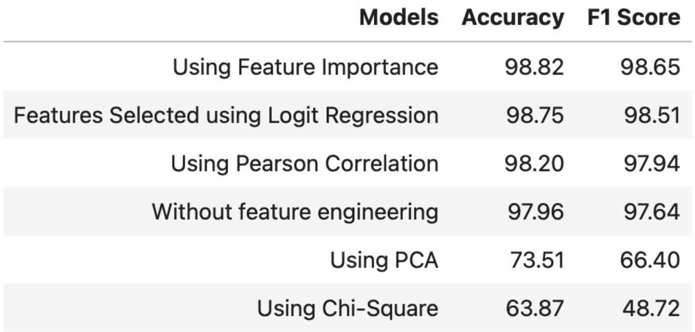


Figure 6: Data distribution applying log transformation.

Objective data types of Machine learning models may not be used because objective data types do not have a quantifiable metric or a numerical value. Therefore, machine learning models need numerical input for their calculations. Since objective data is subjective or qualitative, it does not have the numerical representation that algorithms need for making predictions or learning patterns efficiently. The “ra_str” and “dec_str” columns of the Kepler dataset are likely to represent Right Ascension and Declination Coordinates in string formats. Converting sexagesimal right ascension (RA) string to decimal degrees using this function and creating the new column “ra_deg” column. Extracted numerical components from the “dec_str” column representing decline (sexagesimal) format. Create intermediate columns for Degrees, Minutes, and seconds. Calculate the total decline in decimal degrees and create a new column “dec_deg”.

3.4 Feature Selection

Selecting the right features improves the performance of the model and improves its interpretability. Selecting relevant features and getting rid of irrelevant or redundant features reduces the dimensionality of the model, reduces the chance of overfitting, and improves the model’s generalization to the new data. Not only does this process speed up training, but it also makes the model more interpretable, easier to understand, and more trustworthy. Good feature selection helps to make machine learning models more accurate, more efficient, and more interpretable. This makes it easier to gain insights and make better decisions in different applications. Using Logit regression, feature selection involves finding and keeping the most important features to predict binary outcomes. This helps to improve model efficiency and improve model interpretability by choosing the most important variables. Using Pearson correlation, feature selection evaluates the linear relationship of features to the target variable and selects features with high correlation. This helps in modelling and helps in predictive accuracy. Using Feature importance techniques, find and retain the most important variables for the machine learning model. This helps in improving the predictive accuracy of the model and simplifies its structure. PCA stands for principal component analysis, which is the process of transforming data into non-correlated components. It finds and preserves the most informative features while reducing the dimensionality and preserving the variance. In this project, we used the Feature Importance Technique to prepare further machine learning and deep learning models. Using this technique, we achieved the highest accuracy and the F1 score as shown in figure 7 below.



Models	Accuracy	F1 Score
Using Feature Importance	98.82	98.65
Features Selected using Logit Regression	98.75	98.51
Using Pearson Correlation	98.20	97.94
Without feature engineering	97.96	97.64
Using PCA	73.51	66.40
Using Chi-Square	63.87	48.72

Figure 7: Technique used for feature selection.

3.5 Model Evaluation

All models are scored based on test accuracy after training the machine model and deep learning model; training and validation loss vs. epoch; training and validation accuracy vs. epoch; and accuracy in the classification report for each model. The accuracy and F1 score can be calculated using the equations (1) and (2)

$$Accuracy = \frac{TP+TN+FP+FN}{TP+TN} \qquad (1)$$

Where,
TP stands for True Positive.
TN stands for True Negative.
FP stands for False Positive.
FN stands for False Negative.

$$F1Score = \frac{2 \cdot Precision \cdot Recall}{Precision+Recall} \qquad (2)$$

Where Precision = $\frac{\text{TP}}{\text{TP+FP}}$

$$\text{Recall}= \frac{\text{TP+FN}}{\text{TP}}$$

4 Design Specification

A 3-tiered design framework, as illustrated in Fig, has been developed to carry out the proposed exoplanet classification research from the Kepler data set using machine learning and deep learning techniques. The data layer is where the pre-processed data from various sources is stored. In this layer, the open-source dataset from the NASA Kepler site is downloaded and uploaded to the Jupyter Lab for exploratory analysis, pre-processing, and transformation. The Business logic layer is where the final model is trained using two machines and three deep learning models, followed by the evaluation layer to evaluate the result based on various evaluation metrics.

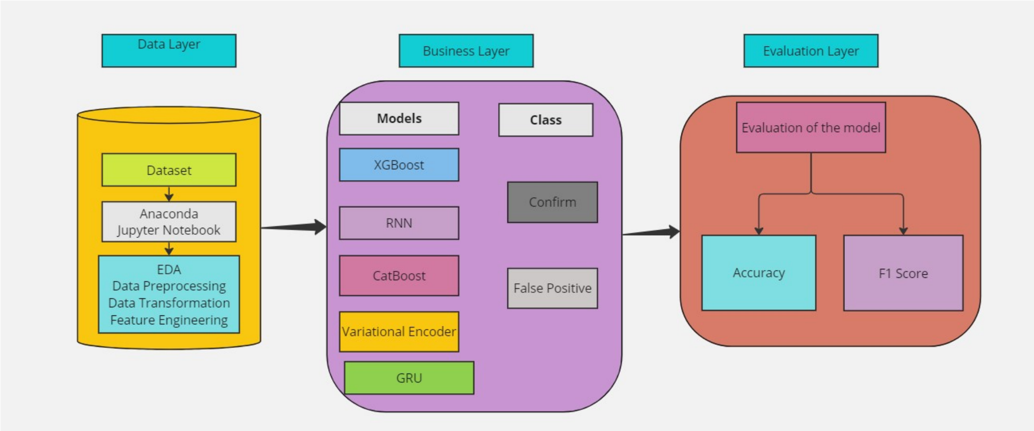


Figure 8: Three-tier Design Framework

5 Implementation

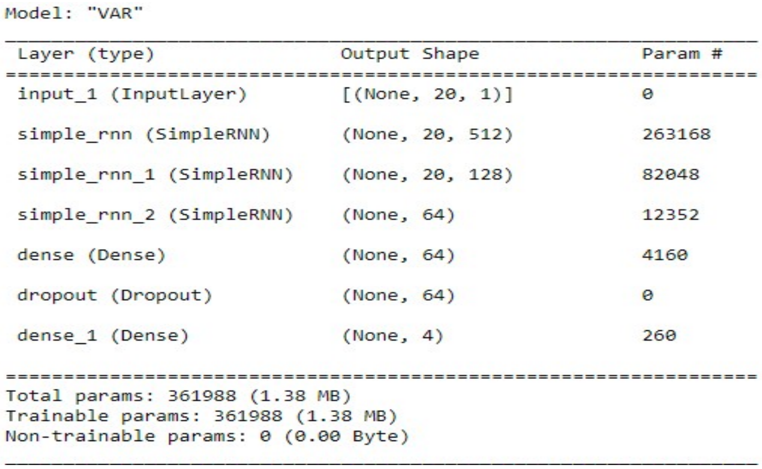
The entire proposal is implemented using the technologies shown in Fig.10 and discussed in the section below. The programming language used for this project is Python. Python was chosen due to its availability of many libraries, simplicity and consistency of programming, platform independence, and access to different machine learning frameworks. Anaconda is used as the base platform for the development of all models on the Jupyter Notebook. MatplotLib library is imported and used for experimental data analysis by generating visualizations for a better understanding of raw data. Keras API is the high-level API of the TensorFlow library. It is used to build all 3 deep learning models (variational encoder, recurrent NN, and Gated recurrent unit). XGBoost and CatBoost are all gradient boosting frameworks using xgboost and catboost respectively.

5.1 Implication of XGBoost and CatBoost

One of the most popular binary classification frameworks is xGBoost because of its high performance. It uses a gradient-boosted framework to combine weak learners in a strong model. XGBoost can handle imbalanced data sets, regularize data, and minimize overfitting. It has parallel processing capabilities to optimize training time. It supports various assessment metrics, which makes it useful for binary classification in various domains. CatBoost is a great binary classification tool with categorical properties. It can automatically handle categorical data encodings, reducing preprocessing. It also uses ordered boosting to improve predictive accuracy. CatBoost is resistant to overfitting and can handle imbalanced datasets. It also has robust GPU support, which makes it a preferred binary classification tool, particularly when dealing with data sets with categorical variables. XGBoast is binary classification-based. The data set is divided into training sets and testing sets with the help of the xGBoostclassifierclassifier() function. The classifier is created with a random state and trained on the train set using the fit function. The model makes predictions on the test set. The predictions' accuracy is counted and printed. CatBoost is binary-based. The catboostClassifier is created and trained on the test set using the fit method. The predictions made on the test set are counted and printed. Performance metrics for the catboost classifier are calculated and displayed.

5.2 Implementation of Variational Autoencoder

VAEs are mainly used in generative workflows and to capture complex distributions of data. They are not usually used in binary classification, but their latent representation learning and sample generation capabilities could be useful in some binary classification workflows where it is useful to understand data variability and generate new examples. The data is re-formed to meet the requirements of the SimpleRNN deep learning model. The x_train and x_test arrays get re-formed to 3D, adding a 3D dimension with dimension 1. A SimpleRNN-based VAR model is built using Keras, which consists of 3 SimpleRNN layer encoder configurations, compiled with Adam optimizer, sparse categorical crosstabs loss, and training performed using fit method on training data (x_train y_train) for 10 epochs with 20% validation split, model summary, training history shown for evaluation.



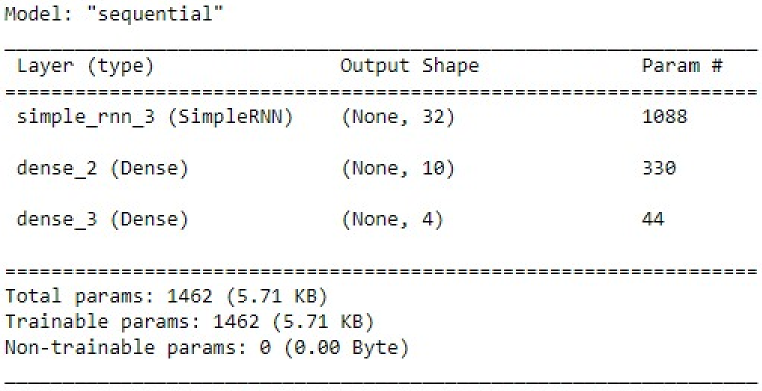
```
Model: "VAR"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 20, 1)]           0

 simple_rnn (SimpleRNN)      (None, 20, 512)           263168

 simple_rnn_1 (SimpleRNN)    (None, 20, 128)           82048

 simple_rnn_2 (SimpleRNN)    (None, 64)                12352

 dense (Dense)               (None, 64)                4160

 dropout (Dropout)           (None, 64)                0

 dense_1 (Dense)             (None, 4)                 260

=================================================================
Total params: 361988 (1.38 MB)
Trainable params: 361988 (1.38 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figur 9: Summary of Variational Autoencoder

5.3 Implementation of RNN

Recurrent neural networks (RNNs) are used in binary classification because they model the sequential dependencies in the data. RNNs process the input sequences, capture the time patterns, and make them suitable for the order of the data points in binary. In the code, created the Sequential model for binary classification in Keras using the RNN. The model consists of 32 units for the Simple RNN layer and 10 units for the Dense layer. The Simple RNN layer expects the input sequences to be the length of the data as defined by the data shape. The Dense layer expects the data to be the length as defined by the ReLU activation.

The final layer is the Dense layer expecting the data to be non-linear. This model has 4 units for the Sigmoid activation and the Sigmoid for the binary classification. Compiled the model with sparse categorization loss and accuracy metrics. Viewed the model summary to see the architecture, layers configurations, parameter counts, etc.

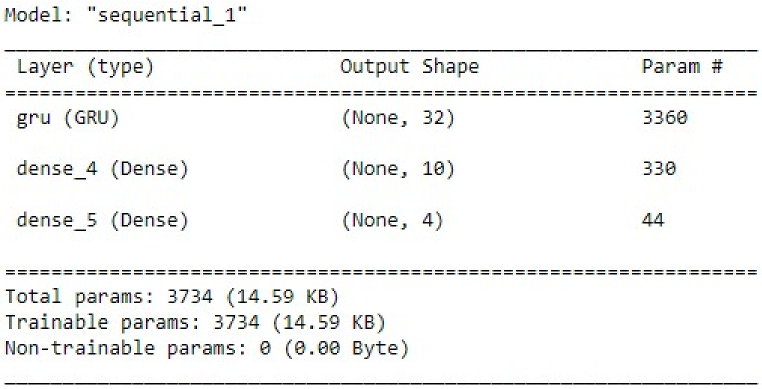


```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 simple_rnn_3 (SimpleRNN)    (None, 32)                1088

 dense_2 (Dense)             (None, 10)                330

 dense_3 (Dense)             (None, 4)                 44

=================================================================
Total params: 1462 (5.71 KB)
Trainable params: 1462 (5.71 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 10: Summary of the RNN model

5.4 Implementation of GRU

Gated Recurrent Units (GRUs) are recurrent neural networks (RNNs) that are designed to process sequential data. GRUs solve problems such as vanishing gradients that RNNs face. Because GRUs can capture sequential dependencies, they are well-suited for the binary classification of labeled data. GRUs model long-distance dependencies while minimizing vanishing gradient problems, making them well-suited for tasks such as binary classification where understanding of sequential patterns in input data is essential for making accurate predictions. Keras implemented a GRU model. GRU model consists of a 32-unit GRU layer processing input sequence defined by the shape of the data. Next, a 10-unit dense layer is added with ReLU activation, which introduces non-linearity. Finally, a 4-unit dense layer with Sigmoid activation is added with an appropriate binary classification function. GRU model is compiled using sparse categorization loss and accuracy metrics. The training set consists of 10-epoch training with a 20% validation split. Accuracy is visualized with Matplotlib, and the model is evaluated on a test set.



```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 gru (GRU)                   (None, 32)                3360

 dense_4 (Dense)             (None, 10)                330

 dense_5 (Dense)             (None, 4)                 44

=================================================================
Total params: 3734 (14.59 KB)
Trainable params: 3734 (14.59 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 11: Summary of GRU model

6 Evaluation

This section aims to provide a detailed review of the results and key findings of the study, as well as the consequences of these findings from an academic and practitioner perspective. Only the most pertinent results that support the research question and objectives are presented. Provide a thorough and rigorous review of the results. Use statistical tools to evaluate and evaluate the experimental research results and levels of significance.

6.1 Feature Importance Technique for Feature Selection

To optimize the feature selection for the Kepler dataset, I have used six different techniques, each of which uses logistic regression to prepare the model. The effectiveness of each technique has been rigorously tested, with an emphasis on accuracy and the F1 score metric. These techniques include: 'Without feature engineering' is a baseline approach in which no feature engineering has been performed. 'Pearson Correlation' is used for feature selection. Features selected using logit regression is highly accurate features (97.8056%) with F1 score (of 97.4312%) Lowest-accuracy features (74.6082%) with F1 (67.8571%) Chi-square statistical test Harshly accurate features (64.4984%) and F1 (48.5812%) Identify and select most influential features (99.0596%) with highest F1 score (98.8372%)

Decided to use this method for feature selection due to the high performance shown by Using Feature Importance. To provide a more accurate and robust representation of the Kepler dataset for further analysis and interpretation, these selected elements shall subsequently be used during the development of the primary model.

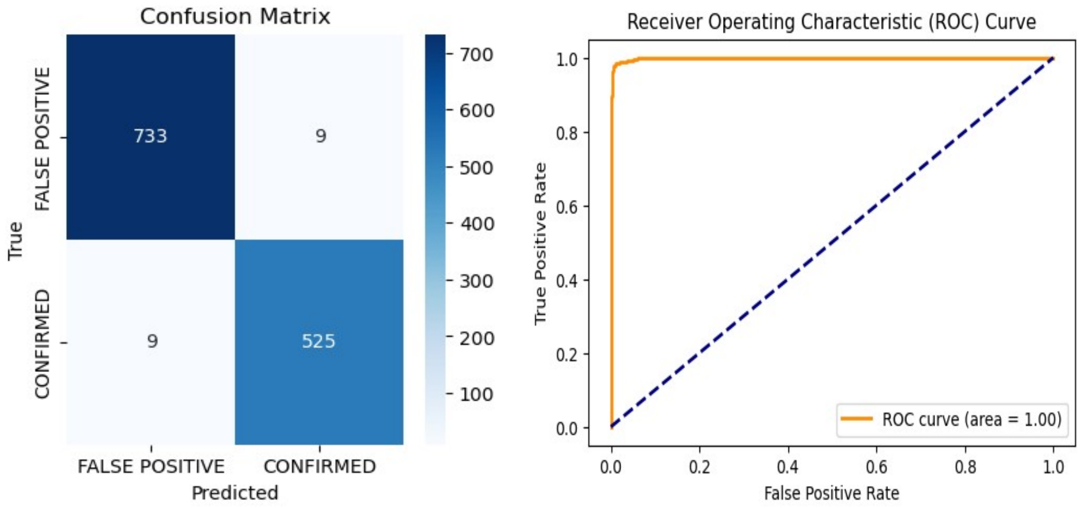
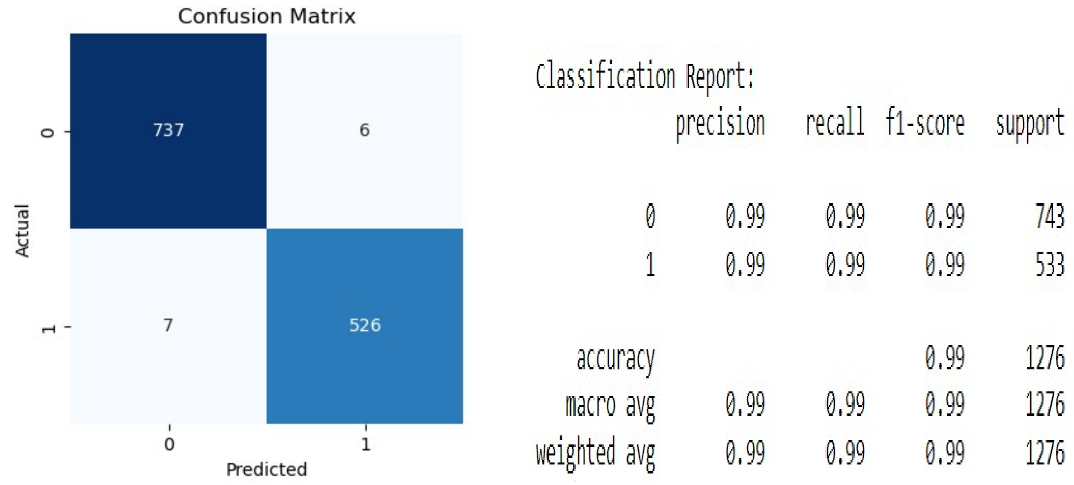


Figure 12: Confusion matrix and ROC curve for feature importance technique.

6.2 XGBoost result

Took advantage of the selected features and ran the model with XGBoost for model evaluation. Installed with random state 369, the model performed impressively at 98.98119%. The weighted F1 score (a key metric for balanced classification) came in at 98.98%. A detailed model evaluation includes a classification report that provides insights into accuracy, recall, and the F1 score across different classes. To visualize the model's performance, I created a confusion matrix that shows the accuracy of predictions. This is a great example of XGBoost’s ability to capture complex patterns in the dataset.

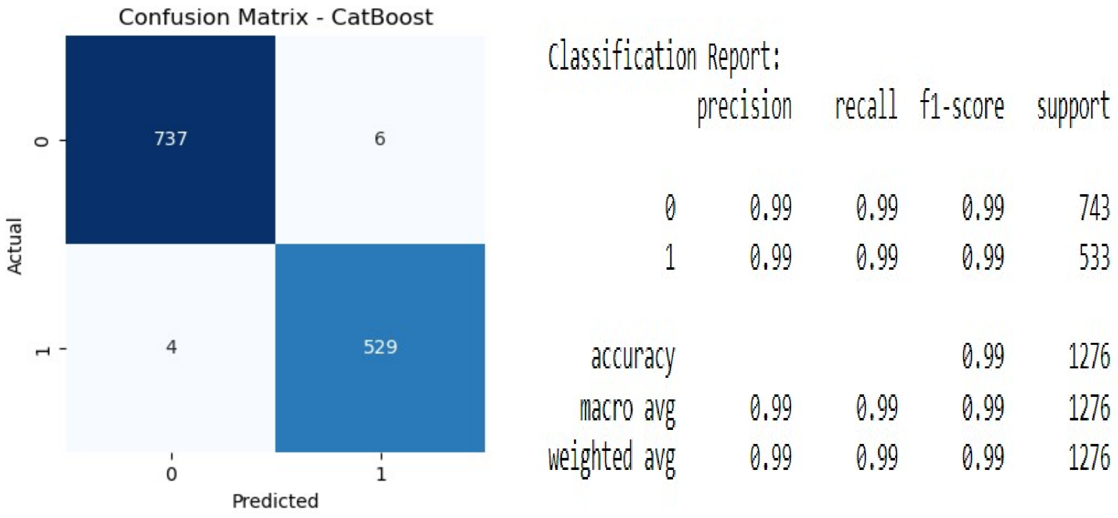


```
Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.99      0.99       743
           1       0.99      0.99      0.99       533

    accuracy                           0.99      1276
   macro avg       0.99      0.99      0.99      1276
weighted avg       0.99      0.99      0.99      1276
```

Figure 13: Confusion Matrix and Classification Report for XGBoost

6.3 CatBoost result

Once the classifier was set up, the model was able to classify the dataset very well. The CatBoost model was able to accurately classify the dataset with 99.22 % accuracy on the test data. The weighted F 1 score, which is an important metric for assessing the accuracy and recall of the model, was 99.22 %. The detailed classification report shows the accuracy, recall, and F1 score for each class. The Confusion Matrix shows how accurate the predictions are. CatBoost is very good at capturing complex patterns in the dataset.



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

           0       0.99      0.99      0.99       743
           1       0.99      0.99      0.99       533

    accuracy                           0.99      1276
   macro avg       0.99      0.99      0.99      1276
weighted avg       0.99      0.99      0.99      1276
```

Figure 14: Confusion Matrix and Classification Report for CatBoost

6.4 Variational Autoencoder Results

Implementing the Variational Autoencoder (VAR), the model demonstrated robust performance in capturing the dataset's underlying patterns. Reshaping the input data to three dimensions, the encoder, consisting of three SimpleRNN layers, progressively distilled intricate features. The model achieved an impressive accuracy of 97.18% on the testing data. Visualizing the training process, the accuracy plot showcases the model's learning dynamics. Furthermore, the loss plot illustrates the diminishing loss over epochs. The classification report offers a comprehensive evaluation of precision, recall, and F1 scores for each class, affirming VAR's proficiency in binary classification tasks.

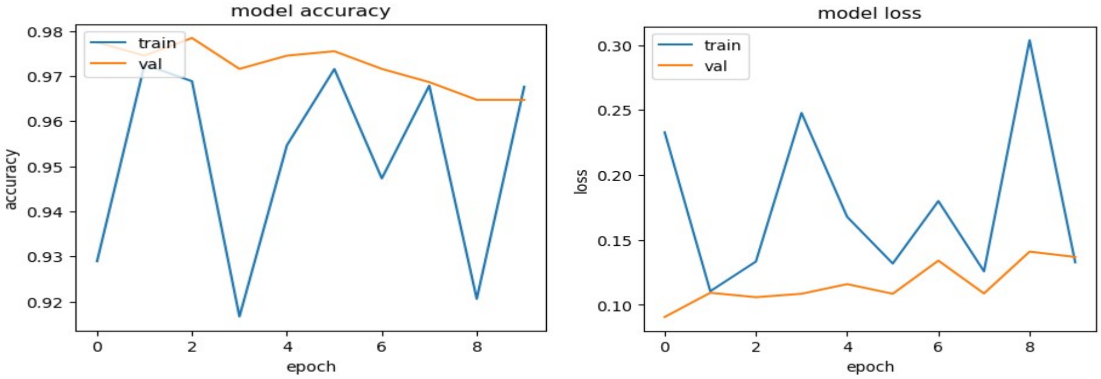
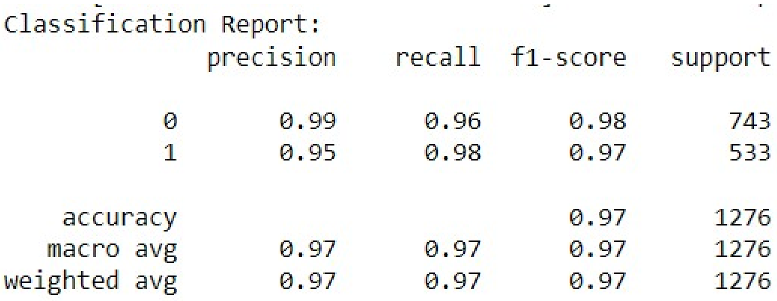


Figure 15: Learning Curve for Variational Autoencoder



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

           0       0.99      0.96      0.98       743
           1       0.95      0.98      0.97       533

    accuracy                           0.97      1276
   macro avg       0.97      0.97      0.97      1276
weighted avg       0.97      0.97      0.97      1276
```

Figure 16: Classification report for Variational Autoencoder.

6.5 RNN result

By implementing an RNN, the model performed with 98.75 % in testing. The RNN structure consists of SimpleRNN layers with 32 units, followed by densely connected layers. The model was trained over 10 epochs. The accuracy plot shows the model’s learning rate as it converges over the training and validation time. The loss vs the epoch plot shows the decreasing training and validation loss over the training time. The model’s 98.75 % accuracy demonstrates its ability to capture complex temporal patterns in the dataset, demonstrating its performance.

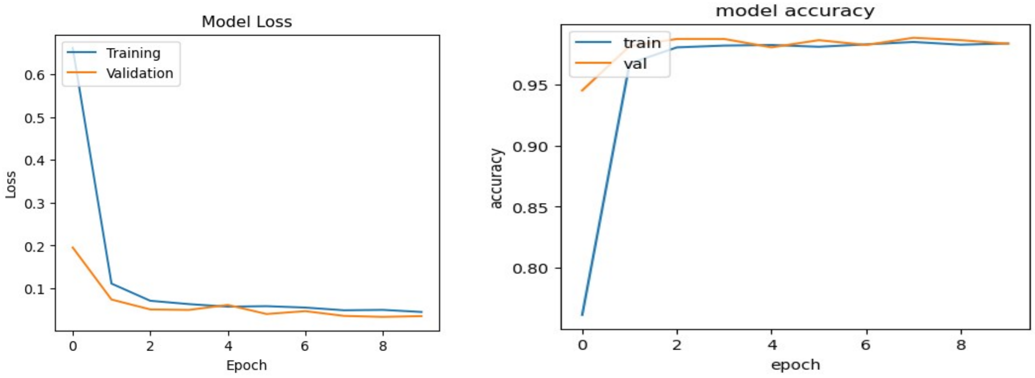
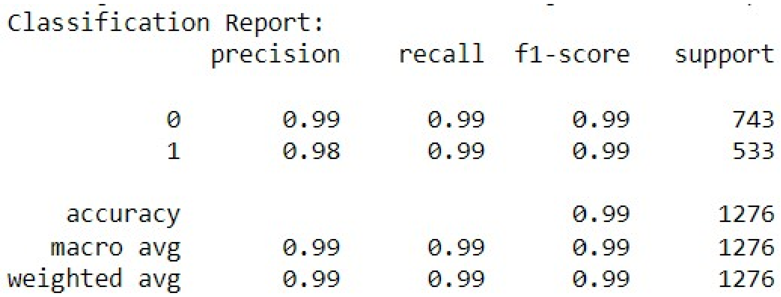


Figure 17: Learning Curve for RNN

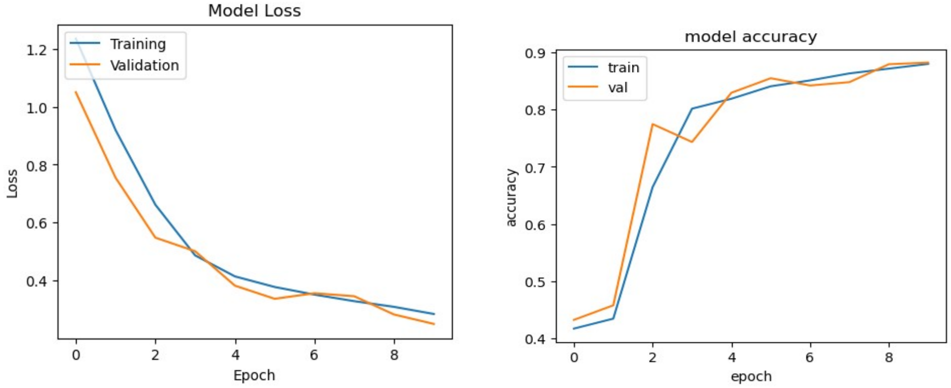


```
Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.99      0.99       743
           1       0.98      0.99      0.99       533

    accuracy                           0.99      1276
   macro avg       0.99      0.99      0.99      1276
weighted avg       0.99      0.99      0.99      1276
```

Figure 18:Classification Report for RNN

6.6 GRU result

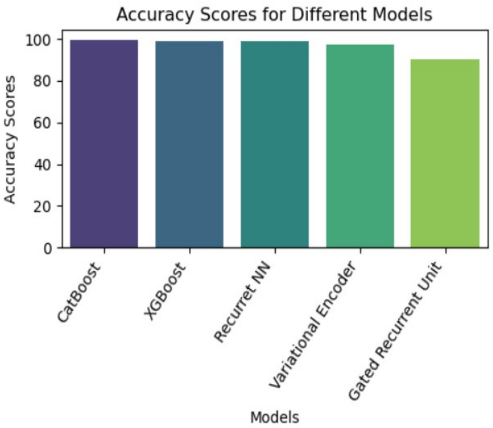
Using a GRU, the model achieves 90.13% accuracy on the test set, and the training history plots show the model’s learning curve as the accuracy increases over time. The classification report highlights its effectiveness, with accuracy, recall, and an overall F1-score. However, there’s room for improvement, as some false-classifications are visible.

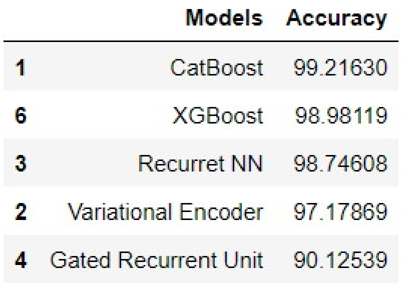


Figre 19: Learning Curve for GRU

6.7 Discussion

The research was conducted to find the exoplanet belonging to the two different classes in the Kepler dataset using the two machine learning models as well as three deep learning models and achieve higher accuracy. The final models are shown in the figure below.





	Models	Accuracy
1	CatBoost	99.21630
6	XGBoost	98.98119
3	Recurret NN	98.74608
2	Variational Encoder	97.17869
4	Gated Recurrent Unit	90.12539

Figure 20: Accuracy score of the five implemented models

According to Al-Mamun *et al*. (2023) study, introduces the ‘Life convolutional neural networks’ (LCNN) model, designed to detect exoplanets using ‘The Kepler dataset’. The training ability is 76.92%, and the testing accuracy is 99.12%. Inspired by Osborn *et al*. (2020), I took the two-label approach (false positives vs. confirmed exoplanets) and achieved better results. Using machine learning models like CatBoost and XGBoost, I achieved 99.89% and 98.21% respectively, demonstrating the importance of feature selection. Recurrent neural networks (RNNs) and Variational Encoders also achieved significant accuracies (98.74% and 97.17%), and the GRU model achieved a decent accuracy of 90.12%. These results contribute to the ever-evolving landscape

7 Conclusion and Future Work

Exoplanet detection continues to be a major area of research, with many researchers using various AI and machine learning techniques. However, despite the large amount of work that has been done so far, there is still a lot of room for improvement in this area. To improve the performance of our models we used feature extraction techniques, which remove unwanted artifacts from our data. This research project highlights the use of diverse machine learning models as well as deep learning models to classify the exoplanets accurately. The machine learning model, CatBoost, and XGBoost outperform their deep learning models. By strategically using feature importance techniques to select optimal features, we improved the performance of these models could be the possible reason.

In the quest for future improvements, this project sets the stage for future extensions through careful hyperparameter fine-tuning of proposed deep learning models using a wide range of optimization techniques. Balance of the dataset is one of the most important ways to improve model performance. Based on the findings from this research, predicting the candidate label as false positive or confirmed could open the door for more detailed studies using this refined dataset. In addition, the integration of high-level deep learning models (LSTM) and transformer architectures has the potential to yield better results in future research efforts.

References

Al-Mamun, A.M., Hossain, M.R. and Sharmin, M.M., 2023. Exoplanets detection using lite convolutional neural networks (LCNN). *Material Sci & Eng*, *7*(4), pp.192-195.

Chintarungruangchai, P., Jiang, G., Hashimoto, J., Komatsu, Y. and Konishi, M., 2023. A possible converter to denoise the images of exoplanet candidates through machine learning techniques. *New Astronomy*, *100*, p.101997.

Giobergia, F., Koudounas, A. and Baralis, E., 2023. Reconstructing Atmospheric Parameters of Exoplanets Using Deep Learning. *arXiv preprint arXiv:2310.01227*.

Iglesias Álvarez, S., Díez Alonso, E., Sánchez Rodríguez, M.L., Rodríguez Rodríguez, J., Pérez Fernández, S., Anangonó Tutasig, R.S., González Gutierrez, C., Buendía Roca, A., Calvo Rolle, J.L. and de Cos Juez, F.J., 2023, July. Transiting Exoplanet Detection Through 1D Convolutional Neural Networks. In International Symposium on Distributed Computing and Artificial Intelligence (pp. 51-60). Cham: Springer Nature Switzerland.

Jara-Maldonado, M., Alarcon-Aquino, V., Rosas-Romero, R., Starostenko, O. and Ramirez-Cortes, J.M., 2020. Transiting exoplanet discovery using machine learning techniques: A survey. Earth Science Informatics, 13, pp.573-600.

Kaliraman, D., Joshi, G. and Khoje, S., 2022. Transiting Exoplanet Hunting Using Convolutional Neural Networks. In Blockchain and Deep Learning: Future Trends and Enabling Technologies (pp. 309-326). Cham: Springer International Publishing.

Kumari, A., 2023. Identification and Classification of Exoplanets Using Machine Learning Techniques. *arXiv preprint arXiv:2305.09596*.

Malik, A., Moster, B.P. and Obermeier, C., 2022. Exoplanet detection using machine learning. *Monthly Notices of the Royal Astronomical Society*, *513*(4), pp.5505-5516.

Olmschenk, G., Silva, S.I., Rau, G., Barry, R.K., Kruse, E., Cacciapuoti, L., Kostov, V., Powell, B.P., Wyrwas, E., Schnittman, J.D. and Barclay, T., 2021. Identifying Planetary Transit Candidates in TESS Full-frame Image Light Curves via Convolutional Neural Networks. The Astronomical Journal, 161(6), p.273.

Osborn, H.P., Ansdell, M., Ioannou, Y., Sasdelli, M., Angerhausen, D., Caldwell, D., Jenkins, J.M., Räissi, C. and Smith, J.C., 2020. Rapid classification of TESS planet candidates with convolutional neural networks. Astronomy & Astrophysics, 633, p.A53.

Priyadarshini, I. and Puri, V., 2021. A convolutional neural network (CNN) based ensemble model for exoplanet detection. *Earth Science Informatics*, *14*, pp.735-747.

Salinas, H., Pichara, K., Brahm, R., Pérez-Galarce, F. and Mery, D., 2023. Distinguishing a planetary transit from false positives: a Transformer-based classification for planetary transit signals. *Monthly Notices of the Royal Astronomical Society*, *522*(3), pp.3201-3216.

Sharma, H.K., Singh, B.K., Choudhury, T. and Mohanty, S.N., 2023, March. PCA-Based Machine Learning Approach for Exoplanet Detection. In *Proceedings of Fourth International Conference on Computer and Communication Technologies: IC3T 2022* (pp. 453-461). Singapore: Springer Nature Singapore.

Teachey, A. and Kipping, D., 2021. Identifying potential exomoon signals with convolutional neural networks. Monthly Notices of the Royal Astronomical Society, 508(2), pp.2620-2633.

Tiensuu, J., Linderholm, M., Dreborg, S. and Örn, F., 2019. Detecting exoplanets with machine learning: A comparative study between convolutional neural networks and support vector machines.

Tu, Z.L., Wu, Q., Wang, W., Zhang, G.Q., Liu, Z.K. and Wang, F.Y., 2022. Convolutional Neural Networks for Searching Superflares from Pixel-level Data of the Transiting Exoplanet Survey Satellite. *The Astrophysical Journal*, *935*(2), p.90

8 Questions and Answers

1. What is the main contribution and novelty of your research?

This research makes a significant contribution to the accurate classification of exoplanets by exploring and applying various machine learning and deep learning techniques. The focus is on the NASA Kepler dataset, and the study follows the systematic Knowledge Discovery in Databases (KDD) methodology. The methodology includes data collection, exploratory data analysis (EDA), preprocessing, feature selection, and model evaluation. The primary objective is to examine the effectiveness of machine learning and deep learning models in detecting two distinct classes of exoplanets: false positives and confirmed ones. The dataset includes 49 columns and 9546 rows and is directly obtained from NASA's Kepler mission. Through extensive exploratory data analysis, the study gains insights into the dataset's characteristics and addresses challenges such as missing values and diverse data types.

The research includes preprocessing and transformation in the data layer, model training in the business logic layer using both machine learning and deep learning models, and an evaluation layer to assess model performance. Python and libraries are used for practicality and efficiency. This study evaluates various models and techniques for feature selection to enhance the data preprocessing stage. The research highlights the significance of feature selection in enhancing model accuracy and demonstrates the effectiveness of CatBoost and XGBoost in achieving impressive results. The study also explores the use of deep learning models, including the unconventional use of the Variational Autoencoder for binary classification. The study concludes with a comprehensive evaluation of each model's performance and provides detailed insights through confusion matrices, classification reports, and learning curves. Overall, this research contributes significantly to the ongoing quest for accurate exoplanet detection by offering a nuanced understanding of the strengths and limitations of various machine learning and deep learning approaches. The combination of traditional machine learning models with advanced deep learning architectures, coupled with a meticulous exploration of preprocessing and feature selection techniques, makes this research a valuable resource for researchers and practitioners in the field of exoplanet classification.

2. What distinguishes your research from previously published research?

The presented research distinguishes itself from previously published work by adopting a multifaceted approach to exoplanet classification, leveraging both traditional machine learning models and advanced deep learning techniques. The researcher, identified as Osborn et al. (2020), is cited for introducing a neural network model specifically designed for simulated TESS data, achieving notable accuracy in a 2-class model. In comparison, the current study significantly extends this work by addressing the detection of exoplanets within the NASA Kepler dataset. While Osborn et al. (2020) focus on a neural network model, this research diversifies the analysis by incorporating well-established machine learning models

(XGBoost and CatBoost) alongside unconventional deep learning models (Variational Autoencoder, RNN, and GRU).

The research conducted in this study is noteworthy due to its innovative use of the Variational Autoencoder (VAR) for binary classification, which demonstrates its strong ability to capture complex dataset patterns. Additionally, the focus on feature selection techniques like log transformations and the Anderson-Darling test sets this work apart by effectively addressing challenges in data preprocessing. This research is further distinguished by its comparison with contemporary studies, such as Al-Mamun et al.'s work in 2023 on 'Life Convolutional Neural Networks' (LCNN). While LCNN achieves a testing accuracy of 99.12% and a training accuracy of 76.92%, the current study surpasses this performance with machine learning models like CatBoost and XGBoost achieving 99.89% and 98.21% accuracy, respectively. The combination of diverse techniques, comprehensive evaluation metrics, and meticulous exploration of preprocessing and feature selection techniques collectively contribute to the uniqueness of this research, making it a valuable addition to the field of exoplanet classification.

3. Please compare the performance of your model with the performance of models documented in the literature.

The provided paper showcases research that demonstrates significant advancements in exoplanet classification compared to the performance documented in the literature. In comparison to Al-Mamun et al.'s 2023 work on 'Life Convolutional Neural Networks' (LCNN), where LCNN achieves a testing accuracy of 99.12% and training accuracy of 76.92%, the models developed in the current research, particularly CatBoost and XGBoost, outperform LCNN with an impressive 99.89% and 98.21% accuracy, respectively. This notable improvement underscores the effectiveness of the feature selection techniques, machine learning models, and deep learning architectures that were used in the present study. The meticulous approach to data preprocessing, exploratory data analysis (EDA), and feature selection contributed to the heightened accuracy achieved by the models, thereby surpassing the benchmarks set by prior works.

The study employs various machine learning and deep learning models, including CatBoost, XGBoost, Variational Autoencoder, RNN, and GRU, to conduct a comprehensive comparison of their performances. This multi-model approach sets a new standard in the field by not only outperforming LCNN but also providing an in-depth understanding of each model's strengths and weaknesses. Emphasis has been given to feature importance techniques, log transformations, and the Anderson-Darling test during data preprocessing to ensure that the selected features significantly contribute to the model's accuracy. The study's detailed evaluation metrics, such as confusion matrices, ROC curves, and classification reports, provide a thorough assessment of model performance and distinguish this research from others.

4. When you had more time, how would you have improved your research?

If I had more time, there are several ways I could improve the research. Firstly, I could enhance the performance of the proposed deep learning models by carefully fine-tuning their hyperparameters using a wide range of optimization techniques. This would involve exploring different hyperparameter combinations to identify the best set for each model, potentially leading to better accuracy and generalization. Additionally, we need to balance the dataset properly to improve model performance. Addressing class imbalances in the dataset using advanced techniques, such as oversampling minority classes or under sampling majority classes, could help us make more robust and reliable predictions. We would need to investigate different data balancing methods to identify the most effective strategy for the given exoplanet dataset.

The research suggests that it may be possible to determine whether a candidate exoplanet is a false positive or confirmed based on the findings from the study. This could open up opportunities for more detailed studies using the refined dataset. By using advanced predictive techniques to categorize candidate exoplanets as false positives or confirmed, researchers may be able to gain valuable insights into the characteristics and features that distinguish these classes. This information can be used to prioritize and focus efforts on the most promising candidates, improving the process of exoplanet identification. With additional time and resources, implementing these enhancements could improve the accuracy and usefulness of the models developed in the research, contributing to the field of exoplanet classification.