Contents

1 INTRODUCTION 2
1.1 Solar Radiation Prediction 2
1.2 Asteroid Impact Prediction 2
1.3 Exoplanet Classification 3

2 Related Works 3
2.1 Dataset 1: Solar Radiation Prediction 3
2.2 Dataset 2: Asteroid Impact Prediction 3
2.3 Dataset 3: Exoplanet Classification 3

3 Methodology 3

4 Data Cleaning, Exploration and Transformation 3
4.1 Dataset 1: Solar Radiation Prediction 3
4.1.1 Data Description and overview 3
4.1.2 Data Exploration and Cleaning 4
4.1.3 Feature Selection and Engineering . . . 4
4.1.4 Model Selection 4
4.2 Dataset 2: Asteroid Impact Prediction 5
4.2.1 Data Description and Overview 5
4.2.2 Data Cleaning and Exploration 5
4.2.3 Feature Selection and Engineering . . . 5
4.3 Dataset 3: Exoplanet Classification 6
4.3.1 Data Description and Overview 6
4.3.2 Data Exploration and Cleaning 6
4.3.3 Feature Selection 7
4.3.4 Model Selection 7

5 Modelling and Predictions 7
5.1 Dataset 1: Solar radiation Prediction 8
5.1.1 Model 1: Random Forest Regressor . . 8
5.1.2 Model 2: Gradient Boosting 8
5.1.3 Model Analysis 8
5.2 Dataset 2: Asteroid Impact Prediction 8
5.2.1 Model 1: Stochastic Gradient Descent . 8
5.2.2 Model 2: Decision Tree 8
5.2.3 Model Analysis 8
5.3 Dataset 3: Exoplanet Classification 8
5.3.1 Model 1: kNN Classifier 8
5.3.2 Model 2: Naive Bayes classifier 9
5.3.3 Model Analysis 9

6 Conclusions 9

7 Further Work 9

List of Figures

1 KDD Methodology 3
2 Data description for solar radiation dataset . . 4
3 Data head for solar radiation dataset 4
4 Radiation as a function of time 4
5 Boxplot of variables 4
6 Boxplot of variables(cont.) 4
7 Correlation heat map for dataset 1 4
8 Dataset 2- Orbits file description 5
9 Dataset 2 -Impacts file description 5
10 Possible Impacts vs Asteroid Velocity 5
11 Possible Impacts vs Asteroid Magnitude 5
12 Number of asteroids by category 5
13 Correlation heat map 5
14 Possible impacts vs Period 6
15 Object with largest diameter 6
16 Object with highest cumulative impact probability 6
17 Data description for dataset 3 6
18 Percentage of Null values in the columns . . . 6
19 Outliers for dataset 3-(1) 7
20 Outliers for dataset 3-(2) 7
21 Outliers for dataset 3-(3) 7
22 Outliers for dataset 3-(4) 7
23 Outliers for dataset 3-(5) 7
24 Distribution of target varibale categories . . . 7
25 Correlation heat map for dataset 3 7
26 Hyperparamter tuning for random forest model on dataset 1 8
27 Mean squared error for random forest regressor 8
28 R squared for random forest 8
29 Mean squared error for gradient boost 8
30 R squared for gradient boost 8
31 Feature importance analysis for random forest and gradient boosting models 8
32 Overall score for Stochastic gradient descent algorithm 8
33 Accuracy for decision tree 8
34 KNN classification summary 9
35 Classification Summary for Naive Bayes model 9
36 ROC for kNN classifier 9
37 ROC for Naive Bayes classifier 9

Solar Radiation Prediction, Asteroid Impact Prediction and Exoplanet Classification using Machine Learning Models

Jaswinder Singh[1]

[1]Student ID: x19219997, MSc in Data Analytics, National College of Ireland

ABSTRACT This report aims to apply various machine learning algorithms to accomplish prediction and classification tasks using the KDD methodology. The tasks are *Solar Radiation Prediction*, *Asteroid Impact Prediction* and *Exoplanet Classification*. While the *solar radiation prediction*, *asteroid impact prediction* are the regression tasks, the *exoplanet classification*, as the name suggests, is a classification problem. Separate data exploration, cleaning and transformation was performed for each dataset before applying various machine learning models to them. The models were then evaluated on the basis of various parameters. For the classification task, *k-nearest neighbour(kNN) classifier*, *support vector machine(SVM) and Naive Bayes* achieved an overall accuracy of 80.6%., 81.96% and 82.22% respectively. The predictions for regression tasks for different were compared on the basis of RMSE(root mean squared error), MAPE(mean absolute percentage error) and adjusted R^2 values. The conclusions were then drawn and reported comprehensively.

KEYWORDS: Exoplanet, Classification, Regression, kNN Classifier, Support vector Machine(SVM), Naive bayes, RMSE, MAPE, Adjusted R^2

1. INTRODUCTION

1.1 Solar Radiation Prediction

Almost every life form on Earth depends on the Sun in some way to survive and nurture. The Sun acts an ultimate source of energy for almost all of the natural processes occurring on our planet. The humans have been using and exhausting Earth's natural resources since the pre-historic times. Our planet has now reached such a point that its natural order and ecosystem have been completely disrupted by these continuous exhaustion. The scientists and researchers around the world are now working hard towards finding alternative renewable resources that are both cheap and easily accessible to people to meet the ever increasing energy requirements of the human population. One of the most sought after source is our Sun. According to a study [], the Sun radiates enough energy on Earth in a second to satisfy the entire energy demand of the planet for 2 whole hours. Therefore, if all this incredible energy can somehow be harnessed in a way to cater the human energy requirements, it would solve part of world's energy crisis. The idea of using silicon embedded solar cells came long before in the late 19th century when Edmond Becquerel discovered the phenomenon of photovoltaic effect. But we have come a long way since then. Now we have miniaturized these huge cells to fit into any device we want. Today we are just improvising on the already discovered techniques. One such improvisation is the forecast of solar radiation at a place using machine learning algorithms. It would help a great deal when it comes to installing solar cells at a place that would cost a lot. if we could predict within a desired uncertainty range, the amount of solar radiation hitting a place, we would save a lot of money and human efforts which could be used elsewhere. We will use the data from HI SEAS research program funded by NASA, to predict the amount of solar radiation in Watts per sq.m that will fall at a given place using various data mining and machine learning models.

1.2 Asteroid Impact Prediction

Our planet is bombarded by millions of pieces of space debris every day. These debris pieces have different shapes, sizes and masses varying from few centimeters to metres and kilometers. Although most of this debris burns up in the Earth's atmosphere due to friction, some of them make their way to the ground. The asteroids are a special kind of space debris that's not very common in terms of contact with our planet. Asteroids are irregularly shaped bodies that orbit the Sun and some of them fly past our planet every few months or years. There is paleontological evidence that the dinosaurs were wiped out from the planet 65 million years ago after a city sized asteroid hit Earth. Since, these planet killer asteroids hit Earth roughly every 40 to 50 million years, our planet is due to be hit. There is a dire need today, to understand these mysterious and dangerous space objects, if we want the human race to have a chance of survival. Today, scientists are devising various techniques to study and possibly predict the impacts from these hazardous asteroids. one of these techniques is to use machine learning algorithms to predict the probability of impact of these asteroids. We feed the machine learning algorithm with the data containing information for various asteroids, their periods, impact probabilities, sizes, etc. to enable it to learn and make predictions about the potential future impacts. In this project, we will use the data of asteroids provided by NASA and use various machine learning models to make predict the probability of impact. We will also be using various methods to evaluate our models.

1.3 Exoplanet Classification

Since the dawn of humanity, we have been asking the question: "Are we alone in the universe?" Scientists haven't found any conclusive evidence yet to prove the existence of extra terrestrial intelligence but we might just be on the verge of a major breakthrough. This is because NASA has been working on finding the signs of alien intelligence for almost half a century now and researchers around the world think that we might be extremely close to major breakthrough. There are several ways to look for the extra terrestrial life. One is to look out in the space for planets that could potentially harbour a complex or even a primitive life form. NASA has launched several telescopes like Kepler Space Observatory(KSO) in 2009, Transiting Exoplanet Survey Satellite(TESS) in 2018 and others which have been continously scanning the skies in search for exoplanets(planets outside our solar system). These observatories have collected tons of data that is still being analysed. In this report, we will use a small subset of the Kepler Observatory data to classify an extra solar object as an exoplanet or not using various machine learning algorithms.

2. RELATED WORKS

2.1 Dataset 1: Solar Radiation Prediction

- Cyril Voyant and various others [1] have worked and reviewed various models like ARIMA, SVM and random forests to forecast the solar radiation. They also discuss about the employment of neural networks like ANN(Artificial Neural Networks) to overcome the shortcomings of traditional machine learning methods.
- Veysel Coban and Sezi Çevik Onar have done excellent research in solar radiation prediction by using the data from Istanbul region located in Turkey. They focus more on the role of variability in the data for deciding the best model for forecasting [2].
- Xiaoyan Shao and others [3] discuss the very interesting approach of statistically combination of several machine learning algorithms to improve the forecasting accuracy for solar radiation.

2.2 Dataset 2: Asteroid Impact Prediction

- There have been numerous developments in the domain of the prediction of an asteroid's diameter based on its different orbital parameters. One such attempt was made in a Kaggle competition. The candidates were provided with NASAs JPL database for the problem. Blakelobato discusses the whole process of the model building in his blog [4].
- Researchers have also used Artificial Neural Networks(ANNs) for identifying the hazardous asteroids [5]. They discuss the calculation of impact probability using Monte Carlo simulations as adopted by NASA's Sentry system and also build on to demonstrate how artificial neural networks works better than most of the existing models using subtle arguments. They have shown that their instrument HOI(Hazardous Object Identifier) which uses ANN, they are able to identify 95.25 % of the potential hazardous simulated impactors.
- E. R. Nesvold and other [6] discuss the intriguing possibility of developing the technology that would use machine learning framework for deflecting the potentially hazardous objects like asteroids.

2.3 Dataset 3: Exoplanet Classification

- Brychan Manry [7] discusses the pipeline for machine learning that can be used for exoplanet classification tasks. They discuss the random forest classifier in detail for the classification task.
- Abhishek Malik et.al [8] discuss the technique of transiting for classifying an exoplanet. They have used a tree based classifier and a tool lightgbm to train their model. They were able to achieve an impressive accuracy of 98% and a recall value of 0.82.
- Researchers form Harvard have used deep learning framework for identifying the potential exoplanets [9]. They trained a deep convolutional neural network to identify whether the provided signal is an exoplanet or not.

3. METHODOLOGY

In this project we will use the KDD(Knowledge Discovery from Data) methodology for building the machine learning models. It involves the following steps:

- Developing and understanding of the application: The tasks were analysed and objectives were stated clearly.
- Creation of the target variable: The target variable was decided for each of the three datasets.
- Data cleaning and exploration: The data was first imported and cleaned. The null values were handled by proper methods(mean, median, etc.). After that it was then explored using various programming techniques taught in the module. In this step, various variables and their effect on the target variable were also studied and visualised.
- Data Transformation: In this step, the variables were transformed into the prediction useful variables using various techniques.
- A suitable data mining technique(like normalisation, encoding, etc.) was chosen for each of the dataset.
- After that, a suitable mining algorithm was chosen for each of the tasks.
- The models were deployed and the predictions were obtained.
- Conclusions were then drawn from those predictions.

A summary of all the above steps is depicted in the figure 1.

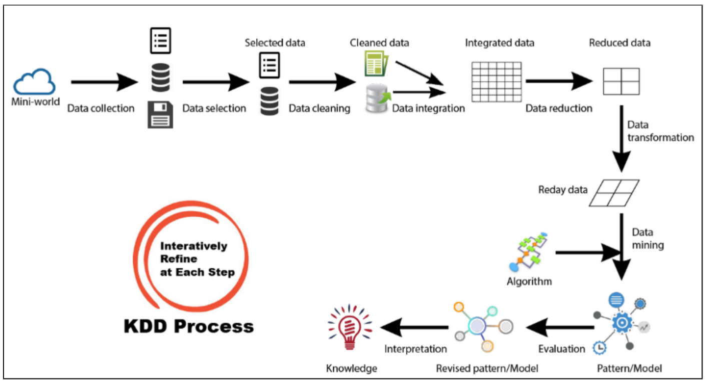


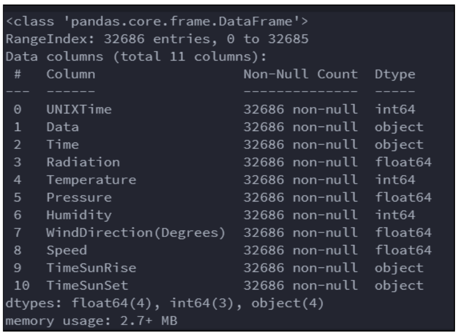
FIGURE 1. KDD Methodology

4. DATA CLEANING, EXPLORATION AND TRANSFORMATION

4.1 Dataset 1: Solar Radiation Prediction

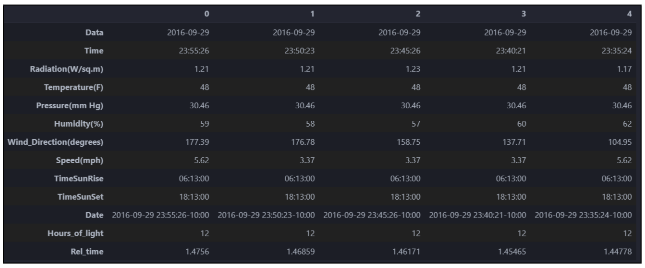
4.1.1 Data Description and overview

The data overview is shown in the figures 2 and 3. As we can see from figure 2, there are no NULL values in our data. it contains 32,686 rows and 11 columns.
Target Variable: 'Radiation(W/sq.m)'



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32686 entries, 0 to 32685
Data columns (total 11 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   UNIXTime                32686 non-null  int64
 1   Data                    32686 non-null  object
 2   Time                    32686 non-null  object
 3   Radiation               32686 non-null  float64
 4   Temperature             32686 non-null  int64
 5   Pressure                32686 non-null  float64
 6   Humidity                32686 non-null  int64
 7   WindDirection(Degrees)  32686 non-null  float64
 8   Speed                   32686 non-null  float64
 9   TimeSunRise             32686 non-null  object
 10  TimeSunSet              32686 non-null  object
dtypes: float64(4), int64(3), object(4)
memory usage: 2.7+ MB
```

FIGURE 2. Data description for solar radiation dataset



	0	1	2	3	4
Data	2016-09-29	2016-09-29	2016-09-29	2016-09-29	2016-09-29
Time	23:55:26	23:50:23	23:45:26	23:40:21	23:35:24
Radiation(W/sq.m)	1.21	1.21	1.23	1.21	1.17
Temperature(F)	48	48	48	48	48
Pressure(mm Hg)	30.46	30.46	30.46	30.46	30.46
Humidity(%)	59	58	57	60	62
Wind_Direction(degrees)	177.39	176.78	158.75	137.71	104.95
Speed(mph)	5.62	3.37	3.37	3.37	5.62
TimeSunRise	06:13:00	06:13:00	06:13:00	06:13:00	06:13:00
TimeSunSet	18:13:00	18:13:00	18:13:00	18:13:00	18:13:00
Date	2016-09-29 23:55:26-10:00	2016-09-29 23:50:23-10:00	2016-09-29 23:45:26-10:00	2016-09-29 23:40:21-10:00	2016-09-29 23:35:24-10:00
Hours_of_light	12	12	12	12	12
Rel_time	1.4756	1.46859	1.46171	1.45465	1.44778

FIGURE 3. Data head for solar radiation dataset

4.1.2 Data Exploration and Cleaning

1. The time series plot of radiation as a function of time is shown in the figure 4. It can be seen that the radiation variable shows a strong seasonality which is expected because there are cycles in the sunlight at days and nights(0 or minimum at night)
2. The boxplot of various variables is shown in the figures 5 and 6 along with their distribution. It can be seen that around 50% of the values of radiation variable are between 0 W/sq.m and 250 W/sq.m. The wind speeds column contains some outliers in the range of 0 to 20 miles/hr.

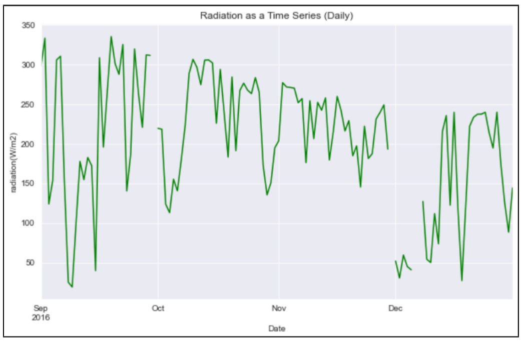


FIGURE 4. Radiation as a function of time

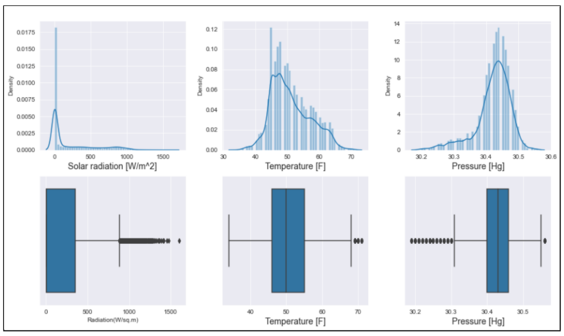


FIGURE 5. Boxplot of variables

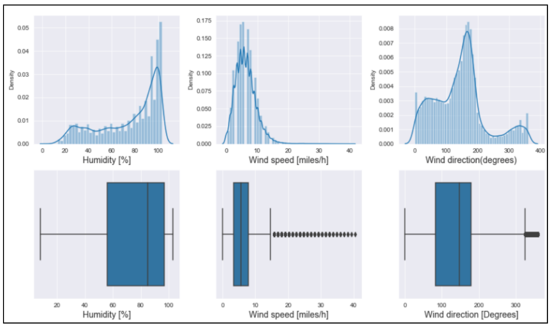


FIGURE 6. Boxplot of variables(cont.)

4.1.3 Feature Selection and Engineering

1. Firstly all the variables were considered for contributing in the model for the radiation prediction.
2. A new variable called 'Reltime' was made which was defined as follows:

$$\text{Relative time} = \frac{(currenttime - sunrisetime)}{(sunrisetime - sunsettime)}$$

3. The variables 'TimeSunRise' and 'TimeSunSet' were converted into timestamp format.
4. After that, the correlation of the variables was analysed using correlation heat map(refer to figure 7)

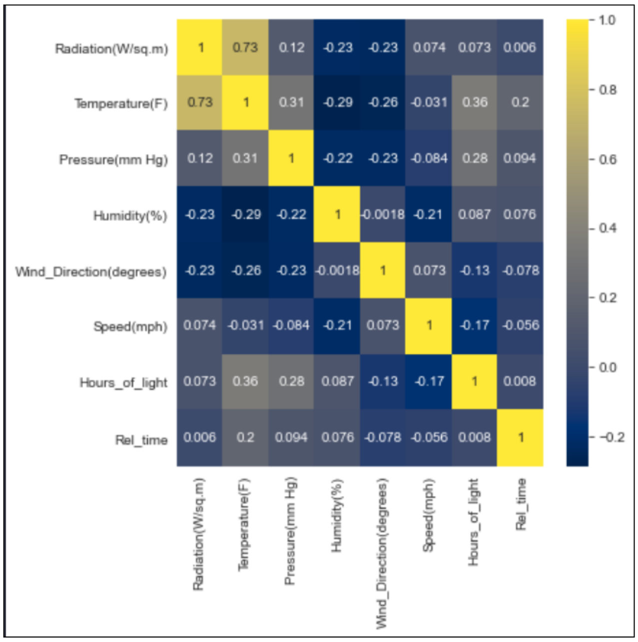


FIGURE 7. Correlation heat map for dataset 1

5. It can be seen that no two variables have a correlation value of more than 0.4. Hence, we can say that variables are not correlated with one another

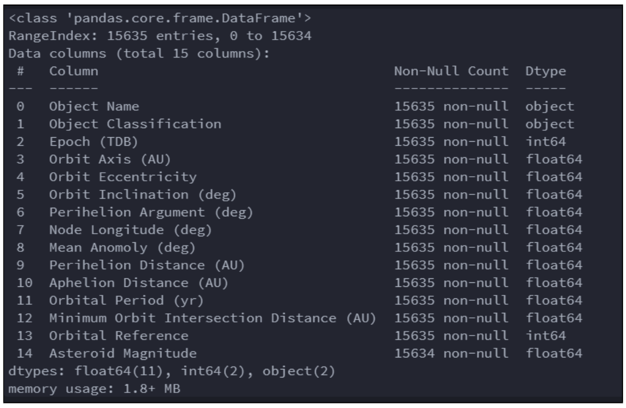
4.1.4 Model Selection

Since the task is a regression problem, two algorithms were used to predict the target variable : Random Forest regressor and the Gradient Boost. The feature importance analysis was also performed for both the models after the model deployment.

4.2 Dataset 2: Asteroid Impact Prediction

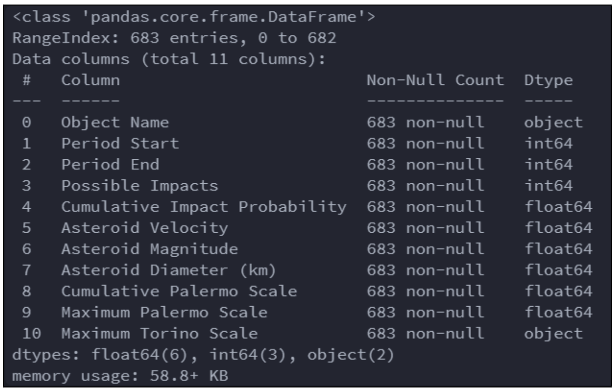
4.2.1 Data Description and Overview

The data overview and description are shown in figures 8 and 9. The data is contained in two different csv files named 'impacts' and 'orbits'. The impacts file contains information about the asteroids and their impact probabilities, diameters, etc. The orbit file contains information about the asteroids and their orbital parameters like eccentricity, inclination, perihelion distance, etc. All the variables except 'Object Name' and 'Object Classification' are numerical. It can be seen from the description that there are no NULL values in our data. Target variable: 'Cumulative Impact Probability'



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15635 entries, 0 to 15634
Data columns (total 15 columns):
 #   Column                                 Non-Null Count  Dtype
---  ------                                 --------------  -----
 0   Object Name                            15635 non-null  object
 1   Object Classification                  15635 non-null  object
 2   Epoch (TDB)                            15635 non-null  int64
 3   Orbit Axis (AU)                        15635 non-null  float64
 4   Orbit Eccentricity                     15635 non-null  float64
 5   Orbit Inclination (deg)                15635 non-null  float64
 6   Perihelion Argument (deg)              15635 non-null  float64
 7   Node Longitude (deg)                   15635 non-null  float64
 8   Mean Anomoly (deg)                     15635 non-null  float64
 9   Perihelion Distance (AU)               15635 non-null  float64
 10  Aphelion Distance (AU)                 15635 non-null  float64
 11  Orbital Period (yr)                    15635 non-null  float64
 12  Minimum Orbit Intersection Distance (AU)  15635 non-null  float64
 13  Orbital Reference                      15635 non-null  int64
 14  Asteroid Magnitude                     15634 non-null  float64
dtypes: float64(11), int64(2), object(2)
memory usage: 1.8+ MB
```

FIGURE 8. Dataset 2- Orbits file description



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 683 entries, 0 to 682
Data columns (total 11 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Object Name                   683 non-null    object
 1   Period Start                  683 non-null    int64
 2   Period End                    683 non-null    int64
 3   Possible Impacts              683 non-null    int64
 4   Cumulative Impact Probability 683 non-null    float64
 5   Asteroid Velocity             683 non-null    float64
 6   Asteroid Magnitude            683 non-null    float64
 7   Asteroid Diameter (km)        683 non-null    float64
 8   Cumulative Palermo Scale      683 non-null    float64
 9   Maximum Palermo Scale         683 non-null    float64
 10  Maximum Torino Scale          683 non-null    object
dtypes: float64(6), int64(3), object(2)
memory usage: 58.8+ KB
```

FIGURE 9. Dataset 2 -Impacts file description

4.2.2 Data Cleaning and Exploration

- As there were no NULL values in the data, not much cleaning was required. Only the columns were renamed accordingly and some unnecessary columns were dropped.
- The possible impacts variable was plotted with Asteroid Velocity and Asteroid Magnitude to study the relationship between the two. The plots are shown in the figures 10 and 11.

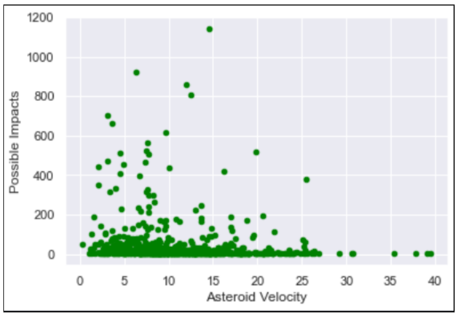


FIGURE 10. Possible Impacts vs Asteroid Velocity

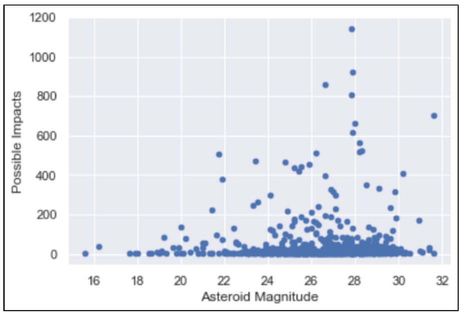


FIGURE 11. Possible Impacts vs Asteroid Magnitude

- The number of asteroids by category is depicted by the figure 12

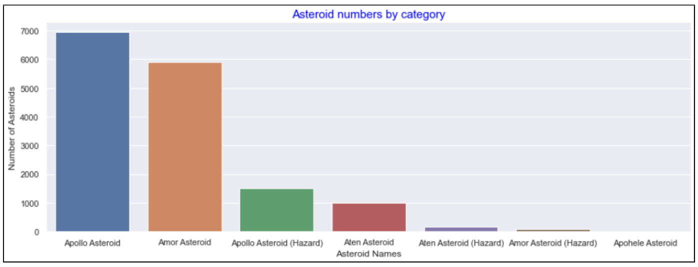


FIGURE 12. Number of asteroids by category

4.2.3 Feature Selection and Engineering

- The correlation between different variables in the merged dataframe is shown in the figure 13. As we can see from the figure, some of the correlation values are very high. For example, the correlation value between the asteroid magnitude and asteroid diameter is -0.6. These high correlation variables were removed from the dataframe before applying the machine learning models.

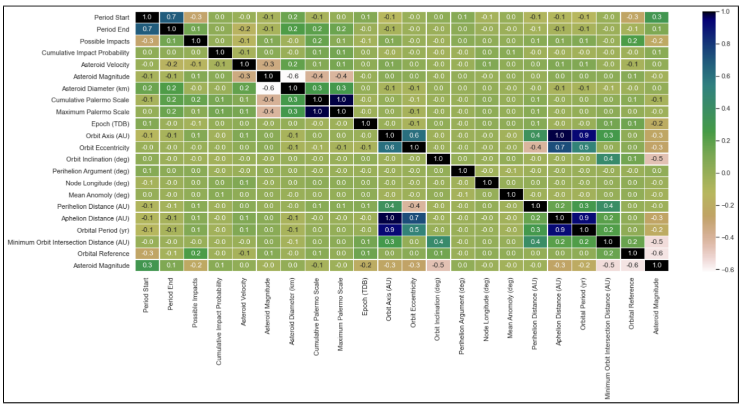


FIGURE 13. Correlation heat map

- The two scales used in this data - Maximum Palermo scale and Maximum Torino Scale are used to define the level of impact hazard of the near earth objects(NEO). These scales are used by NASA to classify the objects as hazardous and not hazardous. But since these variables will not be of use in our model building, these will also be removed before deploying the machine learning models.
- An interesting feature of asteroid impacts can be noticed from the figure 14. It shows that as the period of objects increases, the no. of possible impacts also tend to increase.
- The asteroids with the highest diameter and cumulative impact probability were extracted form th data frames and are shown in the figures 15 and 17 respectively.

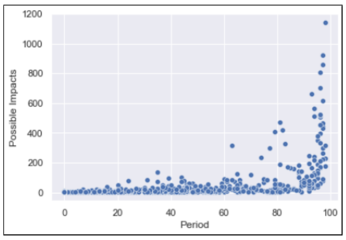


FIGURE 14. Possible impacts vs Period

```
2011 SR52
Object Name                         2011 SR52
Period Start                             2034
Period End                               2115
Possible Impacts                            4
Cumulative Impact Probability         7.6e-10
Asteroid Velocity                       13.55
Asteroid Magnitude                       15.6
Asteroid Diameter (km)                  2.579
Cumulative Palermo Scale                -4.35
Maximum Palermo Scale                   -4.59
Maximum Torino Scale                        0
Period                                     81
Name: 173, dtype: object
```

FIGURE 15. Object with largest diameter

```
2010 RF12
Object Name                         2010 RF12
Period Start                             2095
Period End                               2115
Possible Impacts                           52
Cumulative Impact Probability           0.065
Asteroid Velocity                         5.1
Asteroid Magnitude                       28.4
Asteroid Diameter (km)                  0.007
Cumulative Palermo Scale                 -3.2
Maximum Palermo Scale                    -3.2
Maximum Torino Scale                        0
Period                                     20
Name: 568, dtype: object
```

FIGURE 16. Object with highest cumulative impact probability

4.3 Dataset 3: Exoplanet Classification

4.3.1 Data Description and Overview

The data overview is shown in the figure 17. As we can see some columns like $'koi_teq_err1'$ contain a lot of NULL values(figure 18. But we won't be needing these columns and they will be removed in the subsequent steps.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9564 entries, 0 to 9563
Data columns (total 50 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   rowid              9564 non-null   int64
 1   kepid              9564 non-null   int64
 2   kepoi_name         9564 non-null   object
 3   kepler_name        2294 non-null   object
 4   koi_disposition    9564 non-null   object
 5   koi_pdisposition   9564 non-null   object
 6   koi_score          8054 non-null   float64
 7   koi_fpflag_nt      9564 non-null   int64
 8   koi_fpflag_ss      9564 non-null   int64
 9   koi_fpflag_co      9564 non-null   int64
 10  koi_fpflag_ec      9564 non-null   int64
 11  koi_period         9564 non-null   float64
 12  koi_period_err1    9110 non-null   float64
 13  koi_period_err2    9110 non-null   float64
 14  koi_time0bk        9564 non-null   float64
 15  koi_time0bk_err1   9110 non-null   float64
 16  koi_time0bk_err2   9110 non-null   float64
 17  koi_impact         9201 non-null   float64
 18  koi_impact_err1    9110 non-null   float64
 19  koi_impact_err2    9110 non-null   float64
 20  koi_duration       9564 non-null   float64
 21  koi_duration_err1  9110 non-null   float64
 22  koi_duration_err2  9110 non-null   float64
 23  koi_depth          9201 non-null   float64
 24  koi_depth_err1     9110 non-null   float64
 25  koi_depth_err2     9110 non-null   float64
 26  koi_prad           9201 non-null   float64
 27  koi_prad_err1      9201 non-null   float64
 28  koi_prad_err2      9201 non-null   float64
 29  koi_teq            9201 non-null   float64
 30  koi_teq_err1       0 non-null      float64
 31  koi_teq_err2       0 non-null      float64
 32  koi_insol          9243 non-null   float64
 33  koi_insol_err1     9243 non-null   float64
 34  koi_insol_err2     9243 non-null   float64
 35  koi_model_snr      9201 non-null   float64
 36  koi_tce_plnt_num   9218 non-null   float64
 37  koi_tce_delivname  9218 non-null   object
 38  koi_steff          9201 non-null   float64
 39  koi_steff_err1     9096 non-null   float64
 40  koi_steff_err2     9081 non-null   float64
 41  koi_slogg          9201 non-null   float64
 42  koi_slogg_err1     9096 non-null   float64
 43  koi_slogg_err2     9096 non-null   float64
 44  koi_srad           9201 non-null   float64
 45  koi_srad_err1      9096 non-null   float64
 46  koi_srad_err2      9096 non-null   float64
 47  ra                 9564 non-null   float64
 48  dec                9564 non-null   float64
 49  koi_kepmag         9563 non-null   float64
dtypes: float64(39), int64(6), object(5)
memory usage: 3.6+ MB
```

FIGURE 17. Data description for dataset 3

```
                   Total     Percent
koi_teq_err1        9564  100.000000
koi_teq_err2        9564  100.000000
kepler_name         7270   76.014220
koi_score           1510   15.788373
koi_steff_err2       483    5.050188
koi_slogg_err2       468    4.893350
koi_slogg_err1       468    4.893350
koi_srad_err1        468    4.893350
koi_steff_err1       468    4.893350
koi_srad_err2        468    4.893350
koi_time0bk_err2     454    4.746968
koi_impact_err1      454    4.746968
koi_impact_err2      454    4.746968
koi_period_err1      454    4.746968
koi_duration_err1    454    4.746968
koi_duration_err2    454    4.746968
koi_depth_err2       454    4.746968
koi_depth_err1       454    4.746968
koi_time0bk_err1     454    4.746968
koi_period_err2      454    4.746968
koi_teq              363    3.795483
koi_prad_err2        363    3.795483
koi_prad_err1        363    3.795483
koi_prad             363    3.795483
koi_depth            363    3.795483
koi_model_snr        363    3.795483
koi_steff            363    3.795483
koi_slogg            363    3.795483
koi_impact           363    3.795483
koi_srad             363    3.795483
koi_tce_plnt_num     346    3.617733
koi_tce_delivname    346    3.617733
koi_insol_err1       321    3.356336
koi_insol_err2       321    3.356336
koi_insol            321    3.356336
koi_kepmag             1    0.010456
-------------
```

FIGURE 18. Percentage of Null values in the columns

Target variable: The target variable is the 'planetdisposition'. It is a categorical variable with 3 possible values: 'Confirmed', 'Candidate', 'False Positive' corresponding to the three categories in which the host star can be categorised. 'Confirmed' means that the object is indeed a host star harboring an exoplanet. 'Candidate' means that the object has passed all the tests used for identifying false positives. 'False Positive' means that the object is not an exoplanet.

4.3.2 Data Exploration and Cleaning

- The data has 21 numerical variables. 12 columns contained NULL values after removing the unnecessary columns from the dataframe. Now these values had to be re engineered very carefully. The KSO identifies an exoplanet by analyzing the light coming from it's host star. If the orbiting object shows dips and rises in intensity of lights(since a planet orbits it's host star), then it is considered as a potential candidate. Now if were to replace these NULL values with median or mean, that could alter the outcome of the classification. So they were removed in two steps. First the NULL values were removed from the column containing the highest no. of NULL values i.e 'impact parameter' and then the remaining three columns 'planet disp confidence', 'koi tce plnt num' and 'kepler magnitude' were assigned the median values to get rid of the NULL values.

- For the same reason(discussed in previous point), the outliers were also not removed from the data.
- The categories of the target variable 'planet disposition' were encoded as follows to make classification predictions.
  - 'Confirmed' : 0
  - 'Candidate' : 1
  - 'False Positive' : 2
- The outliers plots are given in figures 19 to 23

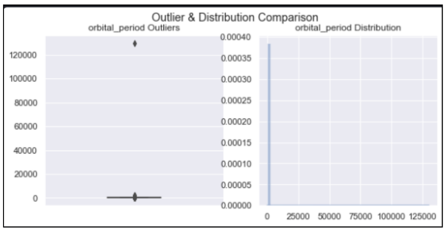


FIGURE 19. Outliers for dataset 3-(1)

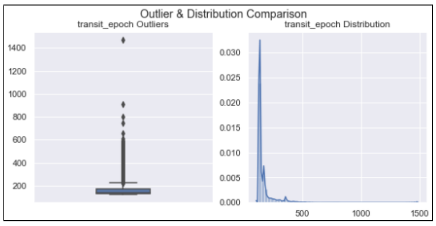


FIGURE 20. Outliers for dataset 3-(2)

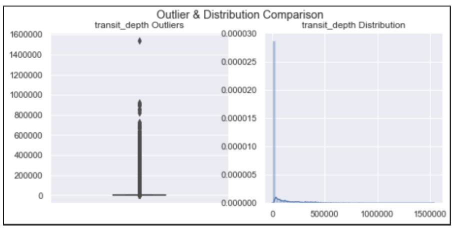


FIGURE 21. Outliers for dataset 3-(3)

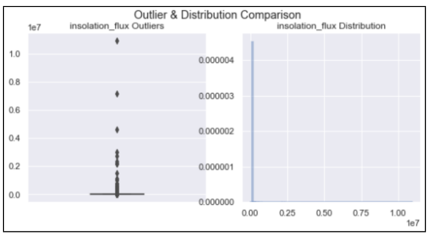


FIGURE 22. Outliers for dataset 3-(4)

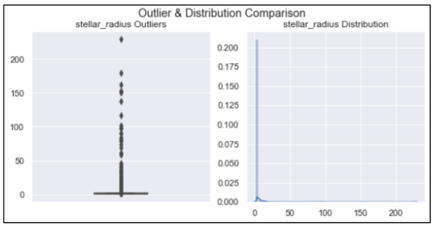


FIGURE 23. Outliers for dataset 3-(5)

- The distribution of the target variable categories is depicted in figure 24

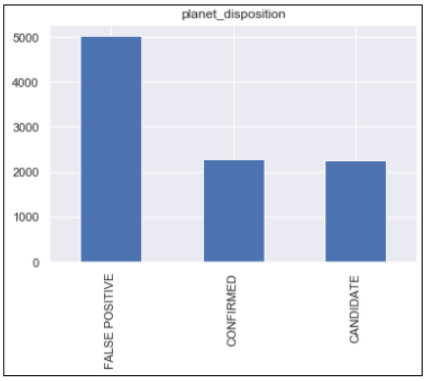


FIGURE 24. Distribution of target varibale categories

4.3.3 Feature Selection

The correlation heat map of the variables is given in the figure 25. In this dataset, no feature was engineered to avoid the alteration in the planet classification outcome.

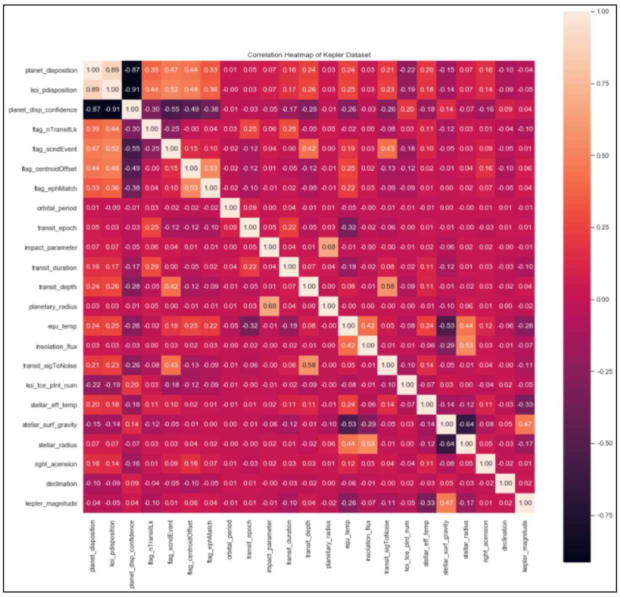


FIGURE 25. Correlation heat map for dataset 3

4.3.4 Model Selection

Since the task in hand is a classification task, two classifiers model - KNN classifier and Naive Bayes classifier were used to predict the outcome. The models were evaluated on the basis of accurcay, F1 score, precision, AUC value and ROC curve.

5. MODELLING AND PREDICTIONS

5.1 Dataset 1: Solar radiation Prediction

For this dataset, the additional hyperparameter tuning was done to select the most important feature for model building. the outputs for hyperparameter tuning are displayed in figure 26. After that a cross validation procedure was run to check if the parameters were overfitting the model or not.

5.1.1 Model 1: Random Forest Regressor

- The mean squared error for training, test and cross validation procedures is given in the figure 27. The model achieves the least value of MSE for training set.
- The R^2 values for training and testing sets is given in figure 28. Both have the same value of 0.77.

```
Best hyperparameters for Random Forest:
 {'max_depth': 7, 'max_features': 'log2', 'min_samples_leaf': 0.025, 'n_estimators': 500}
```

FIGURE 26. Hyperparamter tuning for random forest model on dataset 1

```
Cross Validation MSE for Random Forest:23167.14
Train MSE for Random Forest:22802.18
Test MSE for Random Forest:23363.53
```

FIGURE 27. Mean squared error for random forest regressor

```
Random Forest, R^2 score training set:0.77
Random Forest, R^2 score test set:0.77
```

FIGURE 28. R squared for random forest

5.1.2 Model 2: Gradient Boosting

- The mean squared values for test, training and validation procedures is given in the figure 29. The training set has the lowest value for the MSE among the three.
- The R squared values are given in the figure 30

```
Cross Validation MSE for Random Forest:23167.14
Train MSE for Random Forest:22802.18
Test MSE for Random Forest:23363.53
```

FIGURE 29. Mean squared error for gradient boost

```
Gradient Boosting, R^2 score training set:0.90
Gradient Boosting, R^2 score test set:0.89
```

FIGURE 30. R squared for gradient boost

5.1.3 Model Analysis

Among both the models, gradient boost clearly gives the best result for the prediction of our target variable s it has less values for R^2 and the Mean Squared Error(MSE). The feature importance comparison of both the models is given in the figure 31. As we can see, the temperature variable is the most important feature for both the models which is expected, since the level of solar radiation has to depend on the outside temperature.

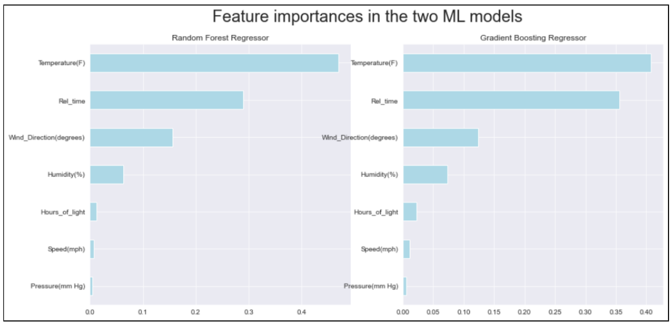


FIGURE 31. Feature importance analysis for random forest and gradient boosting models

5.2 Dataset 2: Asteroid Impact Prediction

5.2.1 Model 1: Stochastic Gradient Descent

- The stochastic gradient descent achieved an overall score of 0.99.(refer to figure 32) which can be considered optimum given the paramters.

5.2.2 Model 2: Decision Tree

- The accuracy of the decision tree was found to be 0.781(Refer to figure 33). This value is not optimum, but it can be further improved by cross validation procedures and hyper parameter tuning.

```
sdg = SGDRegressor()
sdg.fit(Xtrain, ytrain)
sgd = sdg.predict(Xtest)
print(sdg.score(Xtest, ytest))

0.9999999437256196
```

FIGURE 32. Overall score for Stochastic gradient descent algorithm

```
decision_tree = DecisionTreeClassifier()
decision_tree.fit(Xtrain, ytrain)
dt_prediction = decision_tree.predict(Xtest)
accuracy_dt = print('The accuracy of the Decision Tree is', metrics.accuracy_score(dt_prediction,ytest))

The accuracy of the Decision Tree is 0.781021897810219
```

FIGURE 33. Accuracy for decision tree

5.2.3 Model Analysis

Among the two models, Stochastic Gradient Descent gives the best results overall in terms of the prediction of the target variable. this is partially because the stochastic gradient is a modification to the traditional gradient descent algorithms and it can be used to accommodate larger datasets.

5.3 Dataset 3: Exoplanet Classification

5.3.1 Model 1: kNN Classifier

- The k-nearest neighbour classifier model was able to achieve the overall precision of 0.8212. The classification summary is given in the figure 34.
- The kNN Classifier model is correct 75.38% of the time when classifying the target variables as 'CONFIRMED' and 98.7% when classifying the target variable as 'FALSE POSITIVE'.

Overall Precision: 0.8212051366480079

	CONFIRMED	CANDIDATE	FALSE POSITIVE
precision	0.753846	0.533145	0.987080
recall	0.646154	0.656250	0.985171
f1-score	0.695858	0.588327	0.986125
support	910.000000	576.000000	1551.000000

FIGURE 34. KNN classification summary

5.3.2 Model 2: Naive Bayes classifier

- The Naive Bayes achieved the overall precision of 0.822. The classification summary is given in the figure 35.
- The Naive Bayes model is right more than 90% of the time when classifying the target variable as 'CONFIRMED' or 'FALSE POSITIVE', but is only correct 34.2% of the time when it comes to classifying the object as 'CANDIDATE'.

Overall Precision: 0.8221929535726046

	CONFIRMED	CANDIDATE	FALSE POSITIVE
precision	0.928205	0.342736	0.988372
recall	0.617221	0.815436	0.977011
f1-score	0.741423	0.482622	0.982659
support	1173.000000	298.000000	1566.000000

FIGURE 35. Classification Summary for Naive Bayes model

5.3.3 Model Analysis

- Among the two models, Naive Bayes is clearly the better one since it has more success rate when classifying the object as 'Confirmed'. Hence we would have less number of True negatives. But the precision score of both the models is almost equal. Therefore we have to use the ROC curve to compare the two models. This is depicted in the figures 36 and 37.
- As we can see, the AUC value for Naive Bayes is slightly greater than that for kNN classifier. Hence of the two, Naive Bayes is a slightly better model here than kNN classifier.

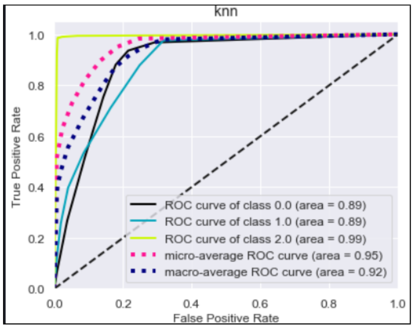


FIGURE 36. ROC for kNN classifier

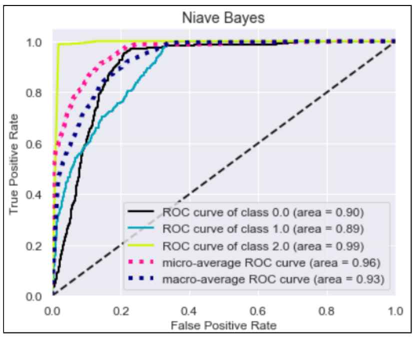


FIGURE 37. ROC for Naive Bayes classifier

6. CONCLUSIONS

In the first dataset, among the random forest regressor and the Gradient Boosting model, the latter gave more promising values for MSE and R^2. The R^2 value for gradient boost was found out to be 0.89. In the second dataset, the stochastic gradient descent achieved an overall score of 0.999. The overall score of a model indicates how efficient he model has performed in accomplishing the given task which in this case the prediction of the probability of the asteroid impacts. On the other hand, the decision tree was only able to achieve the accuracy of 78%. In the third dataset, both the classifier models kNN and Naive Bayes performed nearly equally achieving an overall precision of 0.821 and 0.822 respectively. Later, on evaluating models on the basis of AUC values, Naive Bayes was found to be performing slightly better than the kNN classifier.

7. FURTHER WORK

- The solar radiation prediction task can also be performed by employing neural networks. This idea was discussed in section 2.
- While the prediction of asteroid impact is a very complicated task and requires a lot of variables, it can in principle be achieved using a smaller subset like is our case. Monte Carlo simulations can also be employed to make predictions on impact probabilities. This perspective is also discussed in section 2
- Lastly, for the exoplanet classification task, a larger subset of the data from the Kepler space observatory can be used for making predictions if one wants to improve upon the model accuracy and other parameters.

References

[1] Cyril Voyant et al. "Machine learning methods for solar radiation forecasting: A review". In: Renewable Energy 105 (2017), pp. 569–582.

[2] Veysel Çoban and Sezi Çevik Onar. "Solar Radiation Prediction Based on Machine Learning for Istanbul in Turkey". In: International Conference on Intelligent and Fuzzy Systems. Springer. 2019, pp. 197–204.

[3] Xiaoyan Shao, Siyuan Lu, and Hendrik F Hamann. "Solar radiation forecast with machine learning". In: 2016 23rd International Workshop on Active-Matrix Flatpanel Displays and Devices (AM-FPD). IEEE. 2016, pp. 19–22.

[4] Blakelobato. 'Predicting Asteroid's Diameter Using Machine Learning'. Available: https://medium.com/swlh/predicting-asteroids-diameter-using-machine-learning-e1da883c2196. 2020.

[5] John D Hefele, Francesco Bortolussi, and Simon Portegies Zwart. "Identifying Earth-impacting asteroids using an artificial neural network". In: Astronomy & Astrophysics 634 (2020), A45.

[6] Erika R Nesvold et al. "The Deflector Selector: A machine learning framework for prioritizing hazardous object deflection technology development". In: Acta Astronautica 146 (2018), pp. 33–45.

[7] George Clayton Sturrock, Brychan Manry, and Sohail Rafiqi. "Machine Learning Pipeline for Exoplanet Classification". In: SMU Data Science Review 2.1 (2019), p. 9.

[8] Abhishek Malik, Ben Moster, and Christian Obermeier. "Exoplanet Detection using Machine Learning". In: arXiv preprint arXiv:2011.14135 (2020).

[9] Christopher J Shallue and Andrew Vanderburg. "Identifying exoplanets with deep learning: A five-planet resonant chain around kepler-80 and an eighth planet around kepler-90". In: The Astronomical Journal 155.2 (2018), p. 94.