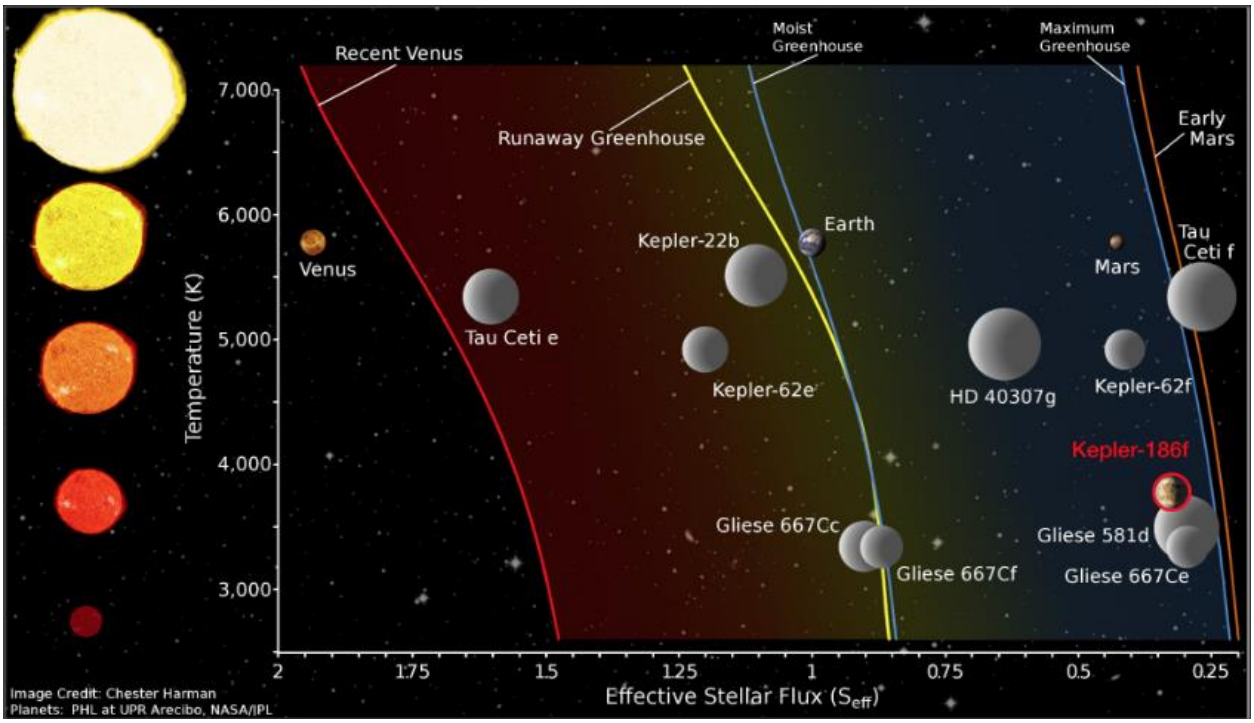
ExoSolar Analysis

Niket Choudhary



2017

Group – Gravity

Exosolar Exploratory Analysis

Summary

One of the most fundamental questions is "*are we alone?*" Last few years only scientists have discovered thousands of planets around other stars, and most of those planets are not like Earth (they're big and gaseous, like Jupiter). And most systems unlike our solar system (big planets orbit close to their parent star, whereas in Earth's solar system, the large planets orbit further out).

The ultimate goal is to see whether the exoplanet can be termed with possible life which we know how to interpret. In habitable zone (Goldilocks zone), the region around a star where a planet with sufficient atmospheric pressure can maintain liquid water on its surface, life has most chance of arising. With the help of this dataset we have tried to analyze these planets, their host star, their size, mass, radius , etc. We have not taken into considerations the atmospheric conditions in planets, and their location inside the galaxy.

Exoplanet definition: An exoplanet is a planet outside our solar system, usually orbiting another star.

Data

Our first glimpse at planets outside of the solar system we call home came in 1992 when several terrestrial-mass planets were detected orbiting the pulsar PSR B1257+12. In this dataset, we can explore by analyzing the characteristics of all discovered exoplanets (plus some familiar faces like Mars, Saturn, and even Earth). Data fields include planet and host star attributes, discovery methods and date of discovery.

Dataset Overview

This data was taken from https://github.com/OpenExoplanetCatalogue/oec_tables/tree/master/comma_separated. The dataset contains 3584 observations. An other data with habitable planets was taken from http://openexoplanetcatalogue.com/systems/?filters=habitable. We have merged the both these datasets.

Variables

PlanetIdentifier	Name given to the planet
TypeFlag	TypeFlag==0,'no known stellar binary companion' TypeFlag==1,'P-type binary (circumbinary)' TypeFlag==2,'S-type binary' TypeFlag==3,'orphan planet'
PlanetaryMassJpt	Mass of planet (Jupiter mass = 1)

RadiusJpt	Radius of Planet (Jupiter Mass = 1)
PeriodDays	To rotatate 1 round around it's parent star
SemiMajorAxisAU	Distance from Sun to Earth = 1AU
Eccentricity	measure of the extent of a deviation of a curve or orbit
PeriastronDeg	the angle nearest to a star in the path of a planet
LongitudeDeg	Mean longitude at a given Epoch (same for all planets in one system)
AscendingNodeDeg	Longitude of the ascending node
InclinationDeg	Inclination of the orbit
SurfaceTempK	Temperature (surface or equilibrium)
AgeGyr	Age Planet or Star
DiscoveryMethod	Discovery method of the planet timing RV transit imaging microlensing
LastUpdated	Date of the last (non-trivial) update
RightAscension	Right ascension
Declination	Declination
DistFromSunParsec	Distance of planet from Sun in Parsecs(1 Parsecs = 3.26 light years)
HostStarMassSlrMass	Mass of Star(mass of Sun = 1)
HostStarRadiusSlrRad	Radius of Star(radius of Sun = 1)
HostStarMetallicity	Stellar metallicity

HostStarTempK	Host Star Temperature
HostStarAgeGyr	Age of Host Star In Billion years
ListsPlanetIsOn	Confirmed planets Confirmed planets, Orphan planets Confirmed planets, Planets in binary systems, P-type Confirmed planets, Planets in binary systems, P-type, Planets in globular clusters Confirmed planets, Planets in binary systems, S-type Confirmed planets, Planets in open clusters Controversial Controversial, Planets in binary systems, P-type Controversial, Planets in binary systems, S-type Kepler Objects of Interest Planets in binary systems, S-type, Confirmed planets Retracted planet candidate Solar System
Probability_of_life	For probability of Life = 1 For no probability of Life = 0

Scope Of Inference – Generalizability

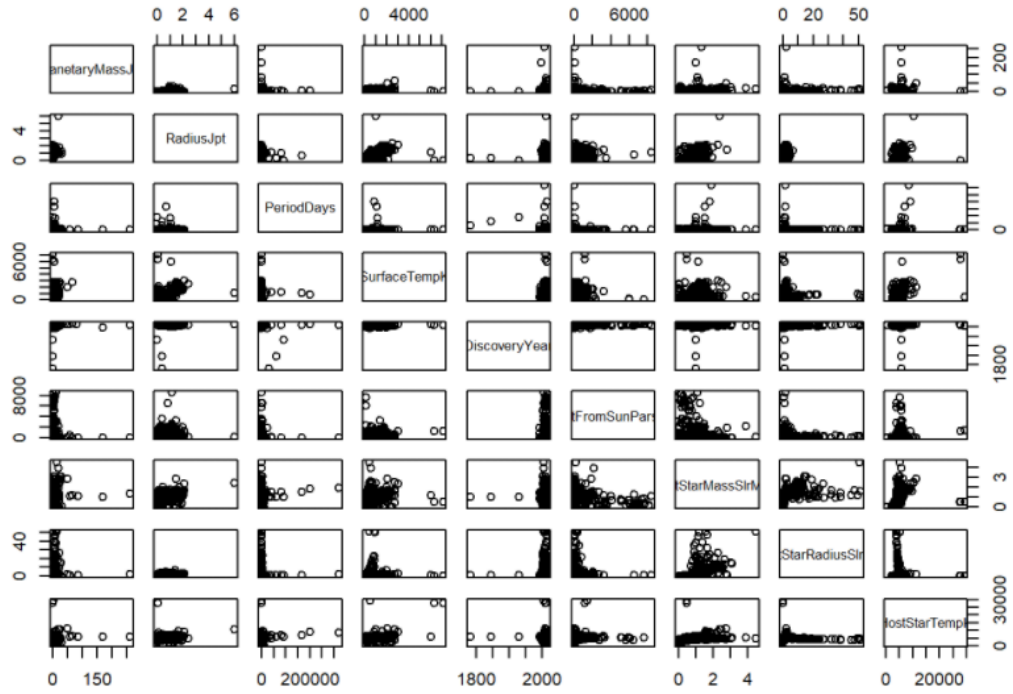
The scope of inference or the population of interest here are the Exoplanets that are discovered using multiple types of imaging techniques. Given the variables such as the Age of star, or disatnce of planet from host star, or period of days, etc. we can predict other variables. The thing to consider here is that the distances are too large to just directly observe any life(the types we know or the one's which are different), so basically astronomers have to go through a lot of data, old and new to check out any new planet under life possibility. Usually there is a bias towards planets with orbit in habitable zone and earth mass but recent study and vast uncertainity in universe prevents such simple possibilities. So to build a model which could cover a wide margin of dependencies or variables here is our aim. From our exploratory analysis we prove the obvious assumptions for cateogarizing life as we think it can be. In simple terms we observe the relationships between variables using plots and find how each and every variable relates to other.

Scope Of Inference – Causality

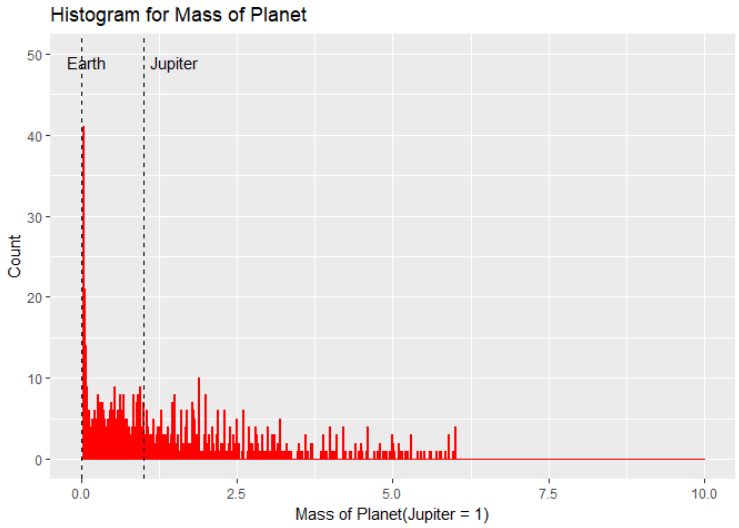
We have considered most of the variables, and we find out how each relates to one or more. We hope to see whether planets considered with Life in our data are nearly earth size, or have a star comparable to our star, or other variables like the semi-major axis has a relation to period of days of rotation. These are some possible inference which we hope to analyze here. We merged an other dataset with a variable with Life = 1 and merged it in our data to perform the analysis.

Exploratory Analysis

After attaching and installing several packages, we loaded the data into exos(for exosolar) dataframe. This dataset has been modified to include a variable Probability_of_life with values 1 for life and 0 for no life. Looking at the strusture of our dataframe we see that most of variables are numerical, except for few which are factors and integers. “num” denotes that the variable “count” is numeric (continuous), and “Factor” denotes that the variable “spray” is categorical with 6 or more categories or levels, and "int" denotes that the variable "count" is numeric(discrete). For more information on each variable we got the summary. Here we saw that there are many values which are NA. Now either we had to replace these NA values, or delete them, or get out our results without doing anything. We will ultimately be using a package Amelia to fill out the missing values which we will discuss in following section. We saw that we have 3584 observations spread around 25 variables.



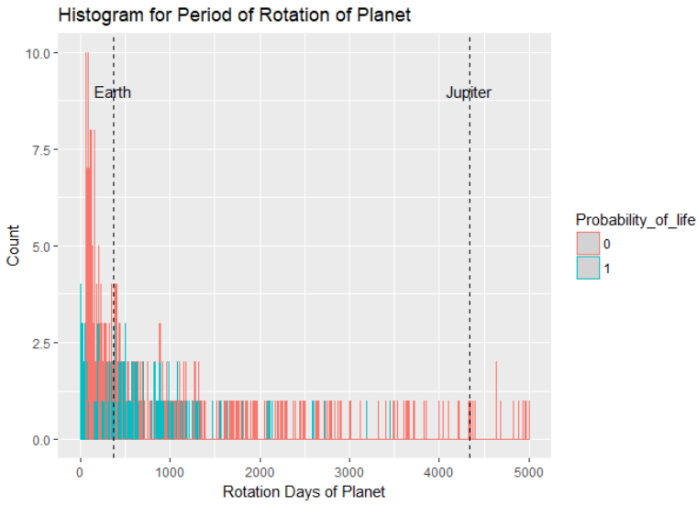
In the above basic Scatterplot Matrix we see relationships between the given variables. But we explored them in detail.

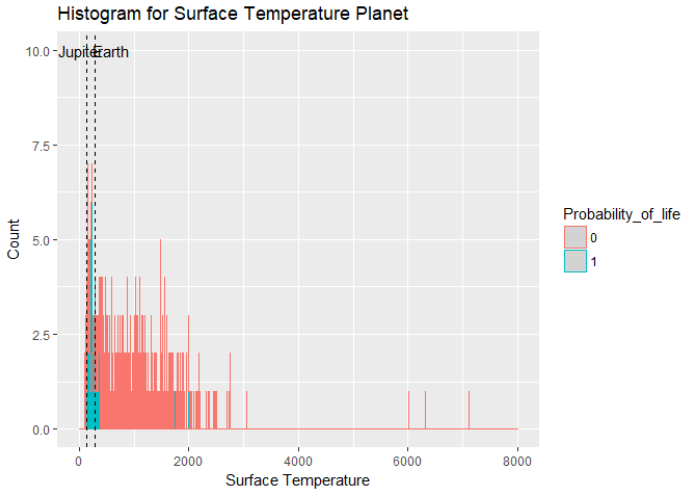


The plot of the mass with the outliers was not very useful. But the main bulk of the planets is within the much smaller range, so removed the outliers. There are still many extremely massive planets . But still, the largest is around Earth's mass.

Most of the planets have less roation cycle as they tend to be close to their star. Our telescopes are still in infancy to locate planets with more number of rotation.

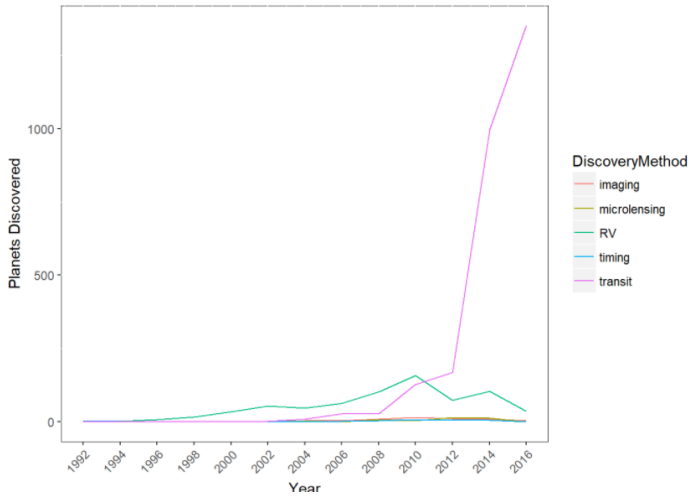
Also most planets with Life probability take less time rotating around their star, it's true, as most planets were found close to dwarf stars, also habitable zone is always nearby a parent star.





More planets observed have much hotter surface temperatures. We can say that they are much closer to their host star. They cannot be mostly habitable.

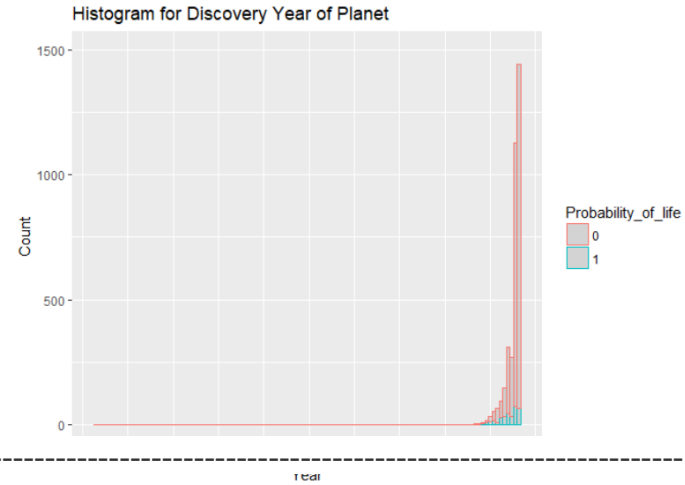
Also most planets with Life probability have close to Earth's surface temperature.

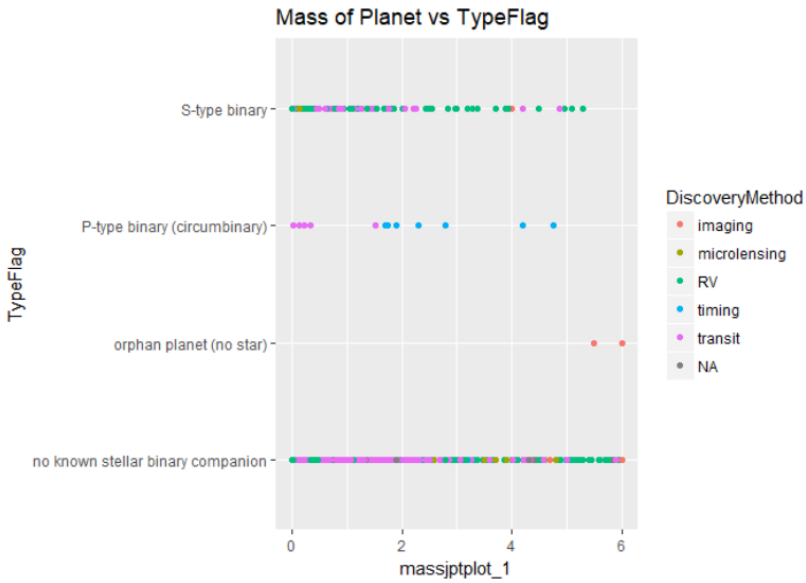


Plot between Discovery Method and Discovery Year

More planets were observed using transit method, as our telescpoes(mainly Kepler) are better at observing the dip in brightness. Also RV found second most amount of planets. And observations are higher recently all thanks to Kepler and K2 missions.

After the launch of Kepler Space Telescope and after 2013(when Kepler's second reaction wheel failed and that the mission had to be replanned) we observed more planets.
Also planets with Life probability were mostly discovered recently due to the advancements in telescopes.



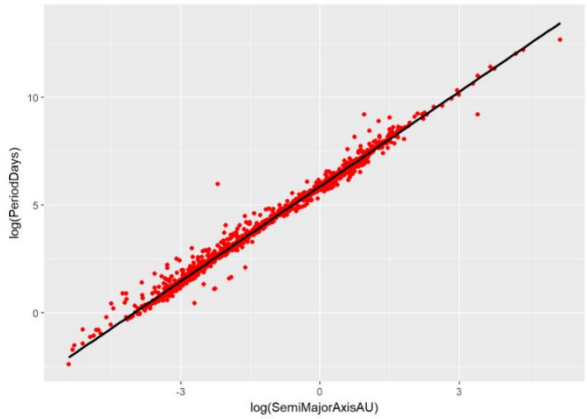


So we can see that our star is an odd compared to others, as most star have a stellar companion.

Planets with higher mass are easily spotted by Transit method.

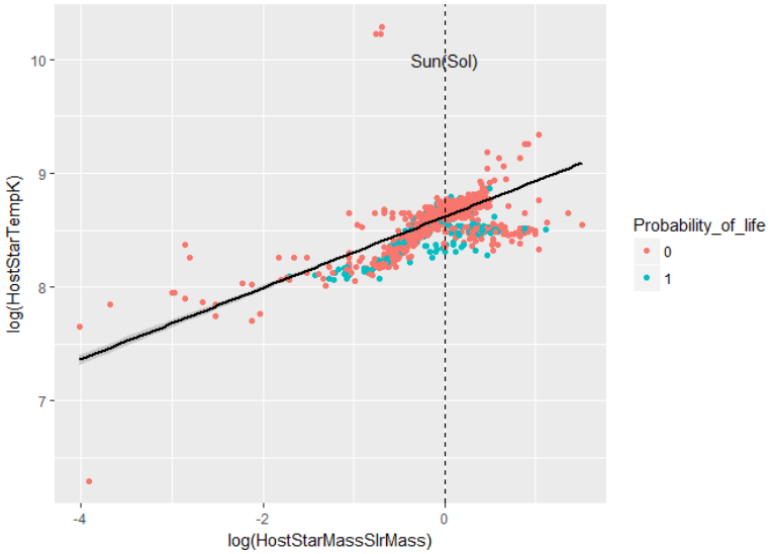
Massive planets are less which are found orbitting P-type binary(both stars) which is true as more massive the planet, more chace it has to destablize the system, or collide with parent star.

Most of the planets are observed by transit Method.



Relationship between SemiMajor Axis and Period

The relationship is linear. We can predict the Period of a planet from the length of its semi-major axis. We did log transform to get readable plot.



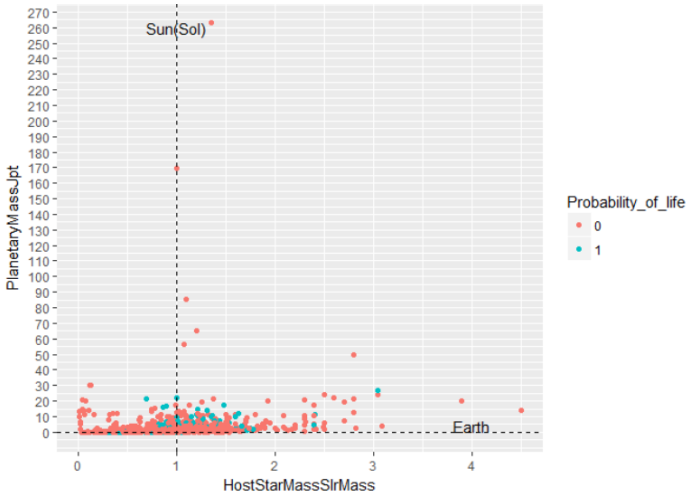
Relationship between star mass and star temperature

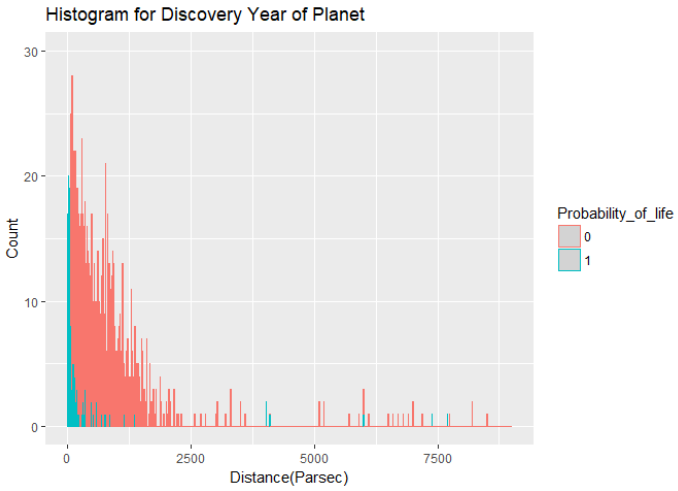
As the mass of the star increases, the temperature of the star also increases.

Also most of the Life probability is around our Sun mass stars.

We did log transform to get readable plot.

We see that most of the planets with some probabolity of life are somewhat massive than Earth.
This is because our telescope technology is not that strong.
Also most of them lie between 0.5 to 2 times our Sun mass
At the cross-section of Sun and Earth most planets with Life probability lie.





So we discover more planets closer to us as our technology is still in infancy.

And obviously our models and telescopes can predict Life closer to us.

Predictive Analysis

After doing the exploratory analysis and looking at how the variables relates to each other now we come on classification. The problem with classifying the planet as with Life or without it includes many variables. Astronomers usually had to look at the other planets with same values, or had to check on some formula. The problem with first part arises because of big dataset, in past they had small data, but now there is a boom in data. For second part having a formula with so many variables will not be perfect. So our aim is to make a model that can classify our data and give us correct prediction.

After the exploratory analysis, where we have observed the relation between attributes, our task is now to make a Model which can classify a Planet with Life(1) or No Life(0). For this we first have to fill all the missing data, as most of the models we are working with don't work well or at all with missing values. We could not remove the missing columns as they were unevenly distributed across whole dataset. We have used Package Amelia for filling out the missing values. As our model will need supervised learning machine algorithm, we are using seven of them here :

KNN Model - When we need classification for a new data, the KNN algorithm goes through the entire dataset to find the k-nearest values to the new data values, or the k number of values most similar to the new record, and then outputs the mode (most frequent class) for a classification problem. The value of k is specified by user.

Decision Tree Model - it is a type of supervised learning algorithm (having a pre-defined target variable). It works for both categorical and continuous input and output variables. Here we split the population or sample into two or more homogeneous set based on most significant splitter / differentiator in input variables.

Random Forest Model - Random Forest (multiple learners) is an improvement over bagged decision trees (a single learner). It can handle large data set with higher dimensionality. It can handle thousands of input variables and identify most significant variables so it is considered as one of the dimensionality reduction methods.

Naive Bayes - To calculate the probability that an event will occur, given that another event has already occurred, we use Bayes' Theorem. To calculate the probability of an outcome given the value of some variable. Naive Bayes can handle missing data. 'naive' because it assumes that all the variables are independent of each other. This should not work very well with our model. But let's look at it too.

Logistic Regression Model - Logistic regression predictions are discrete values(Life or no Life). ITthe output is in the form of probabilities of the default class. As it is a probability, the output lies in the range of 0-1. The output y-value is generated by log transforming the x-value. Then we force this probability into a binary classification.

GBM Model - A boosting algorithm.It is a machine learning technique for regression and classification problems.It produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.

Extreme Gradient Boosting, XGBoost Model - It does parallel computation on a single machine. This makes xgboost at least 10 times faster than existing gradient boosting implementations. It supports various objective functions, including regression, classification and ranking. It only works with numeric vectors.

Analysis

KNN

We checked for correlation. We saw very less correlation between the variables, so we don't have to remove any. Probability_of_life will be our variable for classification in all the models.

The biggest gap we saw is in the using most of the variables in the distance calculation. Taking out some insignificant variables and experimenting with K value could result in increased accuracy.

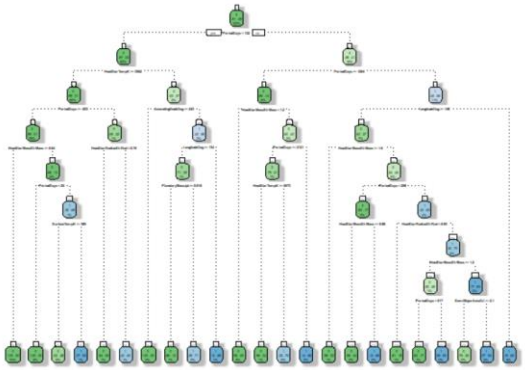
```
## Confusion Matrix
##
##                  test_1_label
## life_predicted_1   0   1
##                 0 652  59
##                 1   2   3
```

A Kappa value of 0.0776 is very low, hence agreement is poor.

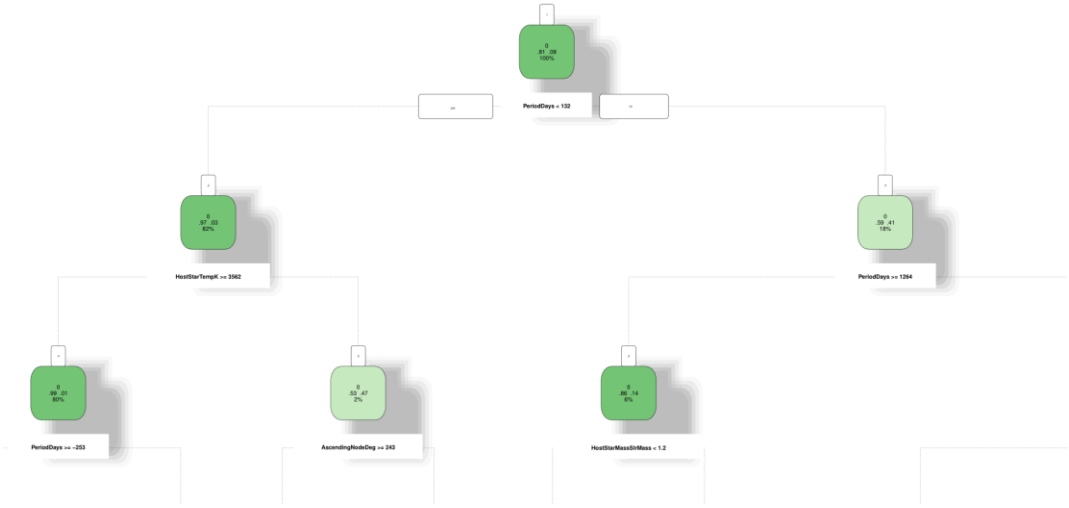
We get an accuracy of 91.48% which is really good, but has a room for improvemnet.

Decision Trees

```
library(rpart) #rpart for "Recursive Partitioning and Regression Trees" and uses the CART decision tree algorithm.
```



We can see here that the model has considered 'Peiod of Days < 132' for split. Decision tree splits the nodes on all available variables and then selects the split which results in most homogeneous sub-nodes. Our Root Node has a ratio of .91 to .09 on 0(No Life). The consecutive Decision Nodes are HostStarTempK and PeriodDays>=1254. We have 23 Terminal Nodes. Let's see the part after zooming.

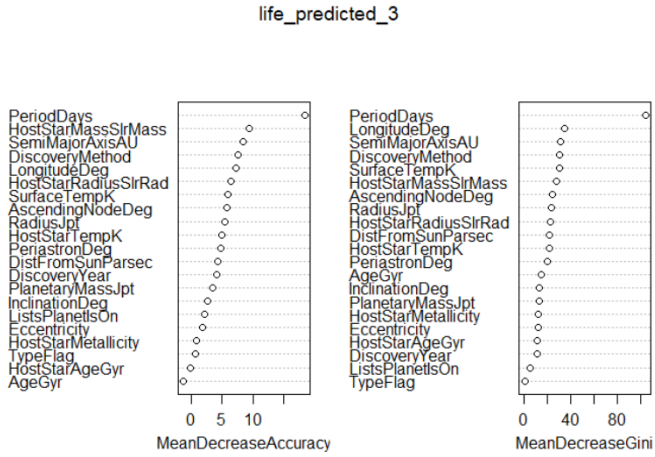


```
## Confusion Matrix
##
##              test_2.test_2_label
## prediction_2   0   1
##             0 634  15
##             1  20  47
```

A Kappa value of 0.7019 is good, hence perfect agreement. We get accuracy of 95.11% which is far better than KNN model. We can prune our model to avoid overfitting if any, or we can jump to Random Forest which betters the accuracy, as it constructs several decision trees on several variables and than does classification.

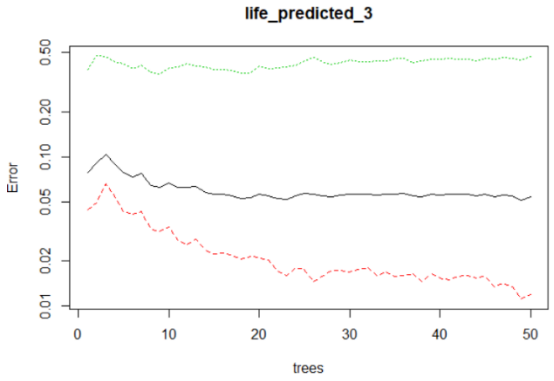
Random Forest

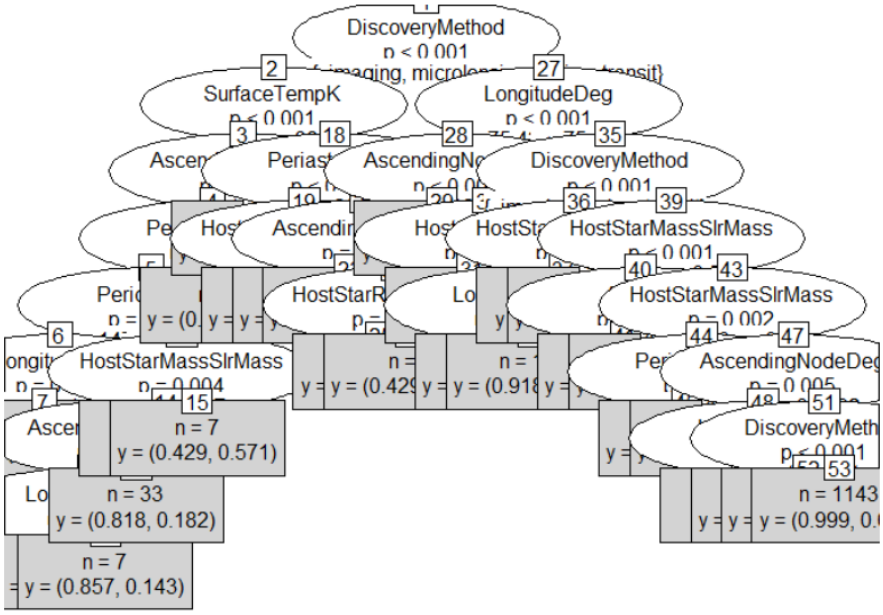
We want enough trees to stabilize the error but not so many that they over correlate the ensemble, which will lead to overfit so we keep ntree = 50.



Higher the value of Gini higher the homogeneity. So split occurs accordingly.

Across 50 trees the error rate decreased, we might increase the number of trees for more decreased error, but avoid it because of overfitting.





Here we see the splitting at Discovery Method. This plot has few good aspects. Like on 20th Terminal Node when the value of HostStarRadius is <=1 , the y has value of 88.4% for Life = 1. Same results can be seen on 25, 37 termnal node, Also for transit we have 99% and transit 92% Life chance = 1.

```
## Confusion Matrix
##
##              test_2.test_2_label
## prediction_3   0   1
##            0 647  25
##            1   7  37
```

A Kappa value of 0.6747 is fine, near perfect agreement.

We see a slight improvement at 95.53%.

NaiveBayes

```
## Confusion Matrix
##
##              test_2.test_2_label
## prediction_4   0   1
##            0 554  27
##            1 100  35
```

A Kappa value of 0.2685 is bad, hence agreement is poor.

We get accuracy of 82.26%. Well this was to happen as Naive Bayes considers variables to be unrelated to each other.

Logistic Regression

```
## Confusion Matrix
##
##              test_2.test_2_label
## prediction_5   0   1
##            0 648  57
##            1   6   5
```

A Kappa value of 0.1139 is bad, hence agreement is poor.

We get an accuracy of 91.2% which is less compred to other models, but it outperformed NaiveBayes. Usually logistic regression performs good for binary classification but with our variables it gives less accuracy compared to other models.

Generalized Boosted Regression Models(GBM)

```
## Confusion Matrix
##
##              test_2.test_2_label
## prediction_6   0   1
##            0 650  17
##            1   4  45
```

A Kappa value of 0.7951 is good, hence agreement is good.

We get an accuracy of 97.07% which is best yet. The more ntrees the more accuracy we observe here, but after that we will see overfitting.

XGBoost

XgBoost accepts the missing values in it's prediction but here we took our exosim dataframe.

We converted our data type to numeric, otherwise XGBoost algorithm doesn't work. It involves defining some parameters. They are as follows

```
parameters <- list(
    # General Parameters
    booster            = "gbtree",          # default = "gbtree"           # gbtree
(tree based) or gblinear (linear function)
    silent             = 0,                 # default = 0                  # silent =
0 will stop results from displaying
    # Booster Parameters
    eta                = 0.3,               # default = 0.3, range: [0,1]  # Low eta
value means model is more robust to overfitting.
    gamma              = 0,                 # default = 0,   range: [0,∞]  # Larger
the gamma more conservative the algorithm is.
    max_depth          = 2,                 # default = 6,   range: [1,∞]  # less
depth so to avoid overfitting
    min_child_weight   = 1,                 # default = 1,   range: [0,∞]  # It might
help in logistic regression when class is extremely imbalanced.
    subsample          = 1,                 # default = 1,   range: (0,1]  # 0.5
means that XGBoost randomly collected half of the data instances to grow trees, this
will prevent overfitting.
    colsample_bytree   = 1,                 # default = 1,   range: (0,1]
    colsample_bylevel  = 1,                 # default = 1,   range: (0,1]
    lambda             = 1,                 # default = 1
    alpha              = 0,                 # default = 0
    # Task Parameters
    objective          = "multi:softmax",   # default = "reg:linear"
    eval_metric        = "mlogloss",
    num_class          = 20,
    seed               = 1234               # reproducability seed
    )
```

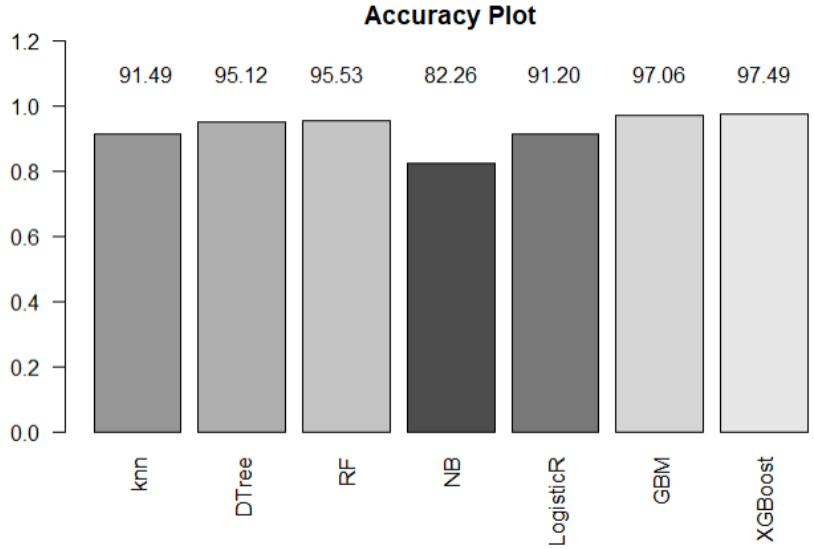
```
## Confusion Matrix
##
##              test_3_label
## prediction_7   0   1
##            0 650  14
##            1   4  48
```

A Kappa value of 0.8286 is best across all models, hence agreement is best here.

We get an accuracy of 97.49% which is best. nrounds = 200 really works, we could further improve accuracy with higher values of nrounds.

Accuracy of our Models:

"Accuracy of KNN Model is: 0.914804469273743"
"Accuracy of Decision Tree Model is: 0.951117318435754"
"Accuracy of Random Forest Model is: 0.955307262569832"
"Accuracy of Naive Bayes is: 0.822625698324022"
"Accuracy of Logistic Regression Model is: 0.912011173184358"
"Accuracy of GBM Model is: 0.970670391061452"
"Accuracy of XGBoost Model is: 0.974860335195531"



We got the highest accuracy using XGBoost model and the Kappa statistic which tells us how much better the measurement system is than random chance is all highest here.

Conclusion

After having a model with 97.48% accuracy we are pretty sure that we can apply it on unseen data. So, we will wait for new data to come by. The best thing about our XGBoost model is that we could further try and increase it's accuracy if we experiment with the parameters. Also we can increase the number of trees.

With our models we saw that all of them considered most of the variables to be significant. Also the categorical variables, being only three in our dataset, played a huge role. We saw with exploratory analysis that finding a planet with life is a hard job as most of the exoplanets are either too large, or in a large orbit, or nearly close to but not in habitable zone, so many factors which can decrease the odds in favour of life. But our models can definitely do a good job.

Future telescopes like James Webb Telescope(JWT), Thirty Metre Telescope, WEBB Space telescope, will give more parameters to work upon. They will generate data on atmosphere, water presence, etc. Which we will use to further the accuracy but with higher confidence.

We can further increase the accuracy of the model if we use Deep Neural Network using H2O, which have proven to be better by people in the field, so this will be our future approach.
