AEDA

Final Project

Raúl García Gómez

Marc Monclús

Table of contents

Report

For the realization of this final project of data analysis, a database from NASA of discovered planets has been used with enough data in columns to meet the specifications of having:

- ID
- Name of the planet
- One binary categorical variable:
  - Controversial Flag (originally an integer class converted to a Boolean)
- 2 polyatomic categorical variables (or more):
  - Discovery method
  - Discovery Year
  - Discovery Facility
  - Spectral Type
- Seven numerical variables.
  - Number of Stars
  - Number of Planets
  - Orbital Period [days]
  - Planet Radius [Earth Radius]
  - Planet Mass or Mass*sin(i) [Earth Mass]
  - Eccentricity
  - Equilibrium Temperature [K]
  - Distance (parsec)
  - Orbit Semi-Major Axis

Dataset origin

The database has been obtained from the NASA Exoplanet Archive in the following link:

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

First of all, the database that contains the link is made up of a large number of columns that we consider prior to downloading in .csv format that are not necessary and it is decided to dispense with them even before downloading the data set itself. Is attached to this project.

Next, so that RStudio can correctly read the data set, the comments are extracted from the file by opening it in notepad and thus saving the csv as a conventional csv (the #comments that correspond to a legend about what each column is are attached in a txt in the present work to understand better the data used).

When importing the database in RStudio, the data that has been considered excessive for the analysis is finished, but. Even so, some more data has been left than is necessarily required for this project due to pure added interest, such as the semi-major axis or some of the variables such as the distance in parsec and the spectral facility.

Exploratory Data Analysis

To treat the data before doing the analysis, as usual, we dispense with the records in the database that contain Not Available (NA) in any of the variables.

Because R doesn't read empty strings of the character variable as NA, we transform them into NA and remove them along the rest of NAs. We do this to eliminate rows that have no complete information on them, especially in the case of the empty character ones because we can't use other alternative methods like filling the blank with the mean of the column.

On the other hand, it is important to eliminate the outliers to facilitate the observation of the results without having values that are excessively distant from the group that makes up the records of the same variable.

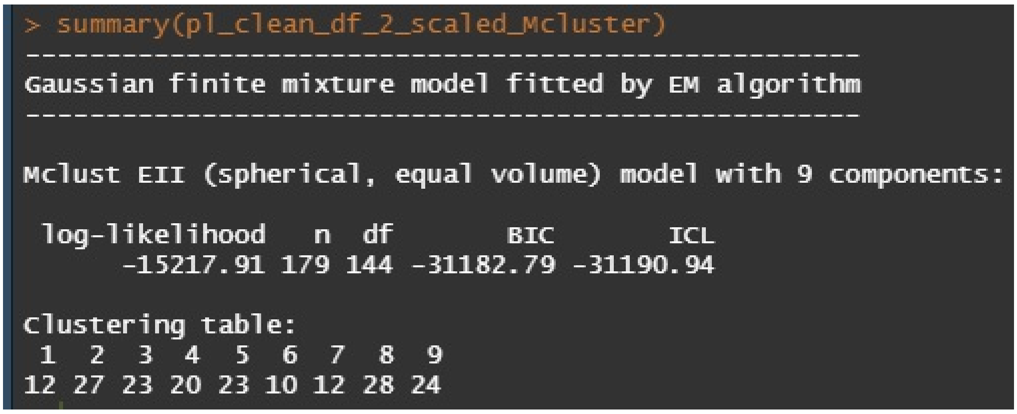
The model is not separated into training and test parts because it is unsupervised and we do not know what kind of prediction we would do with it, we are just looking to find a common denominator that would tie these exoplanets into logical and relevant groups.

Model based and Partitioning Around Medoids (PAM) algorithms:

Due to the enforced addition of binary and categorical variables within the dataset, the number of algorithms available to us is reduced because we can't offer a numerical matrix to the models, requisite for some like k-means. Because of this we used model based and PAM algorithms.

However, even if it can't be done with the binary and categorical values, we thought it was important to scale the dataset. It is especially true in this one because numerical values in the field of astronomy tend to be either really small or really big. Running the summary function and checking the quartiles and means of the variables is an easy way of seeing the big difference in range inside a variable and compared to the others. That's why we split the dataset, take only the numerical variables, scale them and bring the dataset back together. With this we can somewhat offset the impact that big numerical values would have on the resulting models.

The model based clustering algorithm is run first. The result is that a model with 9 clusters would be the more accurate (the summary function of the result of the mclust function gives by default the best model) :



```
> summary(pl_clean_df_2_scaled_Mcluster)
----------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
----------------------------------------------------

Mclust EII (spherical, equal volume) model with 9 components:

 log-likelihood   n  df       BIC       ICL
      -15217.91 179 144 -31182.79 -31190.94

Clustering table:
 1  2  3  4  5  6  7  8  9
12 27 23 20 23 10 12 28 24
```

Figure 1 Summary of the model based clustering

The result is pretty bad as it can be seen in the log-likelihood and BIC. Good numbers would be those close to 0, these two are extremely negative numbers. In the case of this model, it might be because the data does not follow a Gaussian proportion.

The plots that we can do with this model and dataset are the BIC plot, the classification plot and the uncertainty plot. The BIC plot is the one that tells us that a 9 components model is the better choice, as it has the highest BIC. The classification and uncertainty plots don’t tell us much more about the data as it could’ve been predicted after seeing the result of the model. They offer a possible insight into the relation of certain variables, like the Orbital Period and the Orbital Semi-Major Axis but not much more can be gleaned from them:

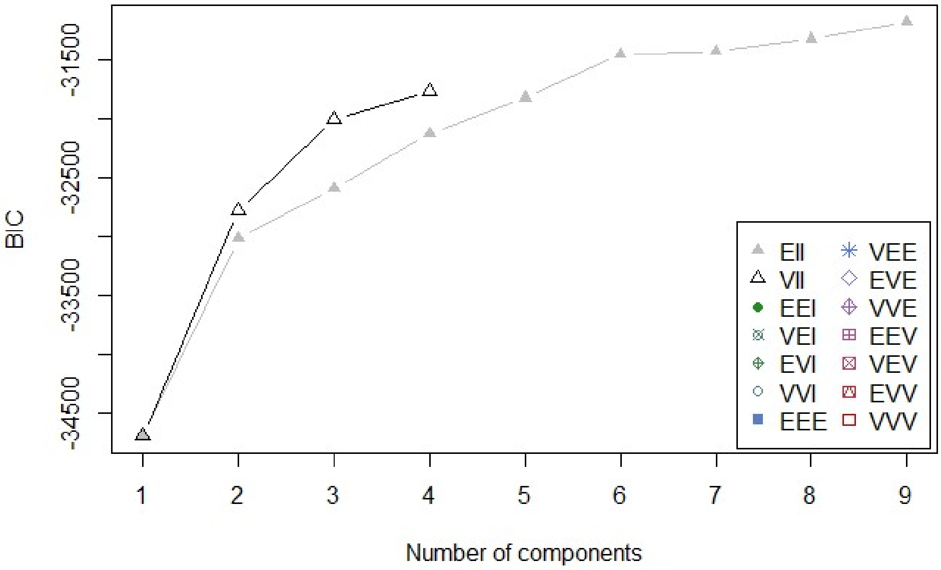


Figure 2 BIC plot of the model based clustering

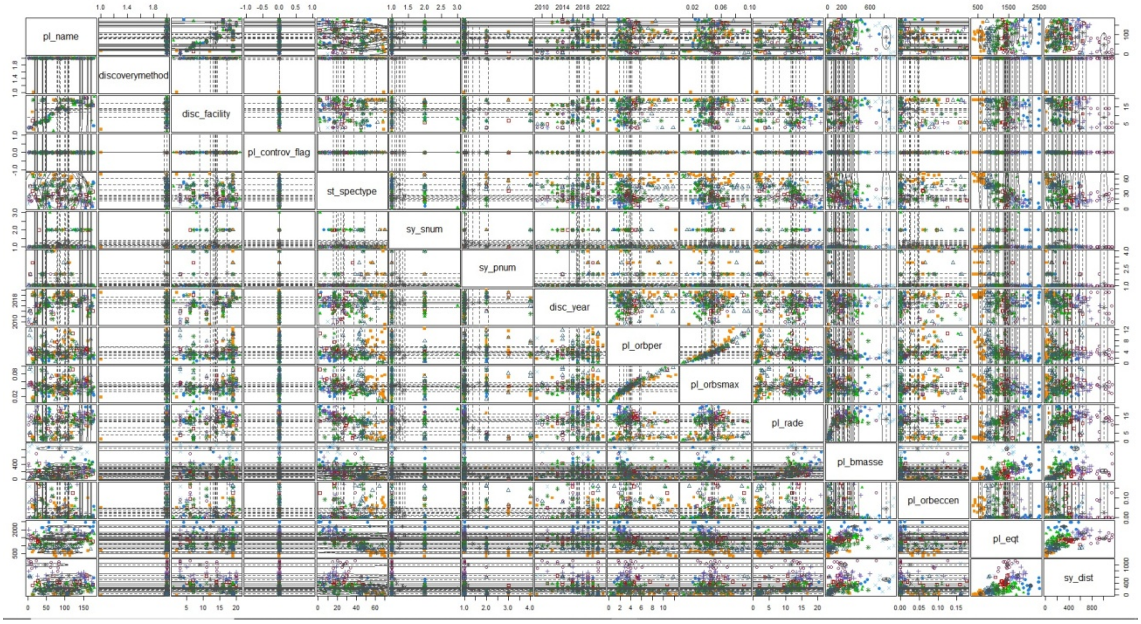


Figure 3 Classification plot of the model based clustering

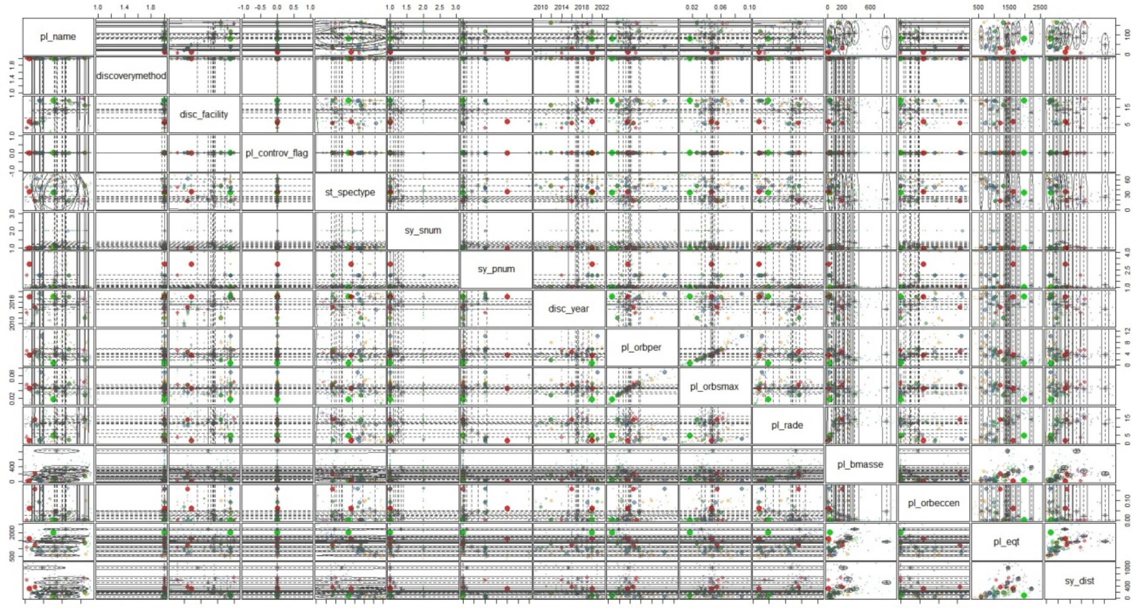
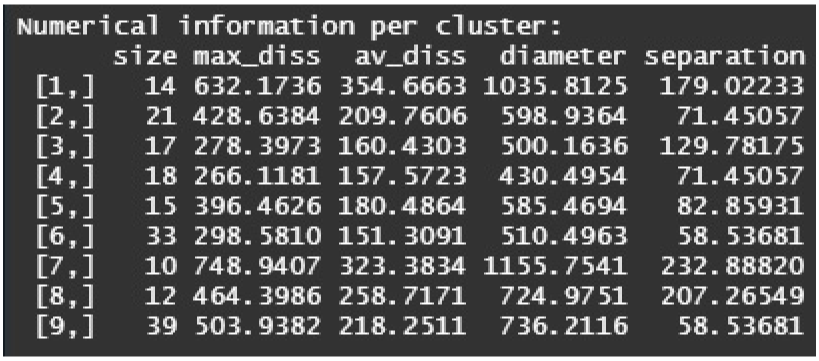


Figure 4 Uncertainty plot of the model based clustering

The choice of using partitional around medoids (PAM) algorithms is due to the sensitivity of the k-medians with the outliers and to the random selection of centroids. A problem in this model, however, is that it must be told the number of clusters it is going to split the dataset in before running the algorithm. This is why we used the model based clustering first, because it we could then use the number of clusters that the model has found as the best, which in our case was 9.

The summary and silhouette plot of the model give us some reasons to hope for better results than in the model based case. The clusters all have big enough sizes leading to believe that there are no anecdotic or outlier clusters and the silhouette plot shows that most classified planets fit within their chosen cluster:



```
Numerical information per cluster:
      size max_diss  av_diss  diameter separation
 [1,]   14 632.1736 354.6663 1035.8125  179.02233
 [2,]   21 428.6384 209.7606  598.9364   71.45057
 [3,]   17 278.3973 160.4303  500.1636  129.78175
 [4,]   18 266.1181 157.5723  430.4954   71.45057
 [5,]   15 396.4626 180.4864  585.4694   82.85931
 [6,]   33 298.5810 151.3091  510.4963   58.53681
 [7,]   10 748.9407 323.3834 1155.7541  232.88820
 [8,]   12 464.3986 258.7171  724.9751  207.26549
 [9,]   39 503.9382 218.2511  736.2116   58.53681
```

Figure 5 Summary of PAM model, cluster size

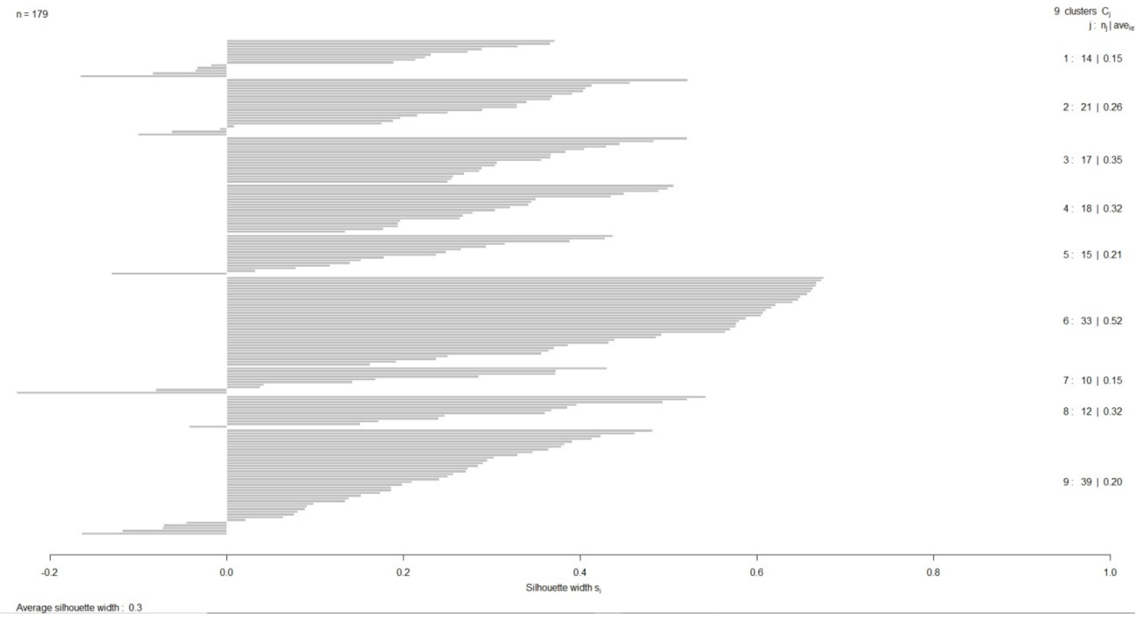


Figure 6 PAM silhouette plot

However when we check the correlation plot we see similar results to the classification plot of the model based clustering algorithm. Even if we add the cluster to the plot, we can see some relations between variables but there is no clear and logical clustering formation:

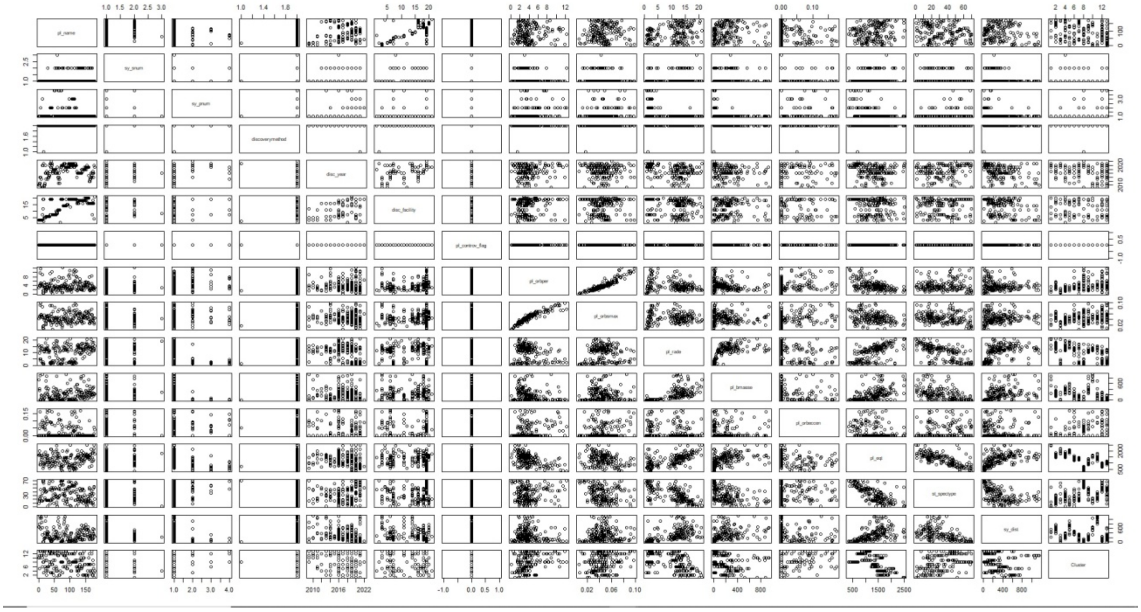


Figure 7 PAM correlation plot

Conclusions:

From these two models we can conclude that there is no clear classification for them. While we can glean some insight into the relations between certain sets of variables, we cannot extract some sort of conclusion that would let us classify these exoplanets into certain groups just through statistical data.

The reasons for these could be various. First the dataset was very incomplete. From the initial 35.000 exoplanets in the database, we extracted 5.000 to use in the dataset after a preliminary filter, and the EDA further decreased this number to the point that maybe there was not enough relevant data.

Another factor could have been our lack of knowledge in the field. Astronomy is an interesting but very complicated field, maybe with more knowledge on it we could've taken other decisions or we could've seen some logic in the models.

R Script

```
## Advanced Engineering Data Analysis FINAL PROJECT

## PLANETARY SYSTEMS DATA ANALYSIS



#Authors:

# Marc Monclús Montalvez    Master's Degree in Space and Aeronautical Engineering

# Raúl García Gómez         Master's Degree in Industrial Engineering



#Professor:

# Daniel Fernandez Martinez



###############################################################################

##### LEGEND

# This file was produced by the NASA Exoplanet Archive  http://exoplanetarchive.ipac.caltech.edu

# Fri Mar 25 12:15:35 2022

#

# See also: https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=PS

#

# User preference: *

#

# CONSTRAINT:  where (default_flag = 1)

# CONSTRAINT:  order by pl_rade desc

#

# COLUMN pl_name:        Planet Name

# COLUMN sy_snum:        Number of Stars

# COLUMN sy_pnum:        Number of Planets

# COLUMN discoverymethod: Discovery Method

# COLUMN disc_year:      Discovery Year

# COLUMN disc_facility:  Discovery Facility

# COLUMN pl_controv_flag: Controversial Flag

# COLUMN pl_orbper:      Orbital Period [days]

# COLUMN pl_orbsmax:     Orbit Semi-Major Axis [au])
# COLUMN pl_rade:        Planet Radius [Earth Radius]
# COLUMN pl_bmasse:      Planet Mass or Mass*sin(i) [Earth Mass]
# COLUMN pl_orbeccen:    Eccentricity
# COLUMN pl_eqt:         Equilibrium Temperature [K]
# COLUMN st_spectype:    Spectral Type
# COLUMN sy_dist:        Distance [parsec]

############################################################################

## Cleaning Data
rm(list = ls())

#Library - Packages import
library(tidyr)
library(dplyr)
library(cluster)
library(ggplot2)
library(factoextra)
library(NbClust)
library(mclust)

## Work Directory (change if needed)
getwd()
setwd("D:/ESEIAAT/Data_Analysis/Final_Project")
plantetary_df <- read.csv("PS_2022.03.25_12.15.35.csv")


# Here we only want specific data, therefore we need to remove the excess.
data_del <- c("default_flag","pl_radj", "pl_bmassj", "pl_bmassprov", "pl_insol",
            "ttv_flag", "pl_insol", "st_teff",
            "st_rad", "st_mass", "st_met",
             "st_logg", "sy_vmag", "sy_kmag", "sy_gaiamag")
```

```
planetary_short_df <- plantetary_df[ , !(names(plantetary_df)%in%data_del)]


## here we delete rows with empty or NA values (tidyr library)


drop_na_df <- planetary_short_df

drop_na_df[drop_na_df == ""] <- NA

drop_na_df <- drop_na_df %>% drop_na(pl_name)

drop_na_df <- drop_na_df %>% drop_na(sy_snum)

drop_na_df <- drop_na_df %>% drop_na(sy_pnum)

drop_na_df <- drop_na_df %>% drop_na(discoverymethod)

drop_na_df <- drop_na_df %>% drop_na(disc_year)

drop_na_df <- drop_na_df %>% drop_na(disc_facility)

drop_na_df <- drop_na_df %>% drop_na(pl_controv_flag)

drop_na_df <- drop_na_df %>% drop_na(pl_orbper)

drop_na_df <- drop_na_df %>% drop_na(pl_orbsmax)

drop_na_df <- drop_na_df %>% drop_na(pl_rade)

drop_na_df <- drop_na_df %>% drop_na(pl_bmasse)

drop_na_df <- drop_na_df %>% drop_na(pl_orbeccen)

drop_na_df <- drop_na_df %>% drop_na(pl_eqt)

drop_na_df <- drop_na_df %>% drop_na(st_spectype)

drop_na_df <- drop_na_df %>% drop_na(sy_dist)


###Other methods that can be used:

#planetary_short2_df<-planetary_short1_df[complete.cases(planetary_short1_df$pl_orbper),]

#planetary_short_df[!is.na(planetary_short_df$pl_orbper)]



#after removing NA's we get a dataframe of 466 rows from 5005 observations


pl_clean_df<-drop_na_df


###############################################################################

#Outliers
```

```
################################################################################
#class of "plantetary_df$default_flag"
class(pl_clean_df$pl_controv_flag)

################################################################################
#we need to convert this column to a binary

pl_clean_df$pl_controv_flag = as.logical(pl_clean_df$pl_controv_flag)
pl_clean_df_2<-pl_clean_df

#Remove outliers
#Some are not checked because the outliers are interesting (like the Controversial flag)
#or because thy cannot be outliers (like discovery center)

#outlierx2<-boxplot.stats(pl_clean_df_2$sy_snum)$out
#outlierx2rows<-which(pl_clean_df_2$sy_snum %in% c(outlierx2))
#outlierx2rows
#pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$sy_snum %in% outlierx2),]

#outlierx3<-boxplot.stats(pl_clean_df_2$sy_pnum)$out
#outlierx3rows<-which(pl_clean_df_2$sy_pnum %in% c(outlierx3))
#outlierx3rows
#pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$sy_pnum %in% outlierx3),]

#outlierx4<-boxplot.stats(pl_clean_df_2$discoverymethod)$out
#outlierx4rows<-which(pl_clean_df_2$discoverymethod %in% c(outlierx4))
#outlierx4rows
```

```
#pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$discoverymethod %in% outlierx4),]


#outlierx5<-boxplot.stats(pl_clean_df_2$disc_year)$out

#outlierx5rows<-which(pl_clean_df_2$disc_year %in% c(outlierx5))

#outlierx5rows

#pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$disc_year %in% outlierx5),]


#outlierx6<-boxplot.stats(pl_clean_df_2$disc_facility)$out

#outlierx6rows<-which(pl_clean_df_2$disc_facility %in% c(outlierx6))

#outlierx6rows

#pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$disc_facility %in% outlierx6),]


#outlierx7<-boxplot.stats(pl_clean_df_2$pl_controv_flag)$out

#outlierx7rows<-which(pl_clean_df_2$pl_controv_flag %in% c(outlierx7))

#outlierx7rows

#pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$pl_controv_flag %in% outlierx7),]


outlierx8<-boxplot.stats(pl_clean_df_2$pl_orbper)$out

outlierx8rows<-which(pl_clean_df_2$pl_orbper %in% c(outlierx8))

outlierx8rows

pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$pl_orbper %in% outlierx8),]


outlierx9<-boxplot.stats(pl_clean_df_2$pl_orbsmax)$out

outlierx9rows<-which(pl_clean_df_2$pl_orbsmax %in% c(outlierx9))

outlierx9rows

pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$pl_orbsmax %in% outlierx9),]


#No outliers in here

outlierx10<-boxplot.stats(pl_clean_df_2$pl_rade)$out

outlierx10rows<-which(pl_clean_df_2$pl_rade %in% c(outlierx10))

outlierx10rows

#pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$pl_rade %in% outlierx10),]
```

```
outlierx11<-boxplot.stats(pl_clean_df_2$pl_bmasse)$out
outlierx11rows<-which(pl_clean_df_2$pl_bmasse %in% c(outlierx11))
outlierx11rows
pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$pl_bmasse %in% outlierx11),]

outlierx12<-boxplot.stats(pl_clean_df_2$pl_orbeccen)$out
outlierx12rows<-which(pl_clean_df_2$pl_orbeccen %in% c(outlierx12))
outlierx12rows
pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$pl_orbeccen %in% outlierx12),]

outlierx13<-boxplot.stats(pl_clean_df_2$pl_eqt)$out
outlierx13rows<-which(pl_clean_df_2$pl_eqt %in% c(outlierx13))
outlierx13rows
pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$pl_eqt %in% outlierx13),]

#outlierx14<-boxplot.stats(pl_clean_df_2$st_spectype)$out
#outlierx14rows<-which(pl_clean_df_2$st_spectype %in% c(outlierx14))
#outlierx14rows
#pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$st_spectype %in% outlierx14),]

outlierx15<-boxplot.stats(pl_clean_df_2$sy_dist)$out
outlierx15rows<-which(pl_clean_df_2$sy_dist %in% c(outlierx15))
outlierx15rows
pl_clean_df_2<- pl_clean_df_2[-which(pl_clean_df_2$sy_dist %in% outlierx15),]

#Scaling for clustering
pl_clean_df_2_scaled <- pl_clean_df_2
pl_clean_df_2_scaling <- pl_clean_df_2
pl_clean_df_2_noscaling <- pl_clean_df_2
pl_clean_df_2_scaling <- pl_clean_df_2[c(2,3,5,8,9,10,11,12,13,15)]
pl_clean_df_2_noscaling <- pl_clean_df_2[c(1,4,6,7,14)]

pl_clean_df_2_scaled <- scale(pl_clean_df_2_scaling)
```

```
pl_clean_df_2_scaled<- cbind.data.frame(pl_clean_df_2_noscaling, pl_clean_df_2_scaling)


#Model based clustering
pl_clean_df_2_scaled_Mcluster <- Mclust(pl_clean_df_2_scaled)
summary(pl_clean_df_2_scaled_Mcluster)
plot(pl_clean_df_2_scaled_Mcluster)


#PAM looking for 9 clusters because model based clustering points to 9 clusters being the
#best choice
pl_clean_df_2_scaled_PAM <- pam(pl_clean_df_2_scaled, 9)
summary(pl_clean_df_2_scaled_PAM)
plot(pl_clean_df_2_scaled_PAM, which.plots=2, main="")
plotPAMcluster <- cbind(pl_clean_df_2, Cluster = pl_clean_df_2_scaled_PAM$clustering)
plot(plotPAMcluster)
```

Legend of variables

- pl_name: Planet Name
- sy_snum: Number of Stars
- sy_pnum: Number of Planets
- discoverymethod: Discovery Method
- disc_year: Discovery Year
- disc_facility: Discovery Facility
- pl_controv_flag: Controversial Flag
- pl_orbper: Orbital Period [days]
- pl_orbsmax: Orbit Semi-Major Axis [au])
- pl_rade: Planet Radius [Earth Radius]
- pl_bmasse: Planet Mass or Mass*sin(i) [Earth Mass]
- pl_orbeccen: Eccentricity
- pl_eqt: Equilibrium Temperature [K]
- st_spectype: Spectral Type
- sy_dist: Distance [parsec]