**Exoplanet Research Documentation**

**Samia Mahmood, Inara Rahman, and Jonathan Penaloza**

**Context:**

* Start date: May 18th, 2024
* End date: TBD
* Number of confirmed exoplanets at the time this project started: 5,630
* Raw original NASA exoplanet dataset found: <https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=PS>
* Software and resources used:
  + Python for coding and scatterplot visualization using the Jupyter Notebook extension on VS Code.
  + Excel for further data analysis and bar graph visualizations.
  + GitHub for uploading out csv files, excel files, and Python Jupyter Notebook files.

**Goal:**

Our goal was to be able to perform data analysis on NASA’s confirmed exoplanets to identify a possible Earth 2.0 which may be capable of sustaining life. We also wanted to see if we could incorporate automation to make our data analysis quicker and more efficient.

**Process and Methodology for Data Analysis:**

* First, we looked at the Original Raw exoplanet dataset (CSV file) to gain a better understanding of the dataset. This dataset had numerous missing values and there were multiple rows which contained duplicates of the same exoplanet. This is because each row was sourced from different authors/researchers.
* From there we created a dataset called Updated ExoplanetData (CSV file) where we manually deleted columns that were unnecessary and renamed each column to one that was easier to understand.
* We then created a Python Jupyter Notebook file called Updated Exoplanet Analysis. This code looked into the dataset and utilized data cleaning and filtering.
  + The missing values were left as missing.
  + Under the solution type column, any exoplanets which were TESS Project Candidates and not confirmed were filtered out.
  + Unneeded columns were filtered out.
  + (Samia’s Code) A program was created to remove duplicates of exoplanets where the mean was computed for numerical columns and the mode was computed for non-numerical columns. This allowed for the rows of duplicates of a exoplanet to become one single row.
  + After filtering, we found that there were 5,630 rows excluding the heading which matched the number of confirmed exoplanets which were also 5,630.
  + This was saved as a CSV file called filtered\_exoplanet\_dataset.
* We then created an excel version of filtered\_exoplanet\_dataset called EXCEL filtered\_exoplanet\_dataset. In this dataset, we highlighted each relevant column a color which will be utilized within our visualizations.
  + There are various sheets included in this excel file, the Detection Method excel sheet includes pie charts for the various detection methods for all 5,630 exoplanets
  + The sheets called Dist Between Planet &Star 1, Dist Between Planet &Star 2, and Dist Between Planet &Star 3 include bar graphs for the distance between an exoplanet and its host star in AU grouped by various intervals for only the exoplanets which orbit Main Sequence Stars (V), Sub-Giants (IV), and Red-Giants (III).
  + The sheet called Host Star Classification 1 looks at the columns’ planet name, host star name, spectral type, and stellar effective temp. This was where we realized that certain exoplanets were orbiting cosmological objects other than main sequence stars, sub giants, and red giants. Thus, we decided that exoplanets which are not orbiting Main Sequence Stars (V), Sub-Giants (IV), or Red-Giants (III) should not be included within our data analysis. This would allow us to focus on the most relevant exoplanets for our analysis.
  + The sheet called main sequence resources includes some references to the different types of host stars.
  + The sheet called distribution of exoplanets includes bar graphs which compare the number of Kepler exoplanets in the dataset with the number of non-Kepler exoplanets within the dataset.
  + The sheets called Filtered Dataset Radius All, Filtered Dataset Radius Kepler, Filtered Dataset Radius Non Kep were all used for the sheet titled Radius Characterizations which included bar graphs and pie charts of Exoplanets orbiting a Main Sequence Stars (V), Sub-Giants (IV), and Red-Giants (III) based on its size categories.
  + The sheet called calculations were simply to look into the different parameters of Kepler 452 b after the Updated ExoplanetDataset was ran though Samia’s Program.
* Next, we considered the idea that the Updated ExoplanetDataset may have included some outliers within the rows of duplicates of planets for certain parameters. So, we decided to create a Python Jupyter Notebook file called exoplanetCleaning which can identify whether there are outliers present for certain parameters (columns) for a given exoplanet.
  + First the code was created using the IQR method where outlier columns corresponding with the columns of the dataset were added. These outlier columns consisted of TRUE or FALSE within its entries which corresponds with the entries of the dataset. So, for a given entry, it will either be TRUE which indicates that entry is an outlier or FALSE which indicates the entry is not an outlier. However, when looking at the CSV output file called cleaned\_exoplanets along with its associated excel file titled EXCEL cleaned\_exoplanets we found that there were too many TRUEs so we decided to use another method.
  + Next for the code we decided to use the multiplier method. However, when looking at the CSV output file called cleaned\_exoplanets\_multiplier along with its associated excel file titled EXCEL cleaned\_exoplanets\_multipler we found that though there were less TRUEs than previously, there were still too many. Thus, we used another method.
  + (Inara’s code) For our code we used the Z-Score method where outlier columns corresponding with the columns of the dataset were added. These outlier columns consisted of TRUE or FALSE within its entries which corresponds with the entries of the dataset. So, for a given entry, it will either be TRUE which indicates that entry is an outlier or FALSE which indicates the entry is not an outlier. When looking at the CSV output file called cleaned\_exoplanets\_z\_scores and its associated excel file called EXCEL cleaned\_exoplanets\_z\_scores we found that this provided us with the most accurate results in which only 25 entries were flagged as TRUE or as outliers.
  + Within the excel file called EXCEL cleaned\_exoplanets\_z\_scores, there are various sheets included.
    - For the sheet titled EXCEL Exoplanets Data - Z-score, we highlighted the duplicate rows for the associated exoplanet where one of its parameters contained an entry which was flagged as TRUE for organization purposes.
    - For the sheet titled calculations, for each exoplanet where one of its parameters contained an entry which was flagged as TRUE, we calculated the mean (average) of said entries within the column and compared that with the mean (average) found from Samia’s program (which computed the mean with the outlier included). From there we manually compared these two means to see if there is a greater or smaller difference. Depending on this difference as well as the parameter in which this outlier was present in, we decided whether to keep said outlier or to remove it. We ended up keeping 18 outliers and removing 7 outliers from the dataset.
    - The sheet called Data with Removed Outliers is where we manually removed those 7 entries that were flagged as TRYE or outliers and kept the reaming 18 outliers.
* Then we decided to use Inara’s code + Samia’s code to create the CSV file called filtered\_dataset\_removed\_outliers along with its associated excel file called EXCEL filtered\_dataset\_removed\_outliers. This dataset provided us with the most accurate entries. In this dataset, we highlighted each relevant column a color which will be utilized within our visualizations.
  + There are various sheets included in this excel file, the Detection Method excel sheet includes pie charts for the various detection methods for all 5,630 exoplanets.
  + The sheet titled Orbital Period in Earth Days includes the orbital period in earth days for all exoplanets orbiting a Main Sequence Stars (V), Sub-Giants (IV), and Red-Giants (III).
  + The sheet titled Mass in Earth Mass includes bar charts of the Mass of exoplanets in Earth mass for all exoplanets orbiting a Main Sequence Stars (V), Sub-Giants (IV), and Red-Giants (III). This includes different bar graphs for different types of groupings.
  + The sheet titled Mass and Radius Relation considers a possible relation between the mass and radius of an exoplanet in terms of earth mass and earth radius.
    - The first table lists the planet classifcations which were previously used in our radius characterization sheets. However, we included mini-Neptune, sub-Neptune, and super-Neptune. Additionally, we noted our own radius range for each category which came from our radius characterization sheets. We also noted our mass range for each category, which we already knew. Next, we noted the radius range based on different sources for each category. Then, we noted the mass range for different categories based on different categories. However, we decided that given the mass and radius range for mini-Neptune and sub-Neptune, we can omit those two from our table.
    - For our second table, we listed the following planet classifications Earth like, Super-Earth like, Neptune like, Super Neptune like, Jupiter like, Super-Jupiter like. Based on our own radius range and the radius range from various sources, we created out radius conditions. Based on our own mass range and the mass range from various sources, we created out mass conditions. Using our mass\_radius\_relation code we found the count of exoplanets that satisfied the conditions for each planet classification. From there we created a bar graph.
    - Our third table is looking at solely the mass of the planet classifications listed in the second table. Using our mass\_radius\_relation code we found the count of exoplanets that satisfied the mass condition and made its associated bar graph.
    - With our fourth table, we attempted to look solely at the radius of the planet classifications listed in the second table. However, we realized this would not be able to be graphed accurately as there is an overlap in the radius conditions for super-Neptune meaning some exoplanets would be counted more than once.
    - For our fifth table, we decided to change the radius condition for the Neptune-like classification and removed super-Neptune which resulted in the following planet classifications Earth like, Super-Earth like, Neptune like, Jupiter like, Super-Jupiter like. Using our mass\_radius\_relation code we looked at the count of exoplanets that satisfy the conditions along with its associated bar graphs.
    - Our sixth table looked at solely the mass of the planet classifications based on the fifth table. Using our mass\_radius\_relation code we found the count of exoplanets and created bar graphs. However, we noticed that there were a lot of values between Neptune-like and Jupiter-like which were still unaccounted for. So, we decided to see if we can create new planet classifications.
    - Our seventh table, we decided to add the planet classifications called Saturn like and super-Saturn like which resulted in the following planet classifications Earth like, Super-Earth like, Neptune like, Saturn like, Super-Saturn like, Jupiter like, Super-Jupiter like. Using our mass\_radius\_relation code we found the count of exoplanets and created bar graphs. However, we found there were still a large number of planets unaccounted for though this time it was less than before.
    - For our eighth table, we added another classification for planets that do not fit the other conditions. Using our mass\_radius\_relation code we found the count of exoplanets and created bar graphs.
    - For our ninth table, we added the super-Neptune like classification back into the table with a different condition than previously. Thus, the table listed the following planet classifications Earth like, Super-Earth like, Neptune like, Super-Neptune like, Saturn like, Super-Saturn like, Jupiter like, Super-Jupiter like, Others. Using our mass\_radius\_relation2 code we found the count of exoplanets and created bar graphs.
    - We created a python jupyter notebook file titled mass\_radius\_relation. This code was aimed at using our filtered\_dataset\_removed\_outliers dataset (inara’s code + samia’s code) to see the relationship between an exoplanet’s mass (in earth mass) and an exoplanet’s radius (in earth radius).
      * First the code created the various conditions for the planet classifications Earth like, Super-Earth like, Neptune like, Super Neptune like, Jupiter like, Super-Jupiter like (based on the second table from Mass and Radius Relation sheet).
      * Then we found the count of exoplanets that satisfies each condition for a particular planet classification.
      * To display the results, we created a CSV output file mass\_radius\_relation. This allowed us to see all the count and names of exoplanets that satisfied each condition for a particular planet classification.
      * Then our code looked at solely the exoplanet’s mass condition for the planet classifications Earth like, Super-Earth like, Neptune like, Super Neptune like, Jupiter like, Super-Jupiter like (based on the third table from Mass and Radius Relation sheet) to find the count of exoplanets.
      * Then our code looked at solely the exoplanet’s radius condition for the planet classifications Earth like, Super-Earth like, Neptune like, Super Neptune like, Jupiter like, Super-Jupiter like (based on the fourth table from Mass and Radius Relation sheet) to find the count of exoplanets.
      * Then our code looked at the mass and radius relation for the planet classifications if we removed the super-Neptune like category and altered the radius conditions for Neptune like. This results in the following classification, Earth like, Super-Earth like, Neptune like, Jupiter like, Super-Jupiter like (based on the fifth table and sixth table from Mass and Radius Relation sheet). Through this we could find the count of exoplanets that satisfied these new conditions for Neptune like.
      * Then we looked at the number of exoplanets between Neptune like and Jupiter like that will be missing if we remove the category super-Neptune like and change the condition for Neptune like. We viewed the results of this with the CSV output file called mass\_radius\_relation\_missing\_planets and its associated excel file called EXCEL mass\_radius\_relation\_missing\_planets. This excel file lists all the exoplanets between neptune like and jupiter like that were unaccounted for along with he associated mass and radius of said exoplanets. All the mass entries were populated but some of the radius entries were not populated. So, for the exoplanets that had both mass and radius entries, we checked the radius range and mass range.
      * Then our code looked at the mass and radius relation for the planet classifications if we added Saturn like and Super-Saturn like categories. This results in the following classification, Earth like, Super-Earth like, Neptune like, Saturn like, Super-Saturn like, Jupiter like, Super-Jupiter like (based on the seventh table from Mass and Radius Relation sheet). From there we looked at the number of exoplanets between Neptune like and Jupiter like that are still unaccounted for. So, we looked at the CSV output file mass\_radius\_relation\_missing\_planets2 and its associated excel file called EXCEL mass\_radius\_relation\_missing\_planets2. From this excel file we concluded how many exoplanets from the number of unaccounted exoplanets between Neptune like and Jupiter like were taken care of after adding Saturn like and super-Saturn like.
      * Lastly our code added an Others category for planets that did not fit into the other conditions. This results in the following classification, Earth like, Super-Earth like, Neptune like, Saturn like, Super-Saturn like, Jupiter like, Super-Jupiter like, Others (based on the eighth table from Mass and Radius Relation sheet).
    - We created a python jupyter notebook file titled mass\_radius\_relation2. This code was aimed at verifying our results from our initial mass\_radius\_relation code.
      * First the code looked at the number of exoplanets in total that satisfies the various conditions (based on the ninth table from Mass and Radius Relation sheet). We displayed these results in a CSV output file called mass\_radius\_relation2. However, we realized that instead of having 4,175 exoplanets that satisfy the condition, we were getting a number that was too large of 4,613 exoplanets.
      * So, we decided to try another approach to look at the number of exoplanets in total that satisfies the various conditions to see a possible verification for the previous method. This resulted in the CSV output file called verify\_mass\_radius\_relation2 however we found the result to be the same with 4,613 exoplanets instead of 4,175 exoplanets. This implied that there may be exoplanets within the code that were counted more than once.
      * Lastly, to see if the exoplanets were in fact counted more than once, we created another CSV output file called verifying\_results. With this file we realized that there were numerous exoplanets in the Others category which were already taken care of in the other categories and therefore resulted in exoplanets being counted more than once.
  + The sheet titled Radius Characterization includes bar graphs and pie charts of Exoplanets orbiting a Main Sequence Stars (V), Sub-Giants (IV), and Red-Giants (III) based on its size categories.
  + The sheets called Dist Between Planet &Star 1, Dist Between Planet &Star 2, and Dist Between Planet &Star 3 include bar graphs for the distance between an exoplanet and its host star in AU grouped by various intervals for only the exoplanets which orbit Main Sequence Stars (V), Sub-Giants (IV), and Red-Giants (III).
  + The sheet called distribution of exoplanets includes bar graphs which compare the number of Kepler exoplanets in the dataset with the number of non-Kepler exoplanets within the dataset.
  + The sheet called main sequence resources includes some references to the different types of host stars.
  + The sheet titled Kepler Stars looked at the names of Kepler exoplanets (highlighted in a bright yellow) and their associated host star names where the host star name column was filtered from ascending order. The sheet titled Kepler Stars 2 served the same purpose expect the planet name column was filtered in ascending order. Originally, only exoplanets that start with Kepler or K2 were considered Kepler exoplanets and thus, highlighted in bright yellow while non-Kepler exoplanets were highlighted in a light yellow. However, we realized that exoplanets that start with KOI and EPIC were also in fact Kepler exoplanets.
  + The sheet titled Jupiter Radius Filled In is where we populated the Jupiter radius column where there was missing data for its associated earth radius. This was highlighted in red,
* Next, we created codes for creating scatterplots through python.
  + (Inara’s code) We created a python Jupyter notebook file called exoplanetScatterplots. This code created visualizations for all the scatterplots from the exoplanet powerpoint except for the last scatter plot.
  + (Samia’s code) We created a python Jupyter notebook file called scatterplots\_exoplanets. This code created visualizations for the last scatterplot from the exoplanet powerpoint with some additional scatterplots in reference to this.
  + (Jonathan’s code) We created a python Jupyter notebook file called Exoplanets\_Visuals. This code created visualizations for all the scatterplots from the exoplanet powerpoint and it serves as a way of confirming Inara’s code and Samia’s code as it was created using a different method from the previous two codes.
* Then, we created a python Jupyter notebook file called best\_exoplanet\_match which aimed at identifying the exoplanet which is the best match to Earth based of various parameters.
* Next, we will be populating the Stellar Type column from filtered\_dataset\_removed\_outliers using scholarly sources. This will allow us to have a better idea of exactly how many exoplanets are not orbiting a main sequence star, sub giant, or red giant. Additionally, it will make it easier to perform data analysis as we can then do additional filtering to find the best match and make visualizations.
* Then, we will be making a Host Star Clssification code to classify the exoplanet’s based on their host star type once we have finished populating the Stellar Type column.