eda

March 3, 2023

1 Dataset

Exoplanet dataset is retrieved from http://exoplanetarchive.ipac.caltech.edu on Sun Feb 26 04:31:47 2023 West Standard Time. They are: - Confirmed exoplanets: 5272 rows, 92 columns - Planet emission spectrum: 574 rows, 14 columns - Planet transit spectrum: 5745 rows, 22 columns

Depending on models used in later analysis, some columns are subject to be dropped or cleaned. But we keep all columns for now in case for new models. The following analysis is based on the confirmed exoplanets dataset.

2 Variables

Variable	Description	Unit
pl_name	Planet Name	
hostname	Host Name	
default_flag	Default Parameter Set	
sy_snum	Number of Stars	
sy_pnum	Number of Planets	
discoverymethod	Discovery Method	
disc_year	Discovery Year	
disc_facility	Discovery Facility	
soltype	Solution Type	
pl_controv_flag	Controversial Flag	
pl_refname	Planetary Parameter Reference	
pl_orbper	Orbital Period	days
pl_orbpererr1	Orbital Period Upper Unc.	days
pl_orbpererr2	Orbital Period Lower Unc.	days
pl_orbperlim	Orbital Period Limit Flag	
pl_orbsmax	Orbit Semi-Major Axis	au
pl_orbsmaxerr1	Orbit Semi-Major Axis Upper Unc.	au
pl_orbsmaxerr2	Orbit Semi-Major Axis Lower Unc.	au
pl_orbsmaxlim	Orbit Semi-Major Axis Limit Flag	
pl_rade	Planet Radius	Earth Radius
pl_radeerr1	Planet Radius Upper Unc.	Earth Radius
pl_radeerr2	Planet Radius Lower Unc.	Earth Radius
pl_radelim	Planet Radius Limit Flag	
pl_radj	Planet Radius	Jupiter Radius

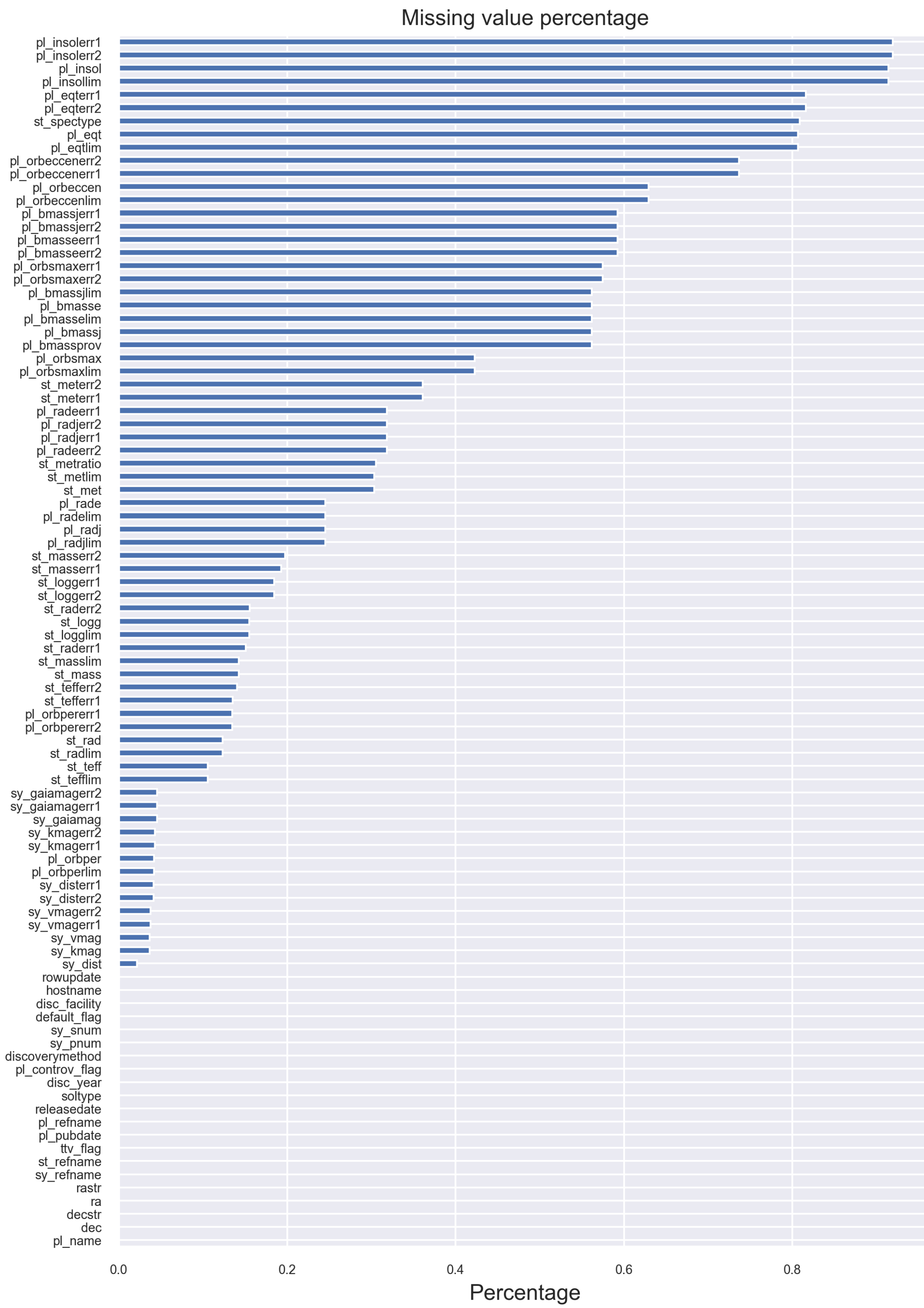
Variable	Description	Unit
pl_radjerr1	Planet Radius Upper Unc.	Jupiter Radius
pl_radjerr2	Planet Radius Lower Unc.	Jupiter Radius
pl_radjlim	Planet Radius Limit Flag	
pl_bmasse	Planet Mass or Mass*sin(i)	Earth Mass
pl_bmasseerr1	Planet Mass or Mass*sin(i) Upper Unc.	Earth Mass
pl_bmasseerr2	Planet Mass or Mass*sin(i) Lower Unc.	Earth Mass
pl_bmasselim	Planet Mass or Mass*sin(i) Limit Flag	Earth Mass
pl_bmassj	Planet Mass or Mass*sin(i)	Jupiter Mass
pl_bmassjerr1	Planet Mass or Mass*sin(i) Upper Unc.	Jupiter Mass
pl_bmassjerr2	Planet Mass or Mass*sin(i) Lower Unc.	Jupiter Mass
pl_bmassjlim	Planet Mass or Mass*sin(i) Limit Flag	Jupiter Mass
pl_bmassprov	Planet Mass or Mass*sin(i) Provenance	
pl_orbeccen	Eccentricity	
pl_orbeccenerr1	Eccentricity Upper Unc.	
pl_orbeccenerr2	Eccentricity Lower Unc.	
pl_orbeccenlim	Eccentricity Limit Flag	
pl_insol	Insolation Flux	Earth Flux
pl_insolerr1	Insolation Flux Upper Unc.	Earth Flux
pl_insolerr2	Insolation Flux Lower Unc.	Earth Flux
pl_insollim	Insolation Flux Limit Flag	
pl_eqt	Equilibrium Temperature	K
pl_eqterr1	Equilibrium Temperature Upper Unc.	K
pl_eqterr2	Equilibrium Temperature Lower Unc.	K
pl_eqtlim	Equilibrium Temperature Limit Flag	

2.1 Missing values analysis

[ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[ ]:
```
# import data
df = pd.read_csv('data/confirmed.csv', header=98)
```

[ ]:
```
fig, ax = plt.subplots(figsize=(8,12), dpi=300)
(df.isna().sum()/len(df)).sort_values().plot(kind='barh', ax=ax)
ax.set_title('Missing value percentage')
ax.set_xlabel('Percentage')
ax.tick_params(axis='both', which='major', labelsize=7)
```



Missing value percentage
pl_insolerr1
pl_insolerr2
pl_insol
pl_insollim
pl_eqterr1
pl_eqterr2
st_spectype
pl_eqt
pl_eqtlim
pl_orbeccenerr2
pl_orbeccenerr1
pl_orbeccen
pl_orbeccenlim
pl_bmassjerr1
pl_bmassjerr2
pl_bmasseerr1
pl_bmasseerr2
pl_orbsmaxerr1
pl_orbsmaxerr2
pl_bmassjlim
pl_bmasse
pl_bmasselim
pl_bmassj
pl_bmassprov
pl_orbsmax
pl_orbsmaxlim
st_meterr2
st_meterr1
pl_radeerr1
pl_radjerr2
pl_radjerr1
pl_radeerr2
st_metratio
st_metlim
st_met
pl_rade
pl_radelim
pl_radj
pl_radjlim
st_masserr2
st_masserr1
st_loggerr1
st_loggerr2
st_raderr2
st_logg
st_logglim
st_raderr1
st_masslim
st_mass
st_tefferr2
st_tefferr1
pl_orbpererr1
pl_orbpererr2
st_rad
st_radlim
st_teff
st_tefflim
sy_gaiamagerr2
sy_gaiamagerr1
sy_gaiamag
sy_kmagerr2
sy_kmagerr1
pl_orbper
pl_orbperlim
sy_disterr1
sy_disterr2
sy_vmagerr2
sy_vmagerr1
sy_vmag
sy_kmag
sy_dist
rowupdate
hostname
disc_facility
default_flag
sy_snum
sy_pnum
discoverymethod
pl_controv_flag
disc_year
soltype
releasedate
pl_refname
pl_pubdate
ttv_flag
st_refname
sy_refname
rastr
ra
decstr
dec
pl_name
0.0
0.2
0.4
0.6
0.8
Percentage

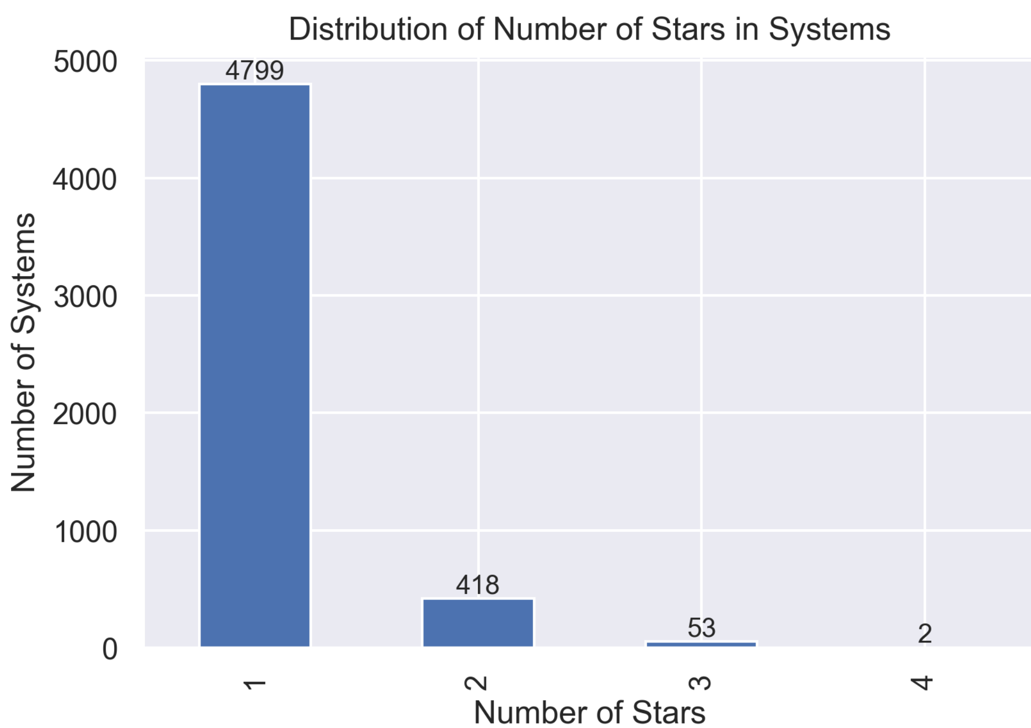
2.2 Summary on missing values

Some variables have a high percentage of missing values. We need to be careful when using these variables in later analysis.

2.3 Distribution of some interesting variables

2.3.1 Number of host stars

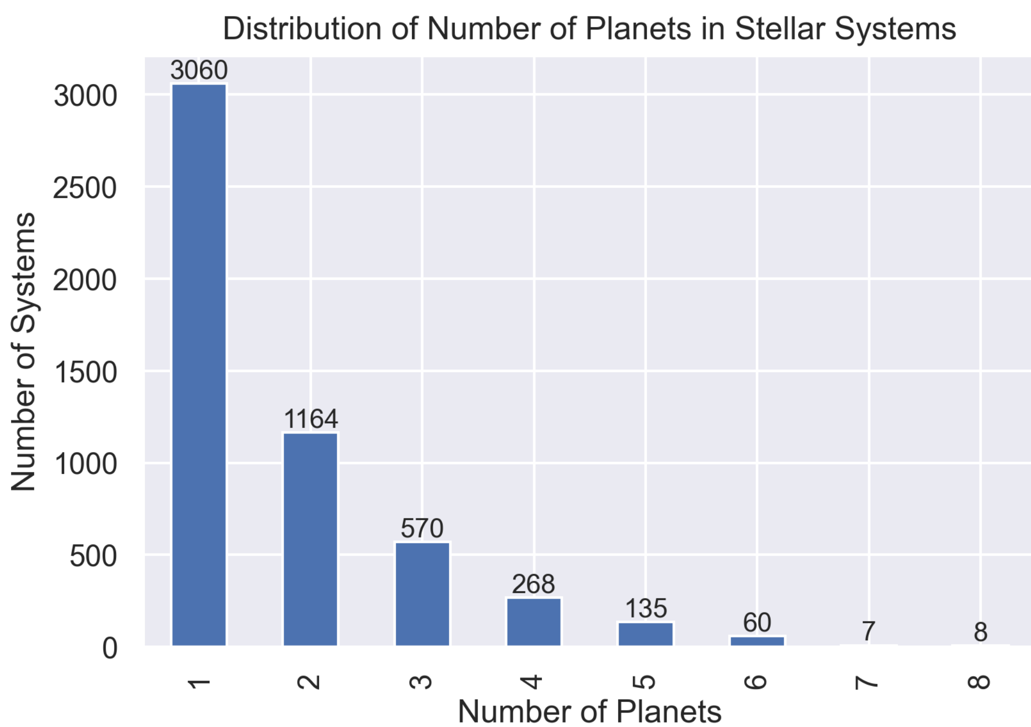
[ ]:
```
fig, ax = plt.subplots(1, 1, figsize=(6, 4), dpi=300)
# bar plots of number of stars
ax = df['sy_snum'].value_counts().plot(kind='bar', ax=ax)
ax.set_xlabel('Number of Stars')
ax.set_ylabel('Number of Systems')
ax.set_title('Distribution of Number of Stars in Systems')

for i in ax.containers:
    ax.bar_label(i, label_type='edge', fontsize=10)
```



As expected, Unary star system is most common in the exoplanet dataset, followed by binary and trinary star system. ### Number of planets

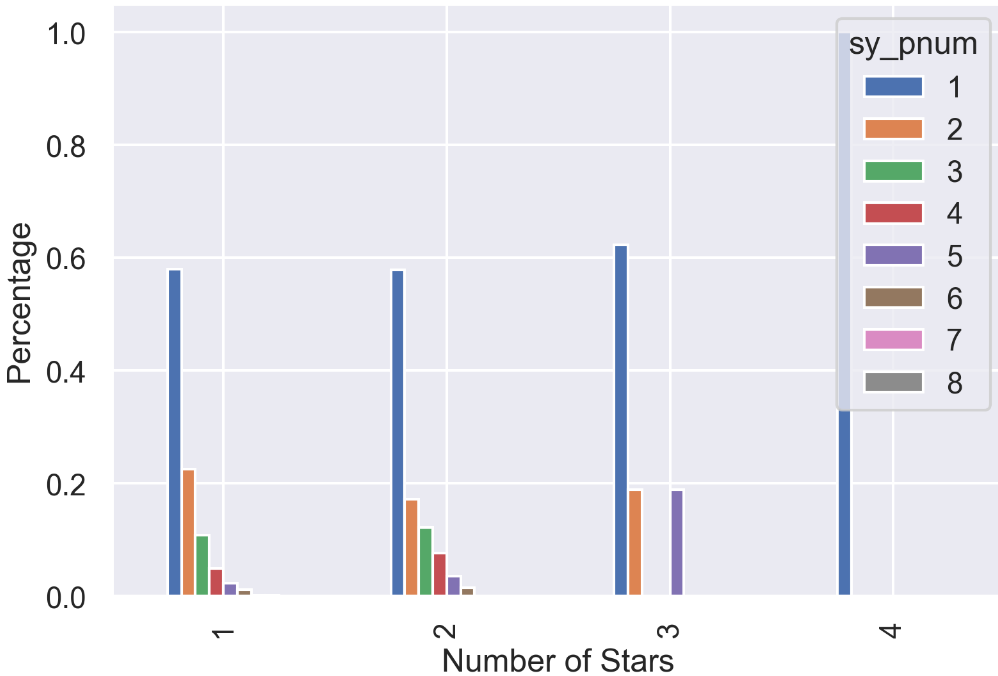
[ ]:
```
fig, ax = plt.subplots(1, 1, figsize=(6, 4), dpi=300)
# bar plots of number of planets.
ax = df['sy_pnum'].value_counts().sort_index().plot(kind='bar', ax=ax)
```

```
ax.set_xlabel('Number of Planets')
ax.set_ylabel('Number of Systems')
ax.set_title('Distribution of Number of Planets in Stellar Systems')

for i in ax.containers:
    ax.bar_label(i, label_type='edge', fontsize=10)
```



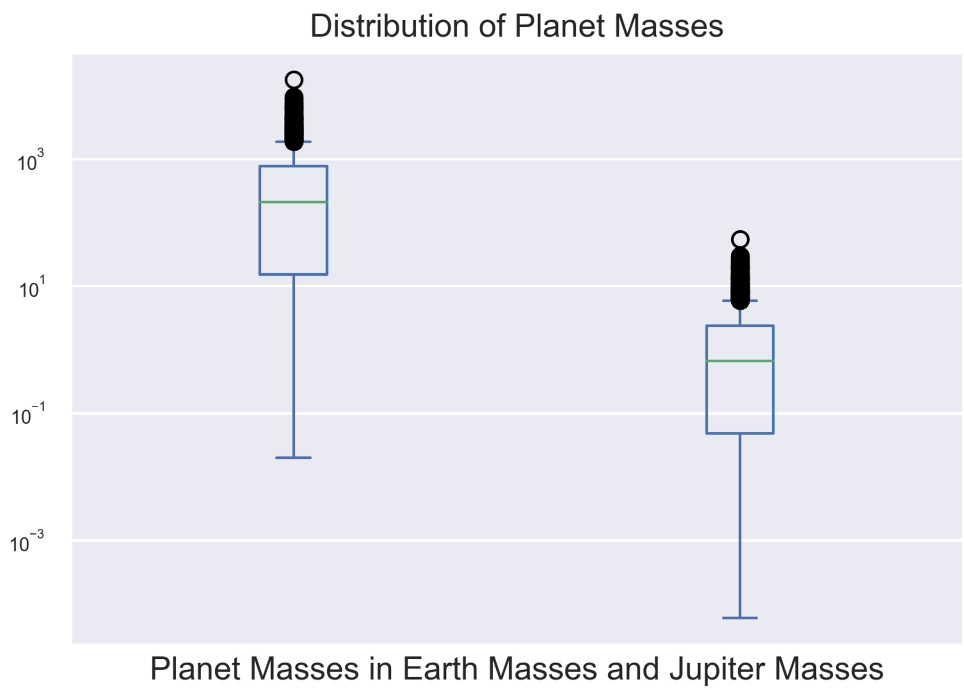
[ ]:
```
fig, ax = plt.subplots(1, 1, figsize=(6, 4), dpi=300)
# bar plots of number of planets unary and binary systems y-axis is percentage
df.groupby('sy_snum')['sy_pnum'].value_counts(normalize=True).unstack().
 ↪plot(kind='bar', ax=ax)

ax.set_xlabel('Number of Stars')
ax.set_ylabel('Percentage')
```

[ ]: Text(0, 0.5, 'Percentage')



For the unary and binary star system, number of planets seem to follow the same distribution. But it seems that higher order stellar system tend to have fewer planets. But it could also be attributed to less data for higher order stellar systems (53 trinary systems and only 2 quadruple star systems). ### Mass of planets

[ ]:
```
fig, ax = plt.subplots(1, 1, figsize=(6, 4), dpi=300)
# plot box plot of mass distribution of planets
ax = df[['pl_bmasse','pl_bmassj']].plot(kind='box', ax=ax)
ax.set_title('Distribution of Planet Masses')
ax.set_xlabel('Planet Masses in Earth Masses and Jupiter Masses')
ax.set_yscale('log')
ax.set_xticks([])
ax.tick_params(axis='both', which='major', labelsize=7)
```

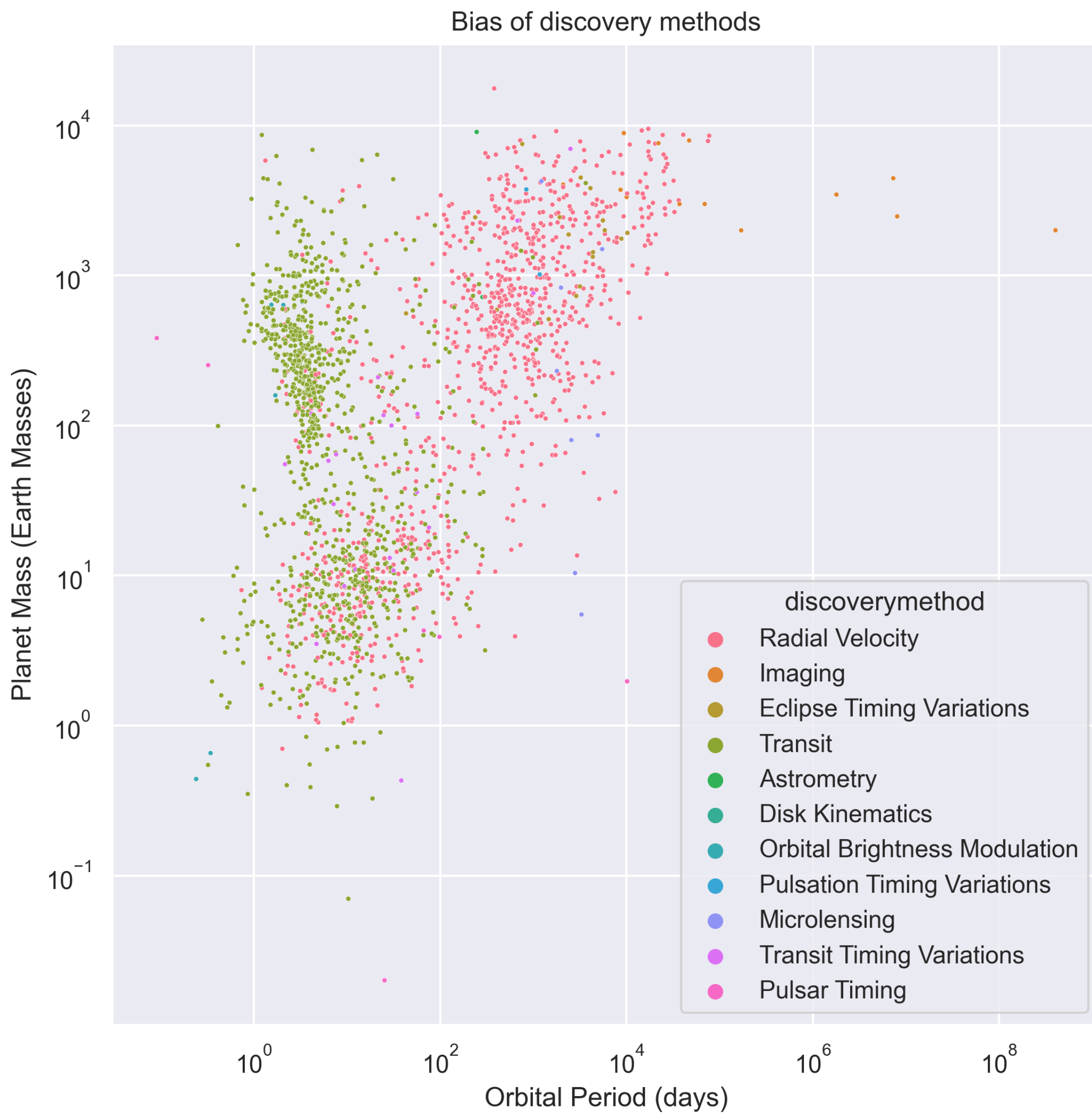


2.4 Variable dependency analysis

2.4.1 Bias of discovery methods

[ ]:
```
# scatter plot of planet mass vs planet periods with discovery method as hue
fig, ax = plt.subplots(1, 1, figsize=(8, 8), dpi=300)
sns.scatterplot(data=df, x='pl_orbper', y='pl_bmasse', hue='discoverymethod',␣
    ↪ax=ax, s=5)
ax.set_xscale('log')
ax.set_yscale('log')
ax.set_xlabel('Orbital Period (days)')
ax.set_ylabel('Planet Mass (Earth Masses)')
ax.set_title('Bias of discovery methods')
```

[ ]:
```
Text(0.5, 1.0, 'Bias of discovery methods')
```



Dominant discovery methods include transit, radial velocity and microlensing. But those methods are not equally biased towards different types of planets. From the scatterplot, we can see transit has bias - On planets with short periods and small semi-major axes because they transit their host star more frequently. - On larger planets are more likely to be detected since they block more light. - On planets whose orbital planes pass earth. Otherwise transit would not be observed.

Radial velocity has bias - On massive planets that are close to host stars. - On planets whose orbital plane close to the earth. The constraint on orbital plane is less strict than transit method.

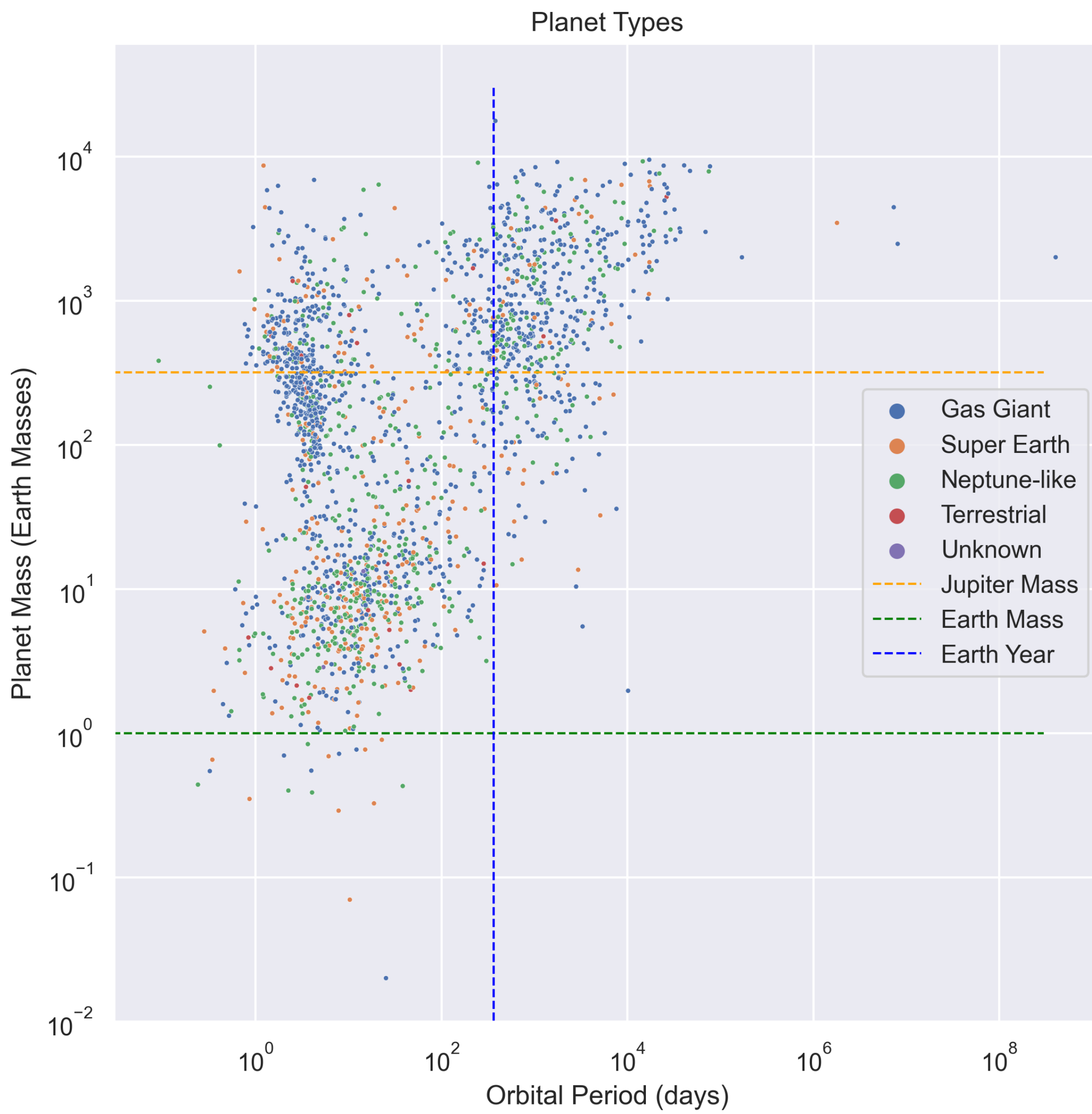
Micro-lensing has bias - On massive planets bc massive planets have stronger lensing effect - On planets distant from host stars (Complementary to transit and radial velocity methods) - On planets with specific orbital period, orientation and eccentricity because lensing effect is strongest when the planet resides in the middle between background starlight and the earth

2.4.2 Planet type

[ ]:
```
planet_type = pd.read_csv('data/cleaned_5250.csv')['planet_type']
```

[ ]:
```
fig, ax = plt.subplots(1, 1, figsize=(8, 8), dpi=300)
sns.scatterplot(data=df, x='pl_orbper', y='pl_bmasse', hue=planet_type, ax=ax,␣
 ↪s=5)
ax.plot([0, 3e8], [318, 318], color='orange', linestyle='--', linewidth=1,␣
 ↪label='Jupiter Mass')
ax.plot([0, 3e8], [1, 1], color='green', linestyle='--', linewidth=1,␣
 ↪label='Earth Mass')
ax.plot([365.25, 365.25], [0, 3e4], color='blue', linestyle='--', linewidth=1,␣
 ↪label='Earth Year')
ax.set_xscale('log')
ax.set_yscale('log')
ax.set_xlabel('Orbital Period (days)')
ax.set_ylabel('Planet Mass (Earth Masses)')
ax.set_title('Planet Types')
ax.legend()
```

[ ]:
```
<matplotlib.legend.Legend at 0x7fcc70643610>
```



There are several counter-intuitive findings from the scatterplot: - Hot Jupiters are common. - Earth-like planets are much rarer than super-earths.
- Most planets have less period than earth-year.

The above findings are contradictory to our solar system. This could be result of the bias of discovery methods or the fact that Hot Jupiters are common and Earth-like planets are indeed rare in the universe.

2.4.3 Discovery year

[ ]:
```
fig, ax = plt.subplots(1, 1, figsize=(8, 8), dpi=300)
# create the countplot
```

```
df.value_counts(['disc_year', 'discoverymethod']).unstack().plot(kind='bar',␣
↪figsize=(10, 6), stacked=True, ax=ax)

# set the title and x/y-axis labels
ax.set_title('Number of Planets Discovered by Year and Method')
ax.set_xlabel('Year')
ax.set_ylabel('Number of Planets')

# rotate x-axis tick labels to avoid overlapping
ax.tick_params(axis='x', rotation=90)

# display the plot
plt.show()
```