Jingxuan Feng s3843790

Data Preparation

In task 1, I went through the dataset column by column looking for errors. First, I did a quick manual inspection of the dataset, mainly keeping an eye out on missing values or NaN or if there are incorrect data types ie an int rather than a string. Then, I proceeded to use python to check for data type inconsistencies, missing values, impossible values and other errors as outlined in detail below.

Error 1 – Incorrect data type (Player, Pos, Tm)

The initial errors located is that pandas dataframe sees 'Player', 'Pos' and 'Tm' columns as an object data type. I have corrected this by using 'pandas.DataFrame.astype' to manually set those columns as string data. This allows for usage of string methods in python later. However, this also notified me to errors in Pos and Tm columns as spaces or type mismatch can result in the data being classified as an object.

Error 2 – Redundant white space & others (Pos)

Some data entries in the Position column contained redundant white space, extra letters / symbols on the values or values were not capitalised. To fix this, I have used '.str.lstrip()' and '.str.rstrip()' to remove redundant white space, removed extra letters and symbols from the data entries then capitalised any values that were in lower case.

Error 3 – Impossible values (Age)

When checking for minimum and maximum ages of the players, it showed a minimum of -19 then a maximum of 280. A more though search into the data showed that these are the only two impossible values for the Age column. I proceeded to replace them with 19 and 28 respectively as they appear to be typos and not false information.

Error 4 – Redundant white space & others (Team)

Same actions have been taken on the Teams column to remove redundant white space & capitalisation. But, this time, the Team name had an instance of 'H0U' rather than 'HOU' which is corrected.

Error 5 – Null values (FG%)

When using '.isnull().values.any()', a true value was returned indicating there are null values in FG%. Null values has been filled with 0.

Error 6 – Null values (3P%)

Same checks and fixing steps taken as the FG% column.

Error 7 – Null values (2P%)

Same checks and fixing steps taken as the FG% and 3P% column.

Error 8 – Null values (FT%)

Same checks and fixing steps taken as the FG%, 3P% and 2P% column.

Jingxuan Feng s3843790

Error 9 – Incorrect values (PTS)

Two players' PTS values were over 2000. I fixed this by recalculating their PTS scores then overwriting their incorrect scores with the recalculated values.

Error 10 – Duplicate Player Data

At the start, we noticed a 10 players who has played for different teams resulting in duplicate entries for them. To fix these values, I added up all the different values such as their scores / points. These are then overwritten onto their last entry, so that their team and position values would base off their current team. All other replicate entries are then deleted.

Unmentioned values

Any columns I did not mention here means that I did not find any problems with them, but validation checks are in the Jupyter notebook.

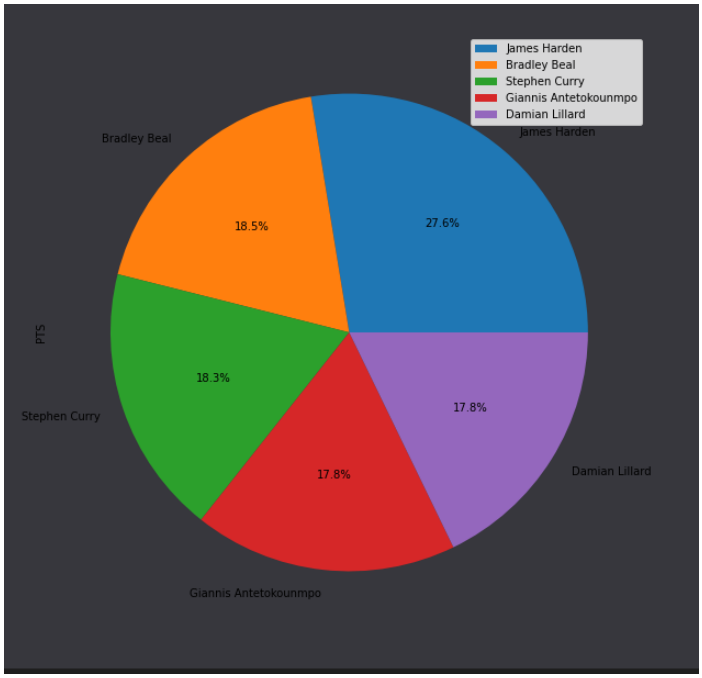
Data checking section

This section performs a few basic checks to ensure no errors remain in the dataset. These tests include checking for whether the lower and upper bounds of the data meet the requirements or expected values. Such as no ages below 18, number of teams and 'PTS' limit ect. Checks on data type for each column is also performed to ensure the actual data type matches what we expect the datatype to be. Finally, another check for NaN values are performed on each column to ensure nothing was missed during data cleaning.

Data Exploration

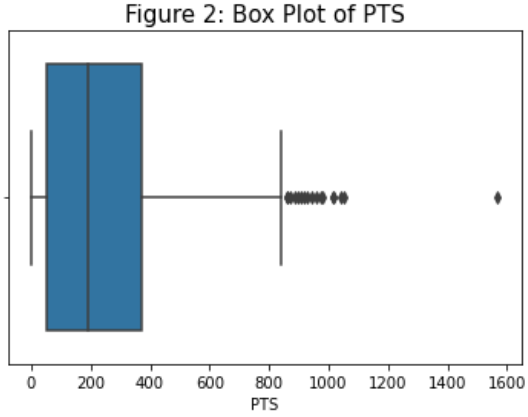
Task 2.1

From the pie chart, we can see that the player James Harden had scored 9.1% more than the next high scorer when comparing the top 5 player's total points. The other 4 players scored similarly to each other ranging from 17.8 – 18.5%.



Furthermore, the boxplot also shows that the top 5 player's PTS is way above average when compared to the average scores of all other player.

Figure 2: Box Plot of PTS

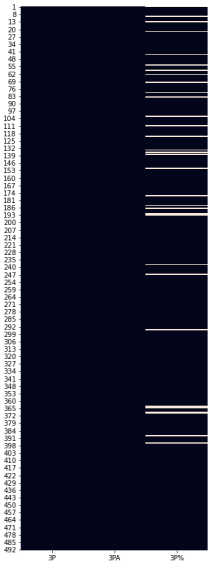


Task 2.2

In this task, I first re-imported the original dataset containing errors, then use '.head()' to check that this had worked as expected. Then using '.info()', we can see quite a few entries are missing since we expect to see 512 non-null whereas it shows there are only 479 non-null. Using a temporary dataframe, I located the players with missing values in their 3P% column and listed out their names.

Rk	Player	3P%
1	Precious Achiuwa	NaN
9	Jarrett Allen	NaN
13	Kostas Antetokounmpo	NaN
22	Udoka Azubuike	NaN
44	Tyler Bey	NaN
54	Marques Bolden	NaN
59	Brian Bowen	NaN
63	Ignas Brazdeikis	NaN
70	Moses Brown	NaN
80	Devontae Cacok	NaN
84	Clint Capela	NaN
102	Tyler Cook	NaN
111	Ed Davis	NaN
121	Devon Dotson	NaN
134	Drew Eubanks	NaN

136	Tacko Fall	NaN
138	Cristiano Felício	NaN
151	Daniel Gafford	NaN
177	Ashton Hagans	NaN
184	Jared Harper	NaN
187	Jalen Harris	NaN
192	Isaiah Hartenstein	NaN
193	Jaxson Hayes	NaN
238	Damian Jones	NaN
247	Enes Kanter	NaN
295	Patrick McCaw	NaN
365	Norvel Pelle	NaN
365	Norvel Pelle	NaN
365	Norvel Pelle	NaN
369	Jakob Poeltl	NaN
370	Vincent Poirier	NaN
389	Nick Richards	NaN
396	Mitchell Robinson	NaN



I also used a heatmap to visualise the rough number and location of these errors using a heatmap. (above right). There was a total of 33 errors in the entries if we count repeated players.

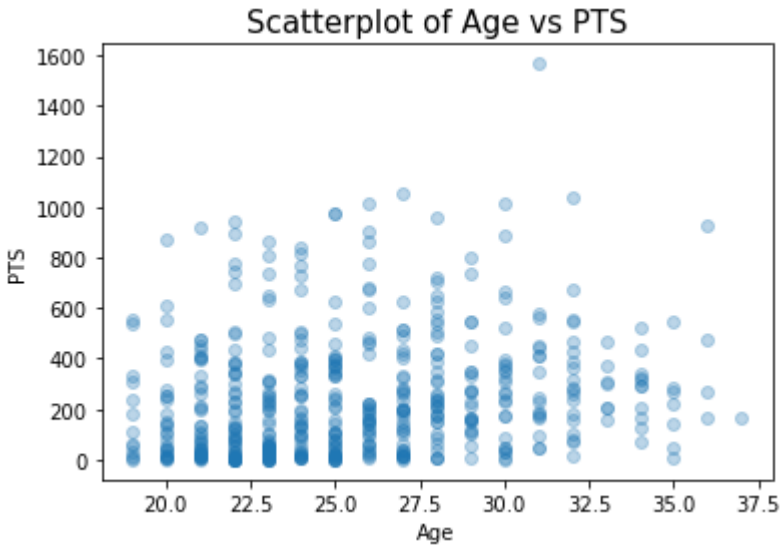
Task 2.3

In this section, I will be comparing the player's Age, offensive rebounds, assists, and steals against their total points. I choose to use scatterplots since they are good at comparing whether two variables are correlated to each other.

Age

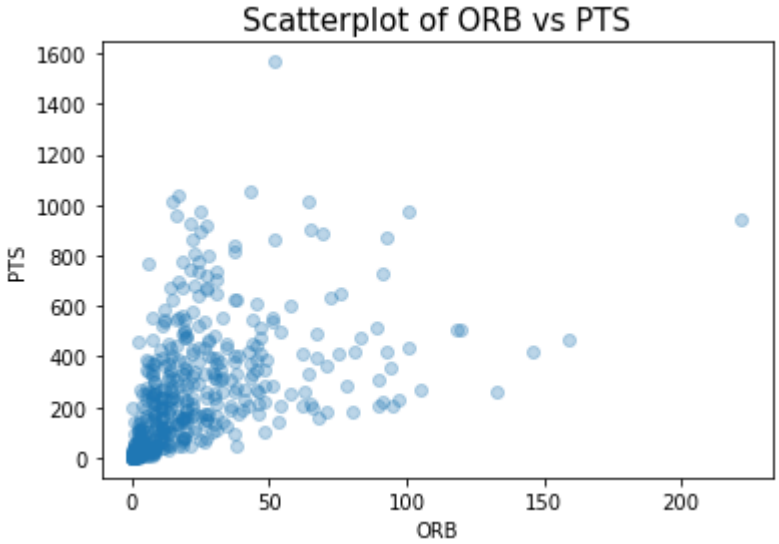
In the PTS vs Age scatterplot, we can see that there is no correlation between Age and PTS. Regardless of the player's age group, there is a wide spread of PTS. Hence, showing that age does not affect the player's ability to score.

Jingxuan Feng s3843790



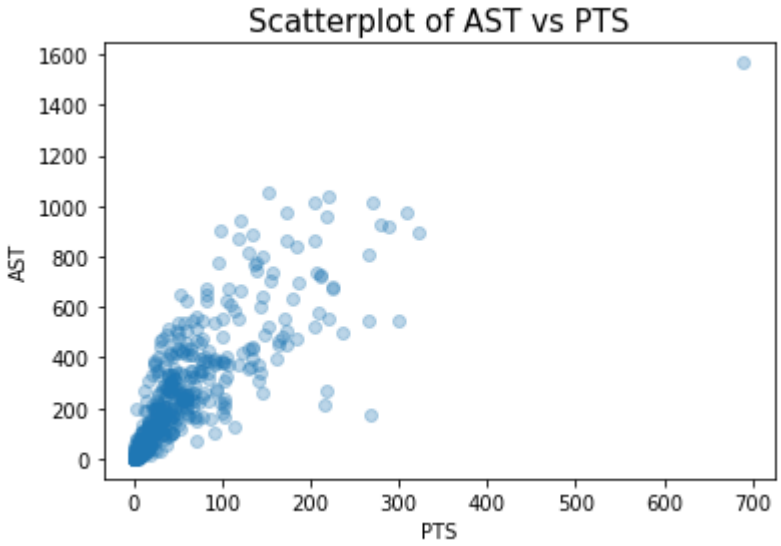
ORB

This scatterplot shows that there is a correlation between the two variables. It suggests that a player with lower ORB will in have a higher PTS. This is likely due to the higher ORB means the more shots a player got to take at the goal but had missed. A low ORB could mean a player was accurately scoring for every throw and did not miss and result in a rebound.



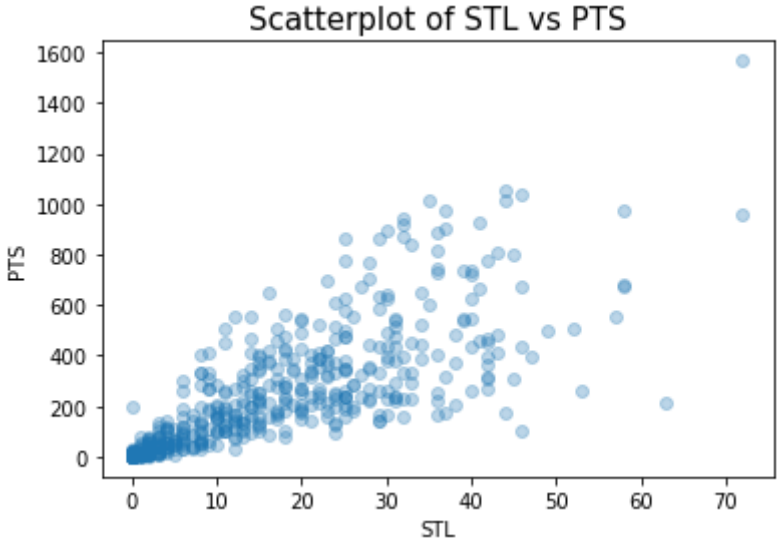
AST

The assists vs PTS shows a correlation between number of assists and a player's PTS. High number of assists sometimes yield a high PTS; this could be due to the assisting player not getting many chances to score themselves. Whilst a low PTS could mean players were not in position to assist or not tasked in a more defensive role. Players with high assets and PST could be passing the ball to a teammate then receiving the ball back to go for the goal.



STL

There is a very weak correlation between number of steals and a player's PTS. Some players with a high number of steals have a high PTS but the opposite can be said as well. However, it is true that players with a low number of steals also have a low PTS. These are likely players tasked with defence whilst others are either helping the offensive player or have an offensive role themselves.



References

- [1]"seaborn heatmap - Python Tutorial", Pythonbasics.org, 2021. [Online]. Available: https://pythonbasics.org/seaborn-heatmap/. [Accessed: 09- Apr- 2021].
- [2]"How to change a Seaborn plot size?", Dataforeverybody.com, 2021. [Online]. Available: https://www.dataforeverybody.com/seaborn-plot-figure-size/. [Accessed: 17- Apr- 2021].