Student Name: Sai Ramishetty Student ID: s3841545

Data Preparation

During Data Preparation, I encountered errors like typos, impossible values, redundant white spaces, letter-case problems and missing values. Outliers were identified too, but as the values were valid and no strong evidence was there to remove them, they were kept.

Error 1- Typos

I used the function "value_counts()" to see the position and team names. There were plenty of typos, such as names with wrong spelling(for example SGa and SF. , which is supposed to be SG and SF). To remove these errors, I identified the correct name for these typos and replaced them using the function "replace()". To justify this replacement, I checked how many values each name had. For example, HOU had 18 values in the data, which is a high number and therefore, it is reasonable to assume H0U(the middle part of the name is zero, not O!) is the same as HOU.

Error 2- Impossible Values

I used if and else statements to identify any impossible value. For example, an NBA player has to be minimum age of 19 years old while the possible highest age is 120 years. After executing the program, I found two errors. Similarly, I found another error in the Points age column, which is a player's points exceeding 2000 points. For fixing the error in Age, I checked the values count to see whether replacing the errors with the values in the data was reasonable. For example, one impossible error in Age was -19, but since there was a significant number of players with 19 years, it was justified to replace it. For points, there is a formula to calculate the total points(which is 3*number of 3P goals + 2*number of 2P goals + number of FT(Free throws)). This was used to calculate the real points and these values were used to replace the impossible errors. In this column, the error 20000 was replaced by 2(0*3 + 1*2 + 0*1 = 2).

Error 3 – Redundant white spaces

Along with typos, these errors were identified too. All redundant white spaces were removed with the function "str.strip()". For example, there were 3 errors which had " PG", but after the strip, it ended up as "PG" and they got added to "PG" .

Error 4 – Letter-case problem

This too was identified along with typos and redundant white spaces. Similarly, to typos, the value of each name in the column was checked to see if they can replace the errors. For example, to replace the error "pg", I checked the number of positions with "PG". Since there was a significant number of positions with PG, I replaced "pg" with "PG" using the "replace()" function.

Error 5 – Missing values

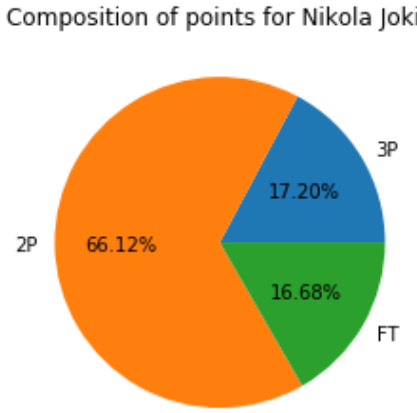
Interestingly, the columns which had null values or missing values was the percentage columns. To detect whether there were null values, I used an if statement inside the for loop and "isnan()" function to check which value was empty. The way to fix this was checking the values of the field goals and field goal attempts. For example, I checked the values of 3P and 3PA for the null values in 3P%. I came to know that both 3P and 3PA had zeroes. This was reflected in other field goal and field goal attempts columns as well. The formula to calculate 3P% is 3P/3PA, so, 0/0= 0. Therefore, I replaced all null values with zero using the function "fillna()".

Data Exploration

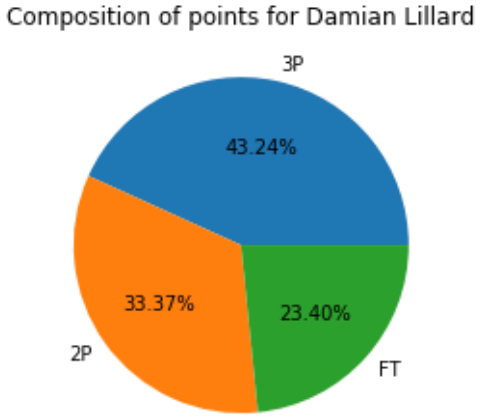
Task 2.1

This task required us to find the composition of the points of top five players. The best way to tackle this problem was using a pie chart. A pie chart can clearly explain the composition of each individual's points. Here are the pie charts for each player below.

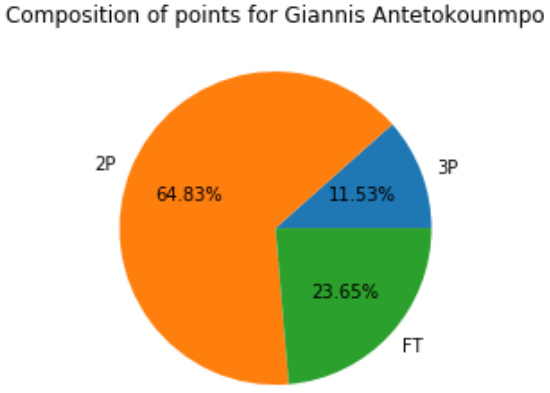
Top 5 player- Nikola Jokić



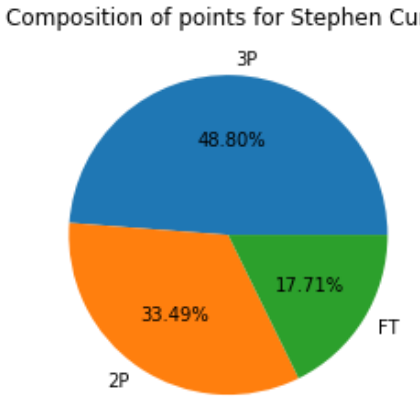
Top 4 player - Damian Lillard



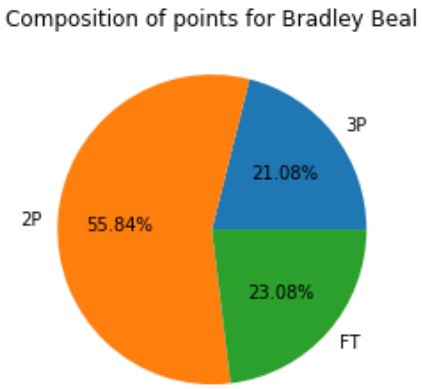
Top 3 player – Giannis Antetokounmpo



Top 2 player – Stephen Curry



Top 1 player – Bradley Beal

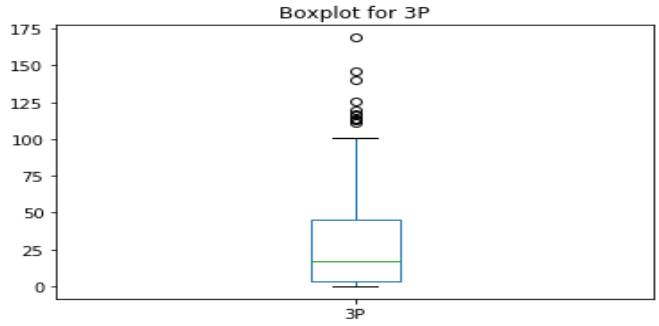


The percentage of 2P goal points was generally the highest. Even if it was lower than the percentage of 3P points, the margin is not the same when percentage of 2P points is higher. For example, the percentage margin between 2P points and 3P points in player one's points composition is 34.76% while it is 15.31% in player two's points composition. Even in top 4 player's composition, the percentage margin between 3P and 2P is 9.87% . Interestingly, the % of 3P goals is less in 3 out of 5 player's points, but this can be explained since many players do not score more 3P goals than 2P goals. The percentage of Free throws was never the highest in any pie chart, despite having higher number of throws. In conclusion, if a player wants to score more points, they should at least aim for a decent number of 2P goals.

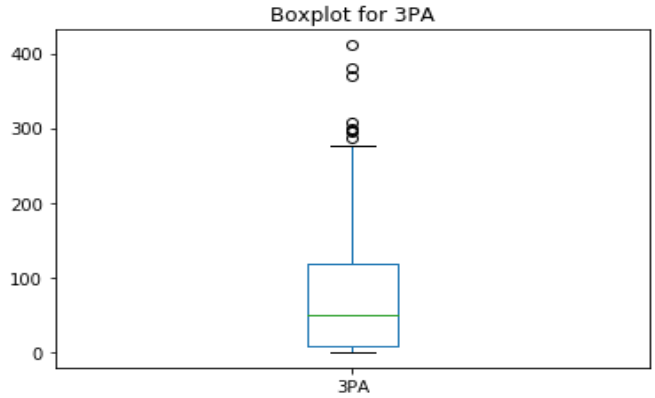
Task 2.2

For each column, a box plot was drawn to identify potential outliers that could be the errors. Below are the graphs.

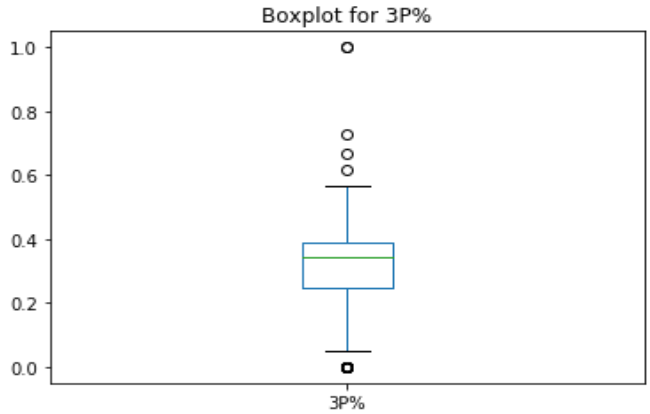
Boxplot for 3P column



Boxplot for 3PA column



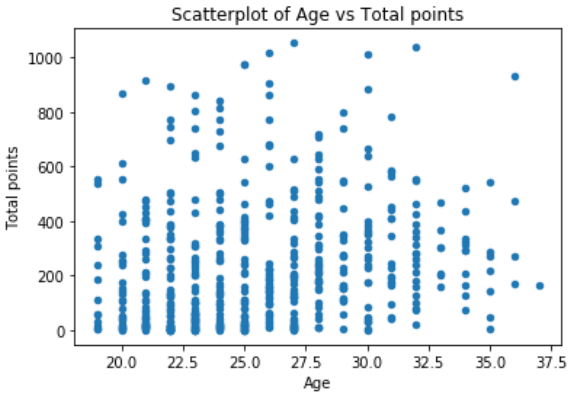
Boxplot for 3P% column



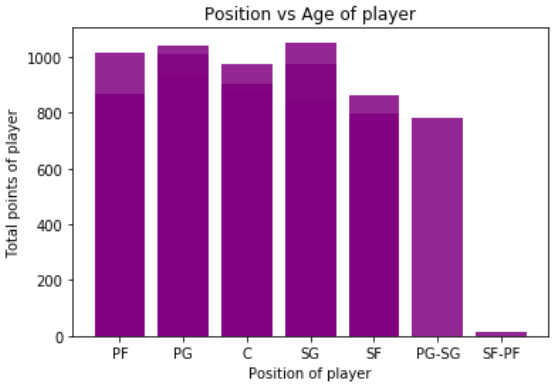
Each box plot was properly analysed for potential errors. However, all the outliers present in each box plot are not invalid. A justification for this is a player can definitely either overperform or underperform in comparison to the rest of the players. For example, a player has made 411 3P goal attempts. However, this is within 35 games, which is definitely reasonable. If it was between 1 and 5, then it would be an invalid value. Another thing to note is a player can have 0.0 3P% since they may not score any 3P goal despite making attempts. As this trend continues for all outliers, there is no strong justification to declare them as errors. Hence, it must mean that the errors were removed in the data preparation stage.

Task 2.3

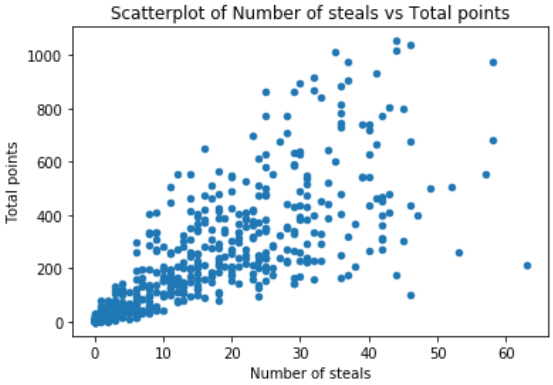
The features chosen for this task is Age, position of player, number of personal fouls, number of steals and number of field goal attempts. Below are series of graphs with explanations



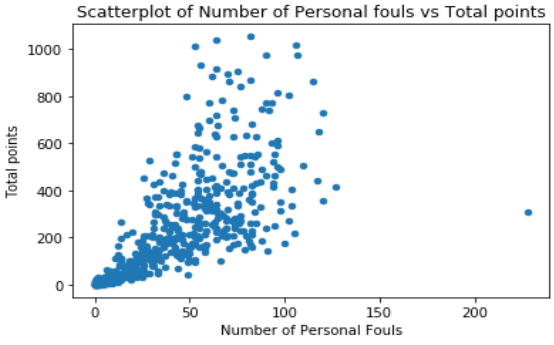
As it can be seen in this scatterplot, there is no correlation between Age and total points of a player. All data points are all rather vertically scattered, therefore, there is no linear relationship. Therefore, the age of a player does not really affect the total points of a player.



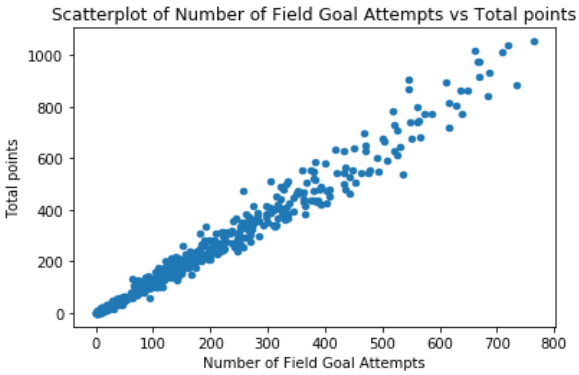
The histogram here clearly shows that a player in position SG scored the highest points among other players. However, it must be observed that PG has a higher range of players scoring from 0 to 1000 points in comparison to SG, which is slightly below 1000 points. Therefore, it is more likely for a player to score more points if positioned at PG. Both PG-SG and SF-PF had less data, therefore, it is not correct to make analysis on those positions.



The scatterplot here shows a positive linear relationship. Even though there are outliers present, the increase in number of steals means higher total points for a player. This shows that if a player makes more attempts to steal from the opposite player, it is guaranteed to generate more points, irrespective of whether it is 2P or 3P goal points. Therefore, a player must aim to make decent number of steals for more points.



This scatterplot was rather surprising among graphs. Generally, if a player makes a greater number of personal fouls, the chances to score higher points reduces. But in this case, we can see an overall positive linear relationship. Therefore, it can be concluded that making more personal fouls does not reduce chances of a player to score more points. It rather depends on the skill of the player.



As expected, there is a positive linear relationship between number of field goal attempts and total points. This is because when a player tries, there is a chance to score a goal and win points. It does not really depend on the type of the goal, as long as the player keeps on trying to make more attempts to score a goal. Therefore, players must make more field goal attempts to score higher points.

References

1. (2020, October 4). How to Replace Values in Pandas DataFrame. Retrieved March 18, 2021, from https://datatofish.com/replace-values-pandas-dataframe/
2. (2021, February 14). Check for NaN Values in Python. Retrieved March 18, 2021, from https://www.delftstack.com/howto/python/check-for-nan-values-python/#:~:text=In%20Python,%20we%20have%20the%20isnan%20()%20function,,to%20check%20for%20nan%20constants%20in%20float%20objects.
3. Descriptive or Summary Statistics in Python Pandas – Describe(). Retrieved March 18, 2021, from https://www.datasciencemadesimple.com/descriptive-summary-statistics-python-pandas/
4. Quartiles, Quantiles and Interquartile Range. Retrieved March 18, 2021, from https://www.codecademy.com/learn/learn-statistics-with-python/modules/quartiles-quantiles-and-interquartile-range/cheatsheet#:~:text=In%20Python,%20the%20numpy.quantile%20()%20function%20takes%20an,data%20=%20[1,2,3,4,5]%20first_quartile%20=%20np.quantile%20(data,%200.25)
5.Cybernetic. (2020, July 16). How to check a string for a special character?. Retrieved March 15, 2021, from https://stackoverflow.com/questions/19970532/how-to-check-a-string-for-a-special-character
6.RMIT. Practical Data Science: Data Curation. Retrieved March 18, 2021 from week2-DataCuration-1.pdf: Practical Data Science with Python (2110) (instructure.com)
7.RMIT. Practical Data Science Tute/Lab 01. Retrieved March 18, 2021 from Tute_lab 01.pdf: Practical Data Science with Python (2110) (instructure.com)
8. RMIT. Practical Data Science: Data Summarisation: Descriptive Statistics and Visualisation. Retrieved March 19, 2021 from week3-Summarisation-Clear-2021.pdf: Practical Data Science with Python (2110) (instructure.com)
9. RMIT. Practical Data Science: Tute/Lab 02. Retrieved March 19, 2021 from Tutorial02.pdf: Practical Data Science with Python (2110) (instructure.com)
10. RMIT. Practical Data Science: Tute/Lab 03. Retrieved March 19, 2021 from Tute_lab 03-2021 (1).pdf: Practical Data Science with Python (2110) (instructure.com)
11. Yang,S..,Berdine G. (2016, January 15). Outliers. Retrieved March 19 2021 from https://pulmonarychronicles.com/index.php/pulmonarychronicles/article/view/252/635
12.qbzenker. (2017, May 17). How to select rows with NaN in particular column?. Retrieved March 20, 2021, from https://stackoverflow.com/questions/43831539/how-to-select-rows-with-nan-in-particular-column
13. Anderson, A. Use Scatter Plots to Identify a Linear Relationship in Simple Regression Analysis. Retrieved March 21, 2021, from https://www.dummies.com/education/math/business-statistics/use-scatter-plots-to-identify-a-linear-relationship-in-simple-regression-analysis/
14. Frost, J. Guidelines for Removing and Handling Outliers in Data. Retrieved March 24, 2021, from https://statisticsbyjim.com/basics/remove-outliers/#:~:text=%20Guidelines%20for%20Removing%20and%20Handling%20Outliers%20in,outliers%20in%20your%20data.%20They%20can.%20More
15.Varun. (2018, September 9). Python Pandas : How to Drop rows in DataFrame by conditions on column values. Retrieved March 24, 2021, from https://thispointer.com/python-pandas-how-to-drop-rows-in-dataframe-by-conditions-on-column-values/
16. Matplotlib In Jupyter Notebook. Retrieved March 25, 2021, from https://vegibit.com/matplotlib-in-jupyter-notebook/