Final Report

NBA Salary Prediction

Abdurrahman Akkurek

1. Introduction

1.1 Background

The NBA has moved past Major League Baseball as the second-most popular sport in the United States and has millions of fans around the world. Moreover, the NBA is still growing and attracting more people as cited in Wikipedia link.

Moreover, across last year's season, the NBA generated about $8 billion in revenue. And the 30 teams making up the NBA have an average valuation of $1.9 billion each. And also with this growth and popularity, the average salary cap for each team has also skyrocketed now it is around 120 Million $ per year. How players on a team perform is the most important factor that determines which team wins the championship. Players' pays are largely based on their past performances. However, player performance changes from season to season. Each year there are a number of players who improve dramatically over last year. Those players bring a lot of value, both competitively and economically, to the teams they belong to.

That's why salary negotiation is vital for players along with their performances.

1.2 Problem

Season data that might contribute to determining player salary includes his performance last season, his age, his experience, his position, and other stats that describe what kind of player he is. This project aims to predict whether and how much salary a player will be worth for the next season based on these data.

Beside this,this prediction might help players during salary negotiation and provide them with insight to foresee what they are worth before signing any contract.

1.3 Interest

Obviously, NBA teams and players would be very interested in accurate prediction of player salary, it is useful for players as a competitive advantage and for teams as business values. Others who are interested in the NBA such as fans and fantasy basketball players may also be interested.

1.4 Questions

1- Which statistic or statistics are the best indicators(predictors) for the player’s salaries?

2- How reliable is the machine learning model I will build to predict NBA salaries?

3- Measuring the error of the model and analyzing top 20 players with their predicted salaries and actual salaries to see if it can be predicted correctly.

2. Data acquisition and cleaning

2.1 Data sources

NBA player stats, position, age, and draft position data can be found on here. However, this data has only stats not years of experience and salary information. I scraped basketball-reference.com for players drafted beteen 1998-2019 and player salary for the 2018 season. Datasets can be also downloaded as csv files from the same website. Initial dataset has 578 players stats and 18 attributes..

2.2 Data cleaning

Data downloaded or scraped from the sources given above were loaded and combined into one Pandas dataframe. While reading the csv file, I encountered an encoding error, that is why files were loaded with (encoding = "ISO-8859-1) parameter.

There are several problems with the datasets. First, there were a lot of missing values for FG%, 2P% and 3P% because of not having any FGA and 3PA for that season.It is double checked and seen that those values are zeros. I decided to fill those values with 0 because there is no field goal attempt at all.

Second, there are players with duplicated names, which is because those players changed their team during mid season. That is why, I decided to drop duplicates and leave the one that player played the most minutes per game. Players were sorted in descending order by minutes and then dropped duplicates by keeping the first duplicate. Now our dataset has 402 rows, 34 columns.

Third, the salary column was categorical and had $ sign. $ sign was removed and the column was converted to numeric. Beside that position column had ‘C’, ‘PG’, ‘SG’, ‘SF’, ‘PF’ as 5 different positions. I wrote a script and assigned numbers for each position, which is also used in the NBA. ‘PG’ is number 1, ‘C’ is number 5 position.

After fixing these problems, I dropped these columns because they will not be used throughout analysis. ['Rk', 'username_y', 'username_x', 'eFG%', 'PF', 'ORB', 'DRB', 'TOV', 'GS', 'FG', '3P', '2P', 'FT',

'BLK', 'FTA', 'FT%', 'STL', '2PA', '2P%']. The reason why I dropped these because I already have another stat represents most of these columns. For example, TRB(total rebound) is equal to addition of ORB(offensive rebound) and DRB(defensive rebound). We don't need these two columns separately.

Moreover, these column names were changed to more meaningful ones to be more clear about what they are. ('Salary 2018-19': 'salary', 'Yrs': 'Years', 'Pos': 'Position', 'Tm': 'Team', 'G':'Game', 'MP':'Minutes', 'TRB':'Rebound', 'AST':'Asist', 'PTS': 'Point'). And then, for aesthetics of the data and visualizations, I wrote scripts that classify Point, Assist, Salary, Age, Years, and Minutes, Games, and Three point attempts and most of them were reduced to 10-15 different groups instead of more. Years column was displaying years of experience as of 2019, that is why, I wrote another script that all years were reduced by 1 to have 2018 season experiences data.

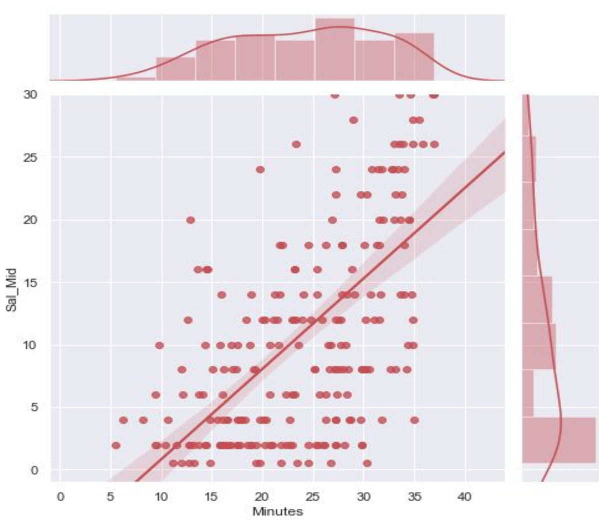
2.3 Feature selection

After data cleaning, there were 402 samples and 34 features in the data. After examining the meaning of each feature, it was clear that there was some redundancy in the features. For example, there was a feature of the number of rebounds a player collected, and another feature of defensive and offensive rebounds he collected. These two features contained very similar information (a player's ability to rebound). Such total vs. rate relationships also existed between other features. These features are problematic for two reasons: (1) A player's certain abilities were duplicated in two features. (2) A player's playing time was duplicated in multiple features. In order to fix this, I decided to keep all features that were total in nature, and drop their cumulative counterparts (Table 1). There were also other redundancies, such as that total rebounds are the sum of offensive rebounds and defensive rebounds. For features that can be calculated by sum of other features, I decided to drop them (Table 1). 2 After discarding redundant features, I inspected the correlation of independent variables, and found several pairs that were highly correlated (Pearson correlation coefficient > 0.9). For example, shots attempted, shots made, and points scored were highly correlated. This makes sense, after all, you score points by making shots. From these highly correlated features, only one was kept, others were dropped from the dataset.For example Years and Age is also highly correlated. Age was dropped. After all, 15 features were selected

Table 1. Simple feature selection during data cleaning.

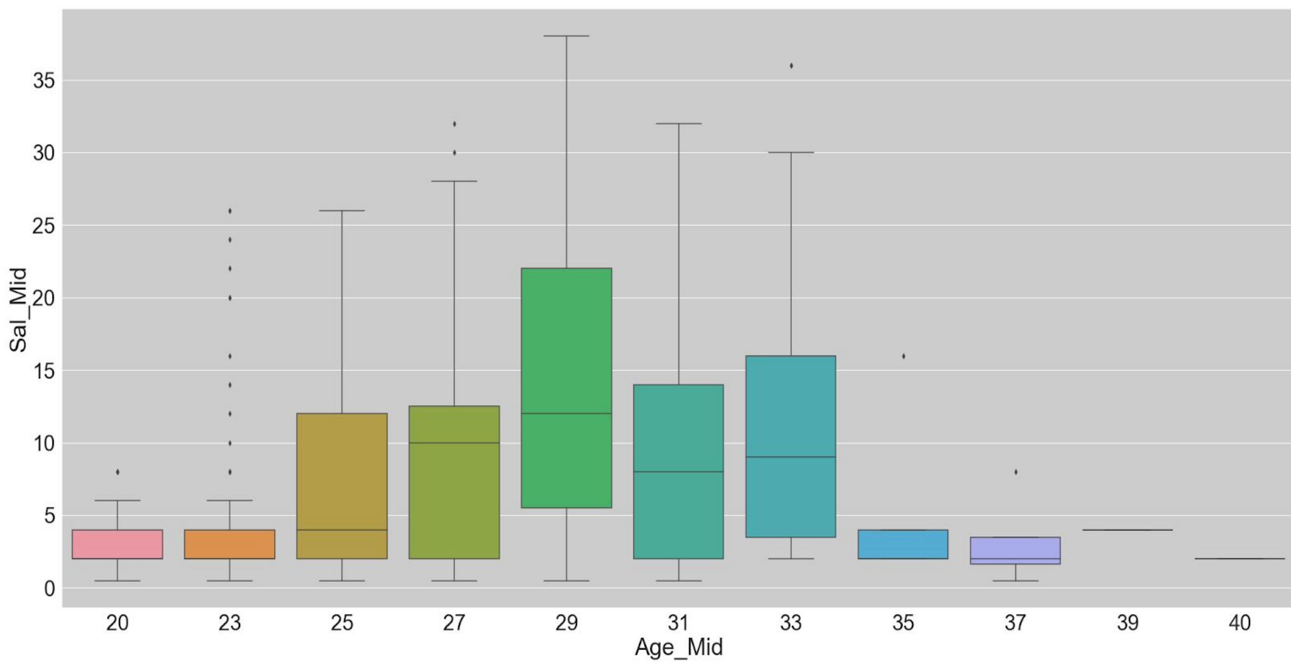
Kept features	Dropped features	Reason for dropping features
TRB	DRB, ORB	Total = offense + defense. Dropped defense.
FGA, FG, FG%, 3PA	2PA, 2P, 2P%, 3P, 3P%	Field goal = 2-point shots + 3-point shots. Dropped 2-point and 3-point shots. Just kept 3PA
Asist, Point, Games played, Minutes, Years	FTA, FT%, FT, PF(fouls), BLK(blocks), STL(steals), GS(games started)	FT% are very similar, no need to add in our prediction, personal fouls and steals are the same. We just kept the main attributes.

2. Exploratory Data Analysis (EDA)

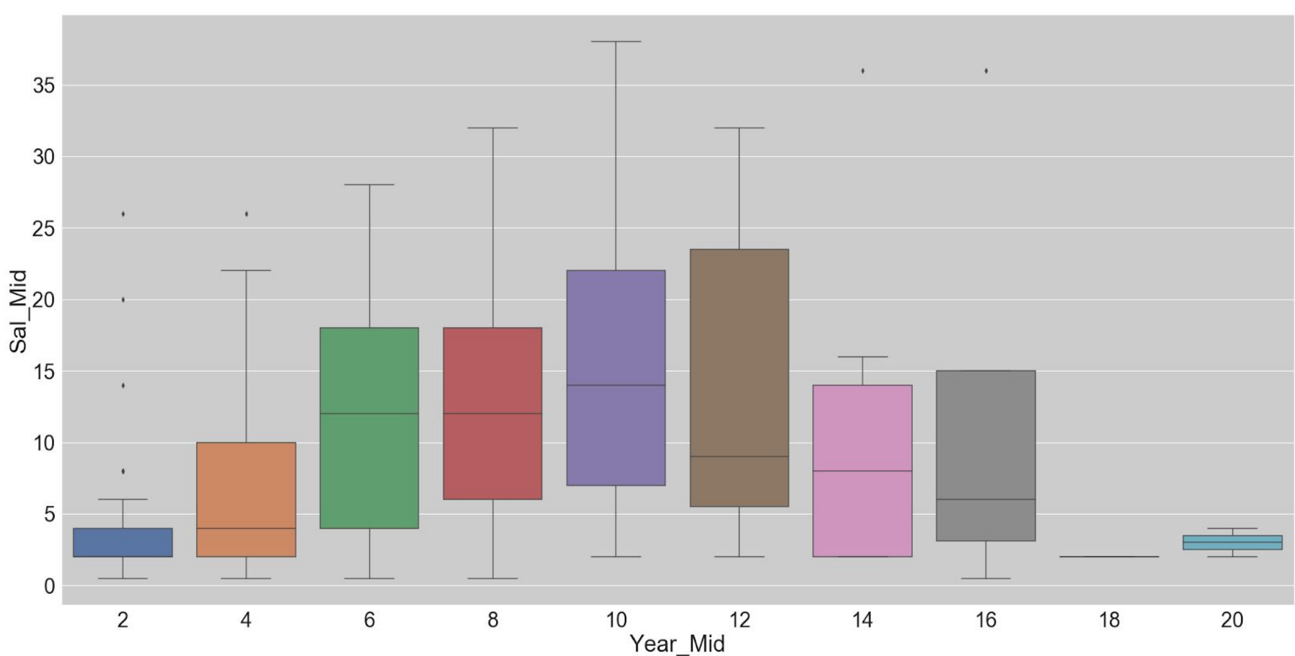


There were so many outliers in our data, many players who played less than 10 minutes and obviously they did score almost any point and they didn't earn a good salary. That's why players who played less than 10 minutes will be dropped.

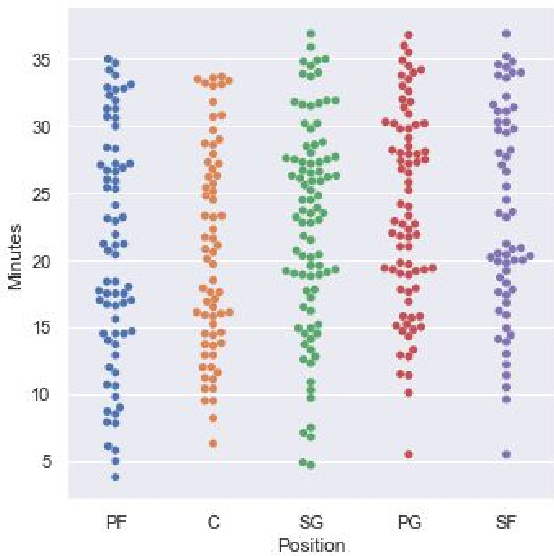
And also, players who did not play more than 20 games were also dropped. A season has 82 games and players should be eligible at least quarter of these games for the sake of analysis.

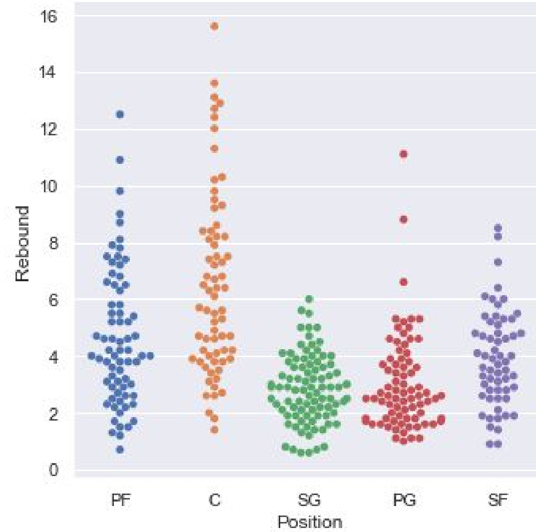


Based on this box plot, we see that ages between 27-33 are the most productive ages for players. And also this plot shows that retirement age is around 35 for NBA players. The reason why 20 years old players are not earning well is because rookie players have a 3-year rookie contract which is relatively small, and they cannot change their salary until it is over.

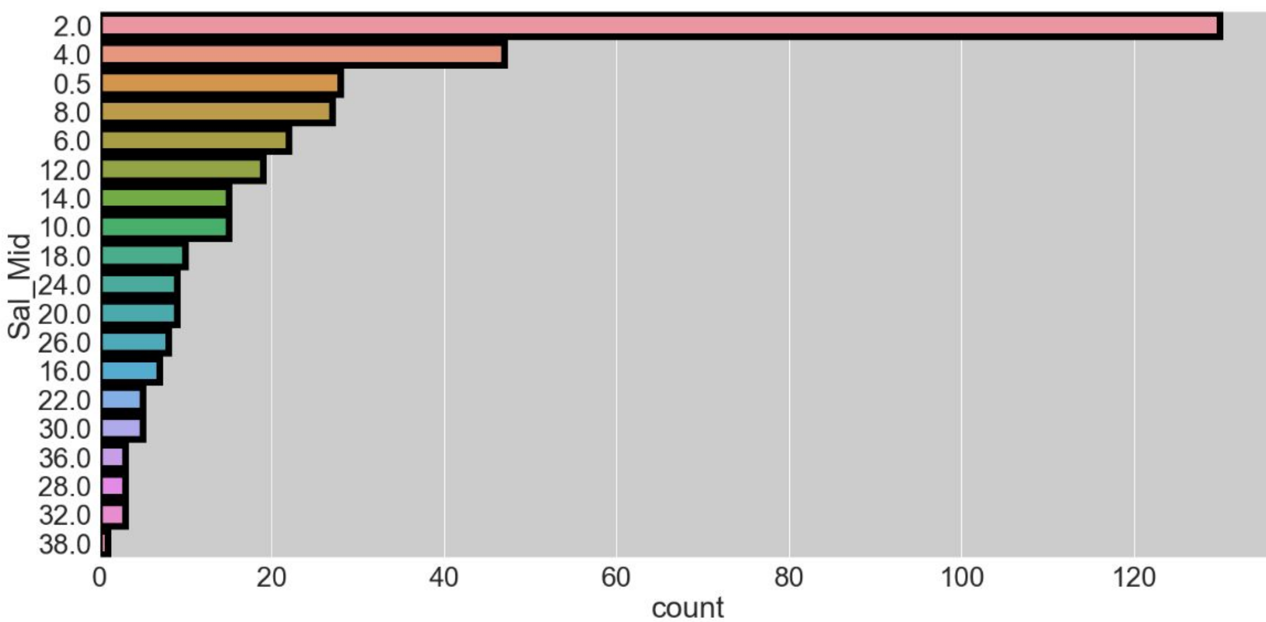


This plot also shows that experience between 8-12 years is the most productive year.

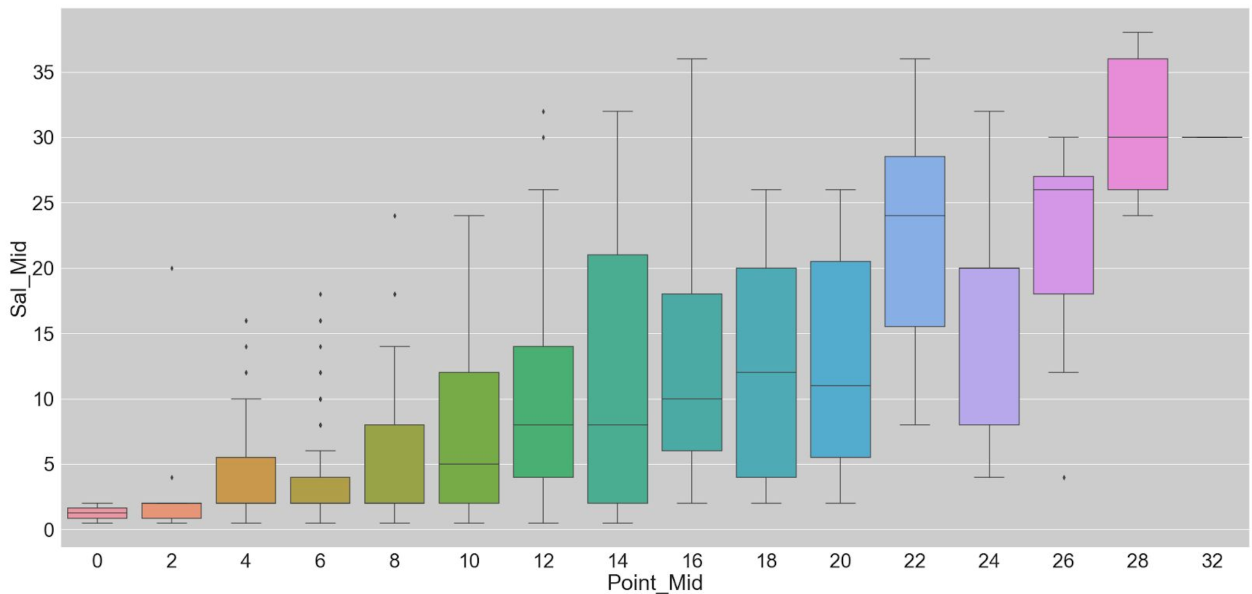




On the scatter plot above, what we notice is that the importance of position differs based on what statistics we are looking for. Center players are able to collect most rebounds along with Power-forward players, however, they are having slightly less minutes than other positions.



In this count plot, we see that most players are earning between 0-14 Million $. It is not easy to earn huge amount of money in the NBA unless you show outstanding performances.



On this box plot, the correlation between points and salary is obviously noticed.If you can score around 28 points, you kind of guarantee over 25 Million $ up to 40 Million $.

3. Statistical Data Analysis

To analyze my variables, I, first, looked at the scatter plot shown below, to see if positions of players affect their points, salary and minutes they play. I mainly focused on two main positions.

I found these results:

1- There is a statistically significant difference between minutes of PG and C players.

2- There is a statistically significant difference between points of PG and C players.

3- Salaries of center players ARE NOT significantly different than point guard players.

4- Correlations:

'Point' :0.61, 'Asist':0.52, 'Rebound': 0.48, 'Defensive Rebound' : 0.52, 'Free Throw': 058, 'FG'(Field Goal Made): 0.59, 'Minutes':0.58, and 'Game Started': 0.53, 'Steal': 0.51, 'Turnover': 0.57

- **C** = **Centers** who are usually the tallest players in the team and defending rim from short range shots and collecting the rebounds.
- **PG** = **Point-guard** who are the brains of the team sets the game, holds the ball mostly and directs other players.

Some information about plots that will be shown below:

Position	salary	Point	Assist	Rebound
C	8.284385e+06	9.374648	1.446479	6.319718
PF	7.762332e+06	8.860870	1.476812	4.586957
PG	8.833254e+06	11.453030	4.328788	3.157576
SF	9.108709e+06	10.518367	2.040816	3.942857
SG	6.543541e+06	10.253659	2.085366	2.770732

Summary statistics for positions

Mean of Minutes and Points of Center and PG players shown on the left

Position	Point	Minutes
C	9.374648	19.888732
PG	11.453030	24.027273

If you look at the statistics, we can easily see the difference in Assists, which shows that PG players are able to assist more(directing the game as said before) and stay in the game longer than center players.
And also, they are earning slightly more than Center players. PG players are also scoring leaders as seen in the table.

T-tests were performed to prove these observations.

H0 = Null Hypothesis =>Center players are staying in the game as long as point guard players.

H01 = Null Hypothesis =>Center players are scoring per game as much as point guard players.

HA = Alternative Hypothesis => There is significant difference between minutes of Center players and minutes of PG players.

HA1 = Alternative Hypothesis => There is significant difference between points of Center players and points of PG players.

p-value is 0.002 and t-value is 3.2143507802050766 for minutes
Since p-value is 0.002 less than 0.05 for minutes, we reject the null hypothesis, and there is a statistically significant difference between minutes of PG and C players.

p-value is 0.038 and t-value is 2.102429387246607 for point
Since p-value is 0.038 less than 0.05, we reject the null hypothesis, and there is a statistically significant difference between points of PG and C players.

Another T-test was performed to see if they are actually earning different amounts;

H0 = Null Hypothesis => Salaries of center players are not significantly different than salaries of point guard players.

HA = Alternative Hypothesis => There is significant difference between salaries of different positions.

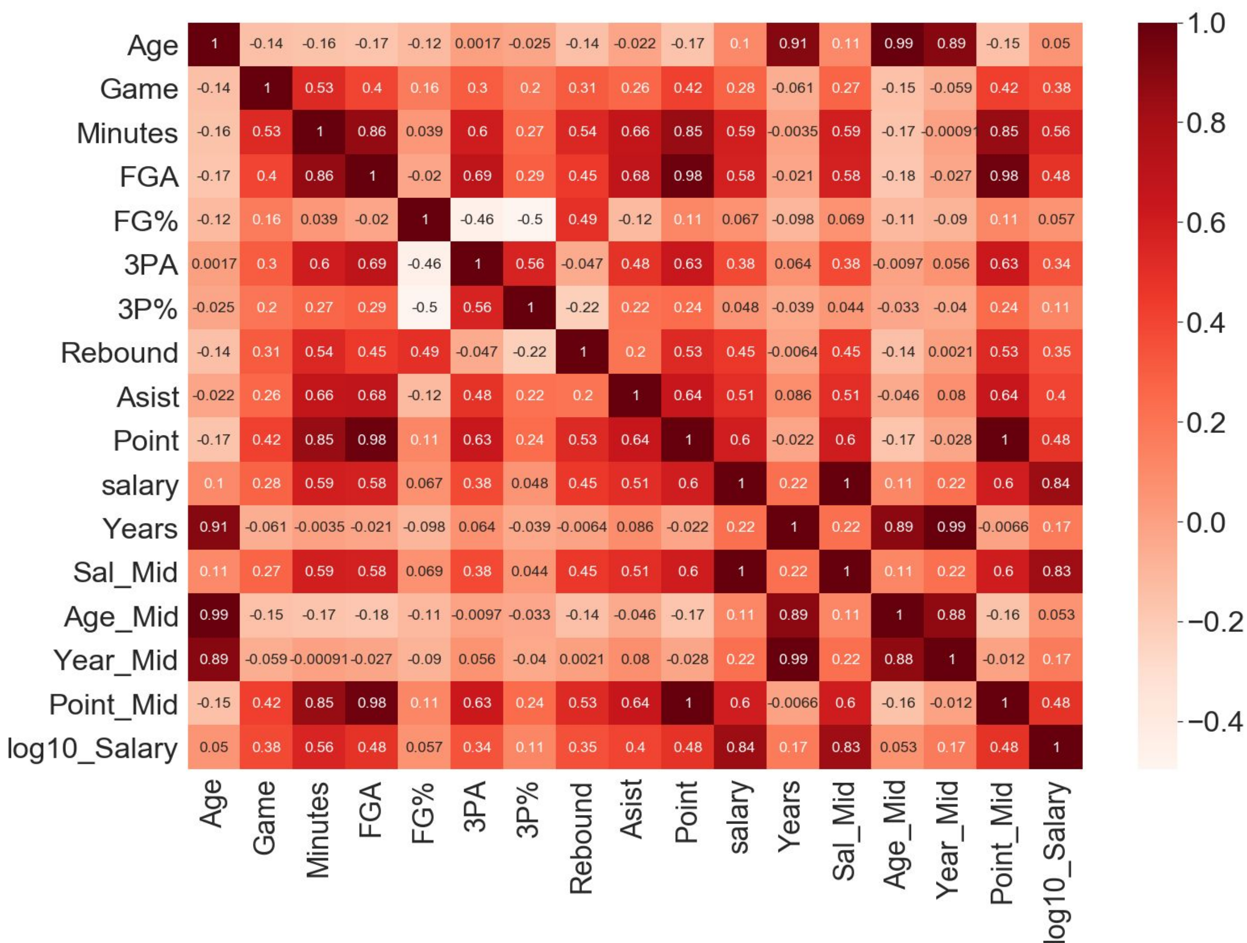
p-value is 0.344 and t-value is 0.9515878705510601

Since p-value is 0.344 more than 0.05, we failed to reject the null hypothesis, and salaries of center players ARE NOT significantly different than point guard players.

Seaborn heatmap is produced to see the correlations between dependent and independent variables.

Dependent variable = salary

Independent variables = all other stats



When we just take a look at the heat map and the correlations between salary(dependent variable) and other stats(independent variables), These independent variable are highly correlated with our dependent variable salary ->

'Point' :0.61, 'Asist':0.52, 'Rebound': 0.48, 'Defensive Rebound' : 0.52, 'Free Throw': 058, 'FG'(Field Goal Made): 0.59, 'Minutes':0.58, and 'Game Started': 0.53, 'Steal': 0.51, 'Turnover': 0.57

Pearson correlation coefficient between point and between salary is **0.599**

Pearson correlation coefficient between minutes and between salary is **0.595**

Pearson correlation coefficient between assist and between salary is **0.508**

Pearson correlation coefficient between field-goal and between salary is **0.581**

Pearson correlation coefficient between rebound and between salary is **0.450**

Pearson correlation coefficient between three-points and between salary is **0.378**

By looking at the heatmap we were able to see some correlations between independent variables;

When we just take a deeper look at the heat map and the correlations between independent variables, These independent variables are highly correlated with each other ->

'Point' : 'FGA' = **0.98**, The more shoot trials the more points.

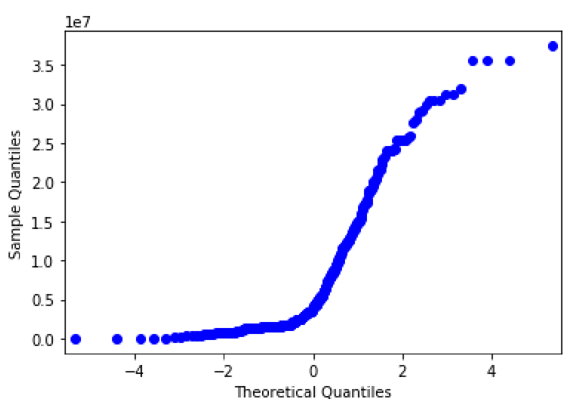
'Point' : 'Asist' = **0.64**, The more assists to teammates the more shooting trials.

'Point' : 'Minutes' = **0.85**, The longer staying in the game the more chance to score.

There is also negative correlation between rebound and 3P%, we can easily say that Center players are not good at shooting 3 points.

'Rebound' : '3P%'(3 point percentage) = **-0.22.**

This QQ Plot shows that our salary data is not normally distributed. That's why we applied bootstrapping tests and results were the same with hypothesis testing results.



4. Machine Learning

For this final section, I ran many different regression models to find the best R^2 value possible. I started out with a univariate OLS using only points per game, the highest correlated feature with salary. I achieved an R^2 value of 0.371, which if you look at the correlation coefficient I found earlier in section 3, is that value squared. I then moved onto two more OLS regressions, one with 3 features strongly correlated with salary, and the other with the 4 features I ended up with in the last section. These each resulted in increases of 0.05 to the R^2 value, with the 4 feature-OLS yielding a 0.03 R^2 score. Judging by the type of dataset that I have, a score of 0.434 is not bad, considering that 1 is completely correlated. So just using this model may prove a decent predictor of a player's salary. I must also consider that with these 3 OLS models, I have used 100% of the data in my training set, and have not yet tried to predict the salaries of unseen data. Then I have used the same OLS models with my filtered data, in which rookie players were eliminated and the results were 0.332 R^2, 0.401 R^2 and 0.425 R^2 respectively. Excluding rookies did have much help to improve scores.

In order to test my regression model with unseen data, I split my dataset into a training set consisting of 80% of the data, and a testing set using the remaining 20%. After training the OLS with 4 variables with the training set, I then predicted the salaries of the test set and took the correlation between the actual values of the salaries and the predicted values of the salaries. I then squared this to get the R^2 , which is what Linear Regression models typically use to score their accuracy. For this particular testing set, the value increased by 0.19 from using the entire set as training and became 0.443.

I moved on to using a Linear Regression model from the sklearn package. With this model I used the same 4 variables used in the OLS and received a score of 0.427. This may seem like a decline, moreover, after cross validation, the score decreased to 0.359. One thing to note is that when I initially ran my cross validation, I was receiving incredibly low scores. Scores lower than I even thought possible (in the negatives). I realized that this was because my dataset was ordered by salary, and certain sections of the data, when isolated, either have a negative or no correlation at all with the salary. For example, there are 20 players with the same minimum 1- year salary, no matter what their stats are. The way the cross_val_score module from the sklearn package works is that it splits the data into a number of slices based on the user's input, and uses each slice as a testing set. But the way it slices the data is chronological and does not do any shuffling, so it picks up on these spots of the data with poor correlations. In order to remedy this, I pre-shuffled the data before feeding it into the cross validator.

My next two regressions are Lasso and Ridge regressions. These two operate differently from Simple Linear regression and OLS by implementing a function that penalizes features with poor correlations to the target by reducing their coefficients either to 0 like in Lasso Regression, or down to a small value like in Ridge. Because of this, I used all of the features for Lasso and Ridge regression and achieved cross-validated scores of 0.35 and 0.53 respectively. Based on these results, Ridge Regression seems to be the best model to use moving forward.

I applied the same tests for my filtered data, and I got 0.58 R^2 value for Ridge regression, which is the highest score so far.

After trying out all of these various models with different meta parameters, cross validation, and with various features, I have decided to conclude that the Ridge Regression with a filtered data is the best model for me to predict players' salaries.

VI. Concluding Thoughts and Revelations

There are a few main things to take away from these findings. The first is that because of the way basketball is played these days, many of the features in this dataset are correlated with one another, leading to only 1 principal component. Some other things are that the correlations between points, rebounds, assists, and minutes per game with salary are the numbers that I found in my initial EDA at a 95% confidence interval. This means that the statistics that allow for the most guaranteed increase in salary are points per game, games started and minutes. I also found out that even though points per game has a strong correlation with salary, increasing points per game by 1 will not result in that large of an increase in salary, so you would have to increase your points per game by more than just 1 if you wanted to see results in your salary. On the other hand, changing your assists may not be as guaranteed to increase your salary, but if you do increase your assists per game by 1, your salary can potentially be increased by over $1,746,000. So, although my model may not be able to predict exactly what a player should be paid, I can come up with a good ballpark number that the Teams' General Managers or the player can use when deciding on business decisions. The lack of predictive power in my model most likely arises from non-statistical factors that go into the salary, such as popularity, previous injuries, and players who willingly take pay cuts in order to stay on a team that has a better chance of making it to the finals.