JinShuo Chen

Final Report:

NBA Players' Salary Analysis

Problem Statement

NBA is one of the most lucrative professional sports. The average salary has continued to increase as time passed by. Currently, the highest paid NBA player is Stephen curry 44.4 million per year. How do the National Basketball Association (NBA) team own decide the player's salaries? The problem I want to solve is how to predict NBA player's salary. That means can statistics be used to determine an NBA player salary. And if yes, what are the most important features. Why predicting NBA player salary are important? Knowing how much the players will cost next year will help team manager to plan the team's annual budget and understand the team cost. Team manager can also use the information to benchmark with competitor teams and make sure that the play is paying fairly to avoid talent churn.

The dataset I am going to use is the Basketball Dataset from Kaggle which is sourced from stats.nba.com. It contains statistics information for more that 4500 NBA players. My method is to use all features of a player that are can be potentially determine the player salary in model to predict player salary. And since salary is numeric values, I am use three regression model: linear regression, random forest regression and KNN regression.

Data Wrangling

The original data set is SQLite database. It contains data on all games, all teams and all players within the NBA. To predict player salary, I am only interested in players' information which are the table "Player_Attributes" and table "Player_Salary". Table "Player_Attributes" has 4500 rows and 37 columns data. Each row is one player, and each column is the player's attributes such height, weight, season experience, average points, assists and rebounds. Table "Player_Salary" has 1293 rows and 12 columns data. Each row contains each player's current season salary and future seasons salary. Next step is to join the tables together. And the best way joins the tables is to join it by NBA player name. However, the 'Player Salary" table contains current season salary and future season salary. And I found interesting that some player salaries stay same and some increase. However, that means there are few rows that have same player name. To solve the problem, I grouped salary together using the average.

I started with creating SQLite engine and query data by selecting 'Player_Attributes' table and 'Player_Salary' table using inner join to combine them into one table by matching the palyer's names. The combined table has 4500 rows and 40 columns. Here is information of the combined table.

```
Data columns (total 40 columns):
 #   Column                            Non-Null Count  Dtype
---  ------                            --------------  -----
 0   ID                                353 non-null    object
 1   FIRST_NAME                        353 non-null    object
 2   LAST_NAME                         353 non-null    object
 3   DISPLAY_FIRST_LAST                353 non-null    object
 4   DISPLAY_LAST_COMMA_FIRST          353 non-null    object
 5   DISPLAY_FI_LAST                   353 non-null    object
 6   PLAYER_SLUG                       353 non-null    object
 7   BIRTHDATE                         353 non-null    object
 8   SCHOOL                            353 non-null    object
 9   COUNTRY                           353 non-null    object
 10  LAST_AFFILIATION                  353 non-null    object
 11  HEIGHT                            353 non-null    float64
 12  WEIGHT                            353 non-null    float64
 13  SEASON_EXP                        353 non-null    int64
 14  JERSEY                            353 non-null    object
 15  POSITION                          353 non-null    object
 16  ROSTERSTATUS                      353 non-null    object
 17  GAMES_PLAYED_CURRENT_SEASON_FLAG  353 non-null    object
 18  TEAM_ID                           353 non-null    object
 19  TEAM_NAME                         353 non-null    object
 20  TEAM_ABBREVIATION                 353 non-null    object
 21  TEAM_CODE                         353 non-null    object
 22  TEAM_CITY                         353 non-null    object
 23  PLAYERCODE                        353 non-null    object
 24  FROM_YEAR                         353 non-null    object
 25  TO_YEAR                           353 non-null    object
 26  DLEAGUE_FLAG                      353 non-null    object
 27  NBA_FLAG                          353 non-null    object
 28  GAMES_PLAYED_FLAG                 353 non-null    object
 29  DRAFT_YEAR                        353 non-null    object
 30  DRAFT_ROUND                       353 non-null    object
 31  DRAFT_NUMBER                      353 non-null    object
 32  PTS                               353 non-null    float64
 33  AST                               353 non-null    float64
 34  REB                               353 non-null    float64
 35  ALL_STAR_APPEARANCES              4 non-null      float64
 36  PIE                               349 non-null    float64
 37  nameTeam                          353 non-null    object
 38  namePlayer                        353 non-null    object
 39  current_avg_salary                353 non-null    float64
```

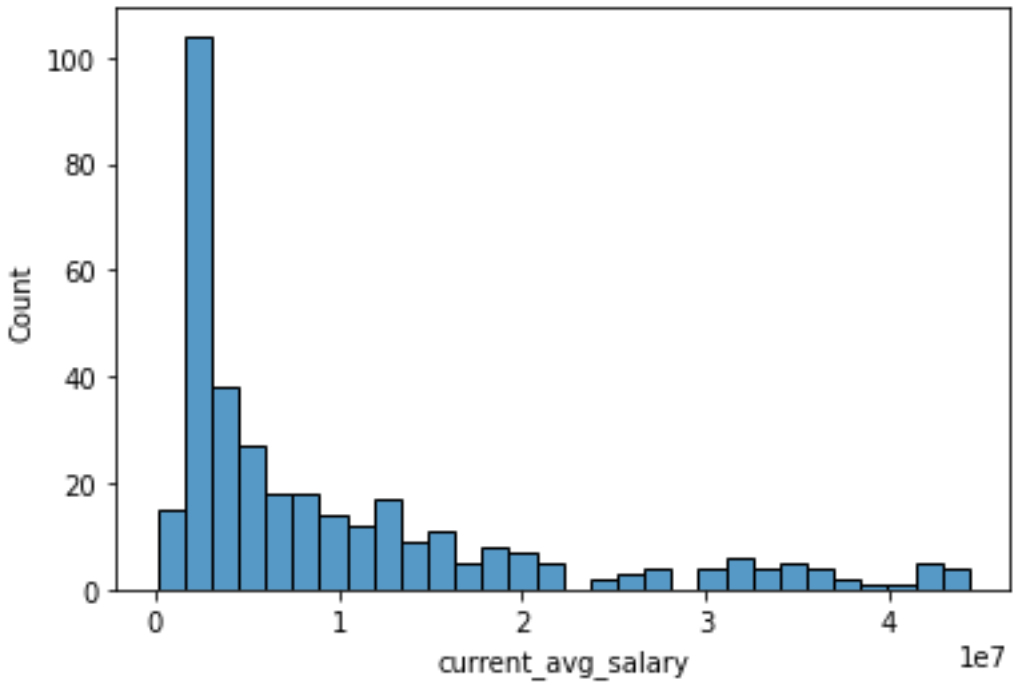
Then, I dropped the 'ALL_STAT_APPEARANCES' column that only has 4 non-null value. But I think 'ALL_STAT_APPEARANCES" maybe an important fact that determine the salary. In the future study, if I found the data, I will include it in my model. After that, I fill null values of PIE column with average. In addition, I converted player's birth to age, then drop the birthday column. And use 'TO_Year' column

minus ‘FROM_year’ column to calculate how many years the player played and named it ‘SEASON_EXP’. Next step, I dropped the columns that obviously did not relate the salary such as 'ID', 'FIRST_NAME', 'LAST_NAME', ect. Finally, the clean data has total 18 columns that contains all potential salary related player attributes and player performance.

```
 0   DISPLAY_FIRST_LAST   353 non-null    object
 1   SCHOOL               353 non-null    object
 2   HEIGHT               353 non-null    float64
 3   WEIGHT               353 non-null    float64
 4   SEASON_EXP           353 non-null    int64
 5   JERSEY               353 non-null    object
 6   POSITION             353 non-null    object
 7   TEAM_NAME            353 non-null    object
 8   DRAFT_YEAR           353 non-null    object
 9   DRAFT_ROUND          353 non-null    object
 10  DRAFT_NUMBER         353 non-null    object
 11  PTS                  353 non-null    float64
 12  AST                  353 non-null    float64
 13  REB                  353 non-null    float64
 14  PIE                  353 non-null    float64
 15  current_avg_salary   353 non-null    float64
 16  age                  353 non-null    int64
 17  year_played          353 non-null    int64
```

Data Exploratory Analysis

First, I plotted histogram the salary to see the distribution.



As we can see the salary distribution is non-normal distributed instead it is right skewed which makes sensor by knowing the start players high salary then non-start player. The average salary is about 10

million, but the median salary is only about 5.6 million. The standard deviation is about 10.8 million

```
count    3.530000e+02
mean     1.009223e+07
std      1.084653e+07
min      9.902000e+04
25%      1.842959e+06
50%      5.655148e+06
75%      1.366667e+07
max      4.439366e+07
```

As we can see, the NBA player salary are unbalanced. Now let see who the most pay NBA player are.

```
DISPLAY_FIRST_LAST
Stephen Curry        44393664.0
James Harden         44310840.0
John Wall            44310840.0
Russell Westbrook    44211146.0
Chris Paul           42784980.0
```

As expect, Stephen Curry has the highest pay, but James Harden and John wall are always close to Stephen Current. All top five highest pay player have more than 40 million salaries.

The consensus of basketball fanatics is the wages are based on points scored. Here are top five players the most point score. Stephen Curry is one of them. Seems like salary dose relate the point score.

```
DISPLAY_FIRST_LAST
Bradley Beal             31.8
Damian Lillard           30.1
Joel Embiid              29.9
Stephen Curry            29.0
Giannis Antetokounmpo    29.0
```

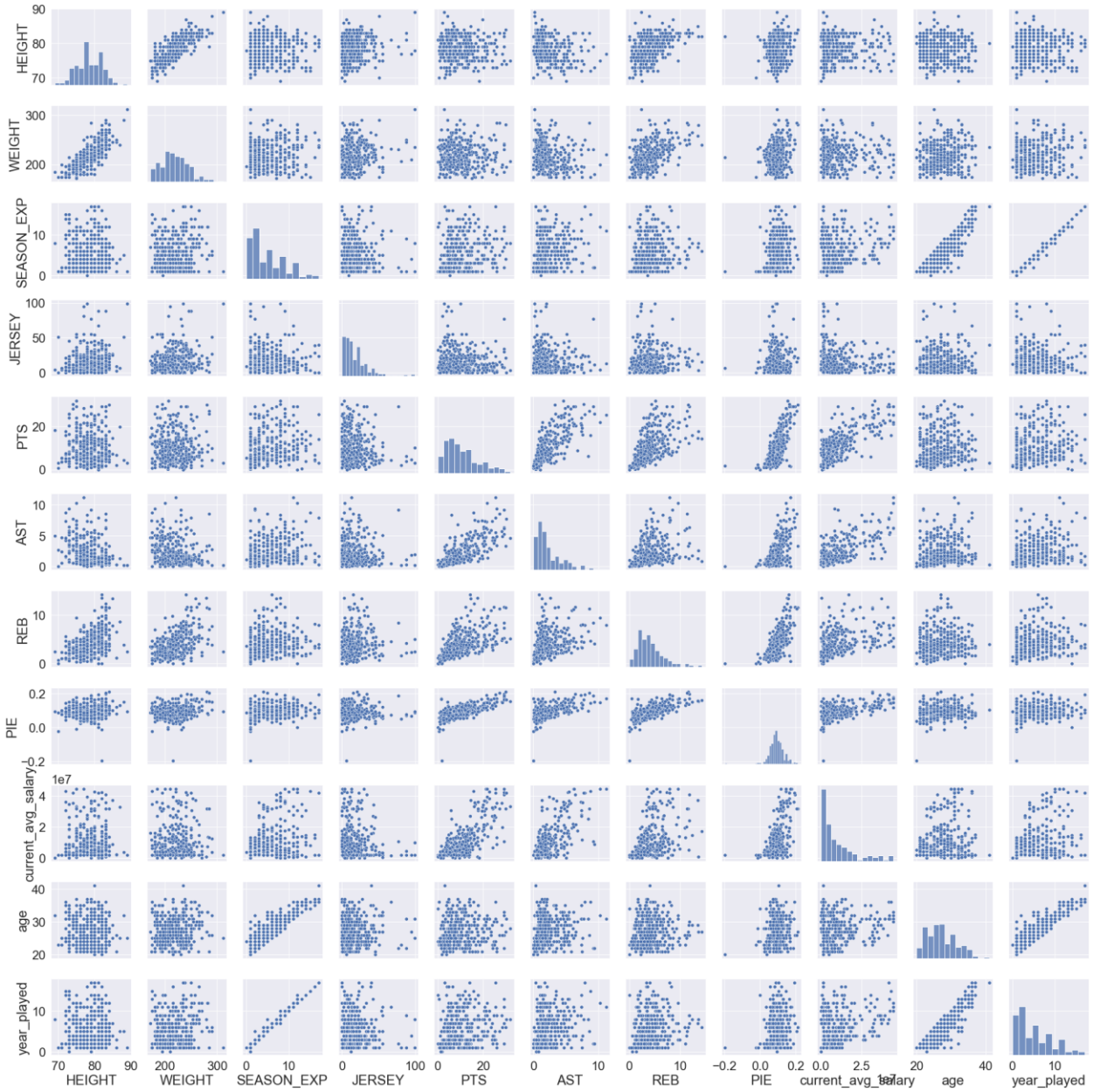
Here are top five players have the most assists. As we can see James Harden and Chris Paul who in the top five salary list are also in the assist list. Seems like assist have even stronger relationship with salary.

```
DISPLAY_FIRST_LAST
James Harden         11.2
Russell Westbrook    10.3
Trae Young            9.4
Luka Doncic           9.2
Chris Paul            8.8
```

And then, here are the top five players have the most rebounds. None of them are in the top 5 salary list. Seems like rebounds are less important compare the scores points and assist.

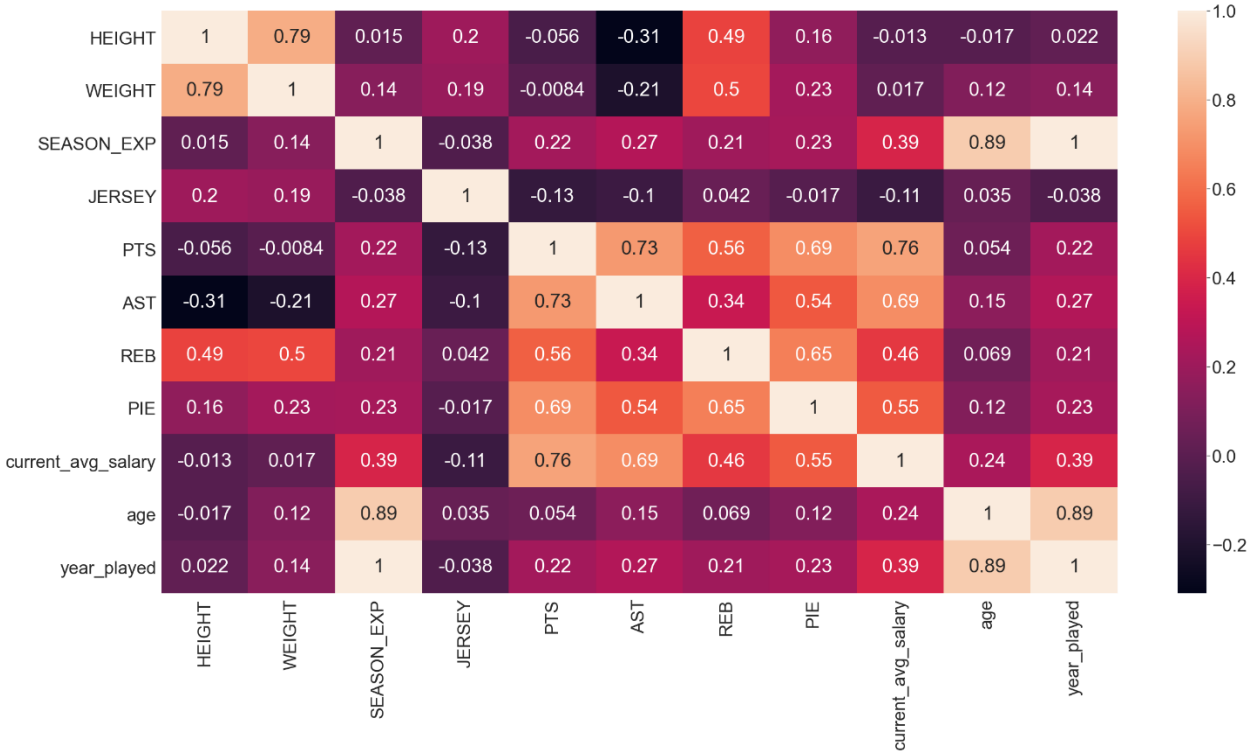
```
DISPLAY_FIRST_LAST
Clint Capela                14.2
Andre Drummond              13.5
Rudy Gobert                 13.4
Jonas Valanciunas           12.3
Giannis Antetokounmpo       11.7
```

Next, I make a pair plot of the data the to see the correlation between the columns.



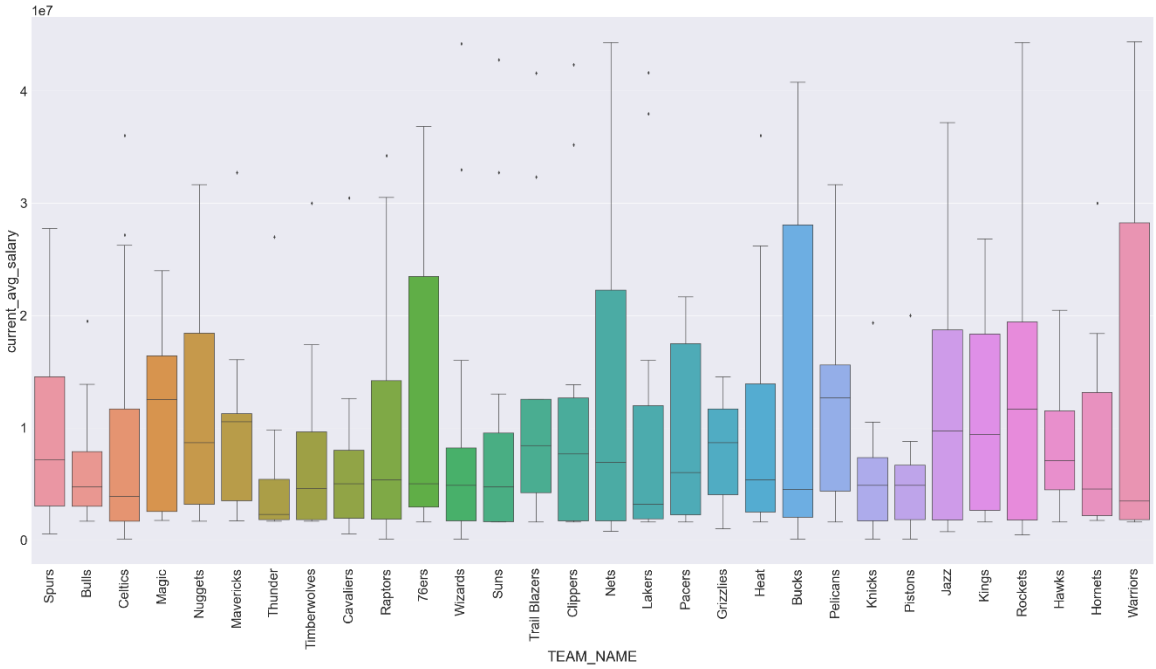
Here, I am only interested in the correlations between salary and other columns which has shown in the third of the last column. As I expect, I found that PTS (points), AST (assist), REB (rebound) and PIE (player impact estimate) have strong correlation with salary. And it is interesting that height and weight seem

do not have correlation with salary. One reason could be NBA players have similar height and weight at each position. I then plotted the heat map to see how strong the colorations are.



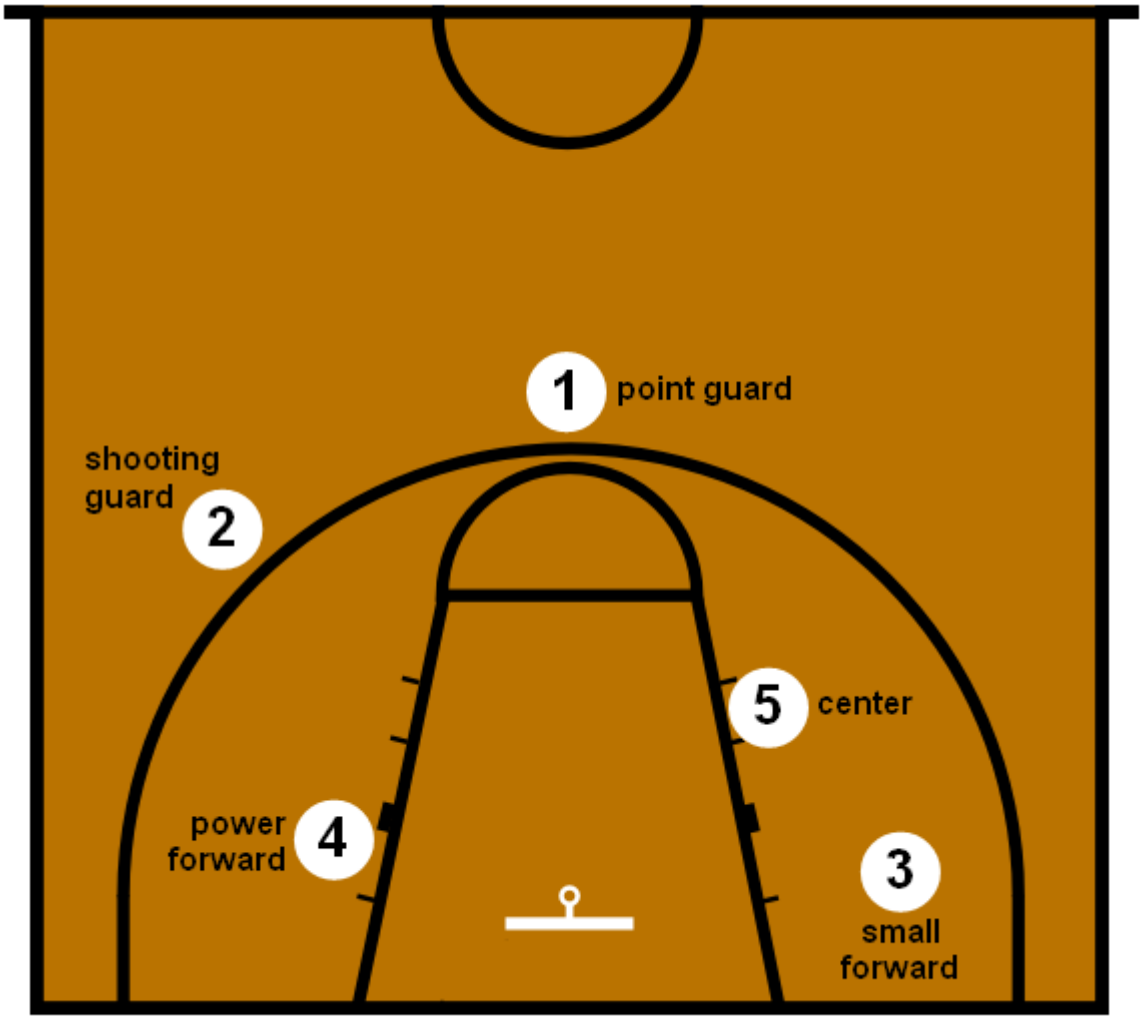
From heat map, I see PTS has the highest correlation 0.76 with salary. And assist is the second highest 0.69. And rebound is relative lower than those two about 0.46.

After exploring the numerical values, I then explore the categorical values.

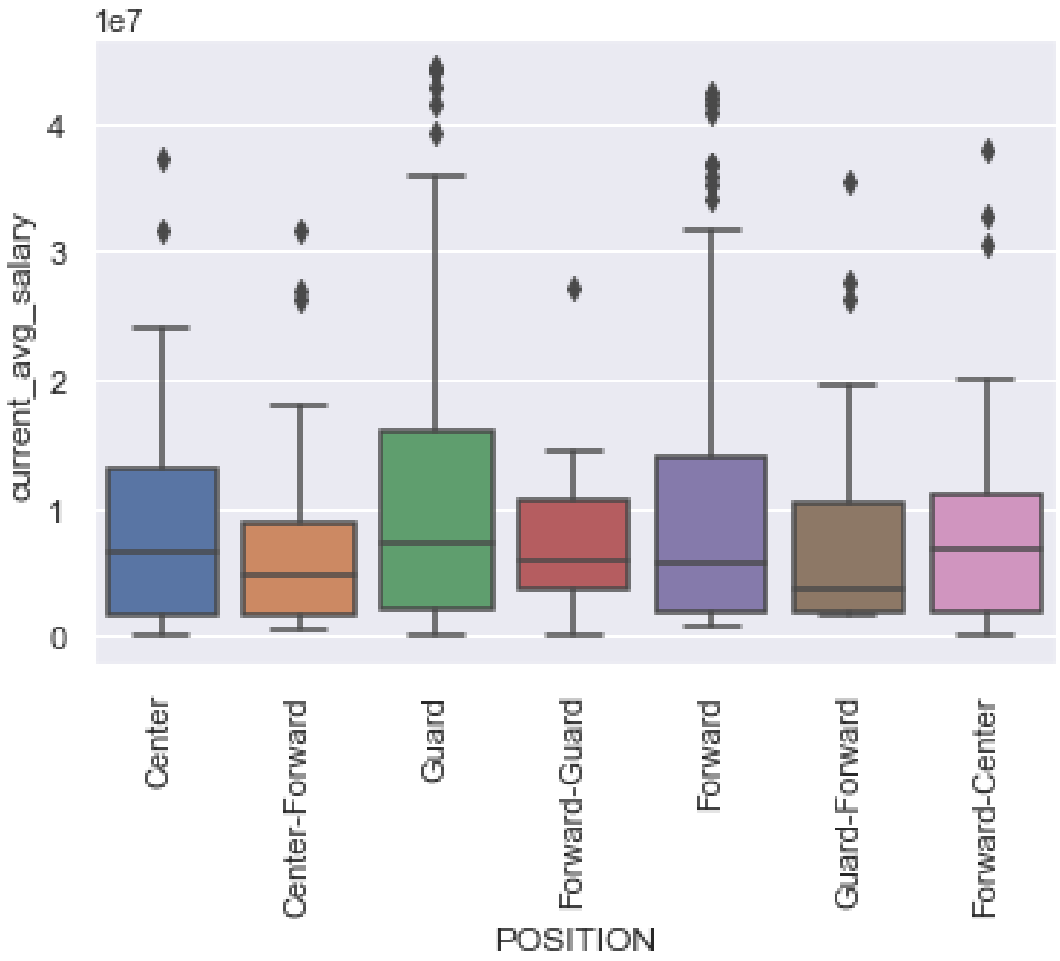


As we can see from the graph above, most of the teams have one to two outliers. They are the star player of the team have much high salary than other non-star teammates. Since I am sure how those outlier star player will affect my model, I will keep them for now.

The next categorical value I explored is position. Each game has total 10 players, 5 for each team. Typically, there are 5 positions, center, power forward, small forward, point guard and shooting Guard for each team. The center is the tallest player on each team, playing near the basket. On offense, the center tries to score on close shots and rebound. But on defense, the center tries to block opponents' shots and rebound their misses. The power forward does many of the things a center does, playing near the basket while rebounding and defending taller players. But power forwards also take longer shots than centers. The small forward plays against small and large players. They roam all over on the court. Small forwards can score from long shots and close ones. The point guard runs the offense and usually is the team's best dribbler and passer. The point guard defends the opponent's point guard and tries to steal the ball. The shooting guard is usually the team's best shooter. The shooting guard can make shots from long distance and is a good dribbler.

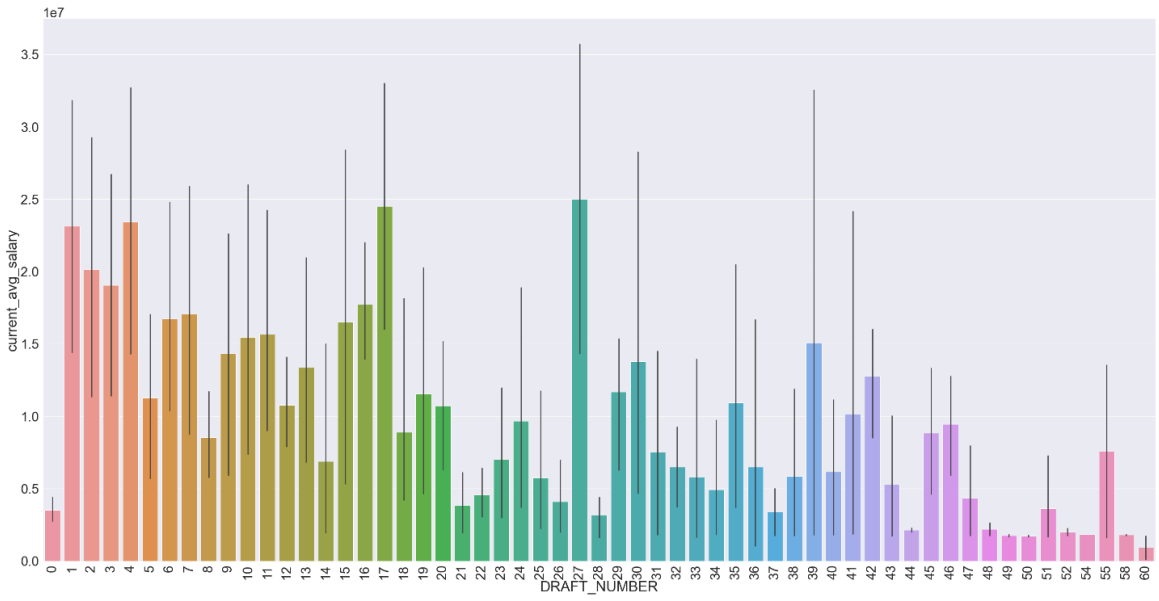


Most of players can player two position. So, guard mean the player can paly point guard and/or shooting guard. Center-Forward means the player have played both forward and center on a consistent basis. And Forward-Guard is a mix of small forward and shooting guard.



I found guard has the highest average salary which makes sensor if score points, and assist are the most important features to determine salary.

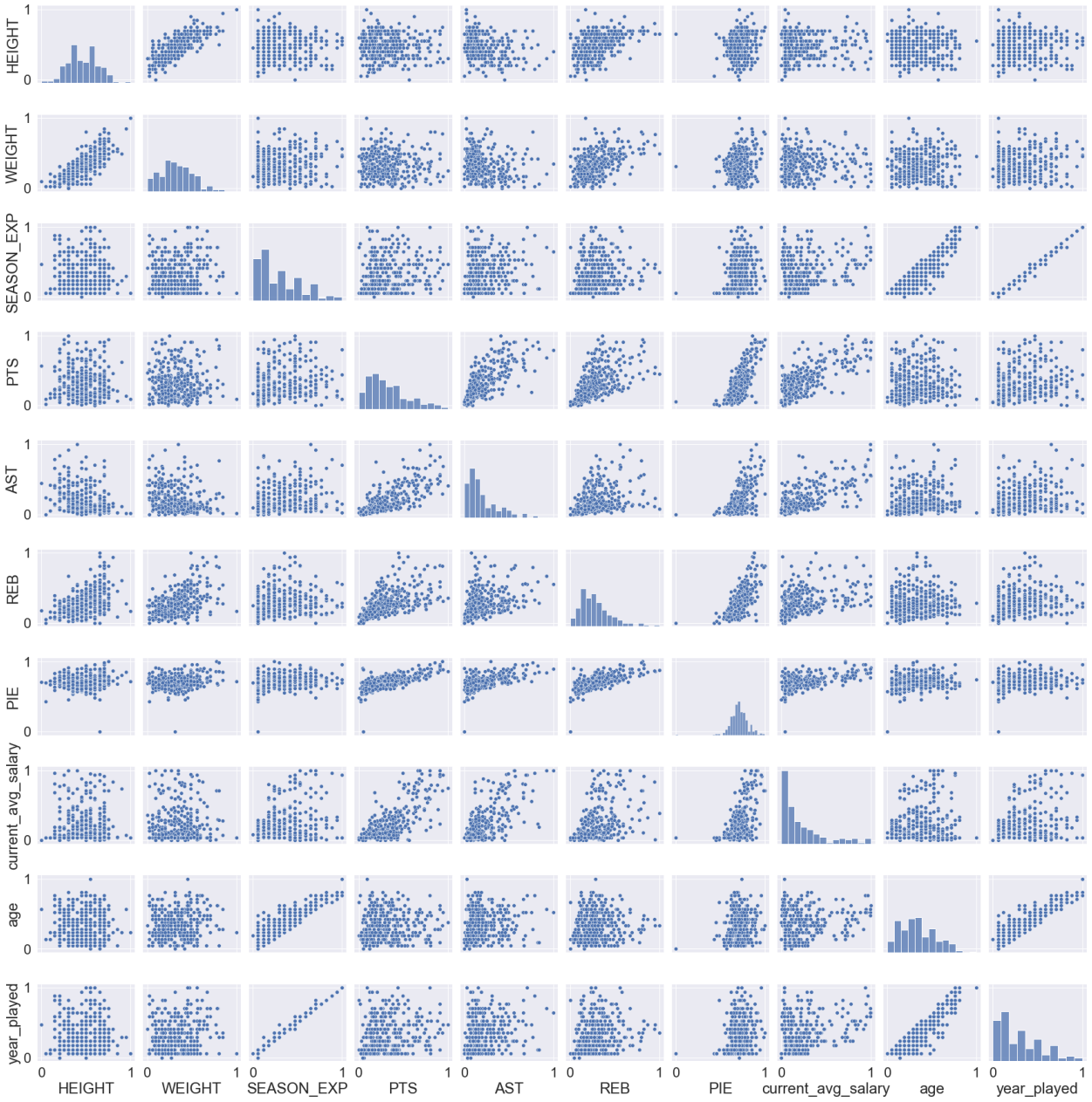
The third categorical value I explored is draft number. The NBA draft is an annual event dating back to 1974 in which the teams from the National Basketball Association (NBA) can draft players who are eligible and wish to join the league. These are typically college players, but international players are also eligible to be drafted. Typically, the player has draft number 1 is the best player in the draft.



In the graph, 0 mean un-draft. I found the first four draft number 1-4 have very high salary, but 5 is relatively low. And I found it is very interesting that draft number 17 and 27 are the highest average salary.

Pre-processing and Training Data Development

First step, since I am going to use regression models to analyze the data, I created dummy features for all the categorical values such as team name, position draft number and school using 'get_dummies'. There are too many schools and most of the schools have only 1 or 2 players. So, I only created dummy features for schools that have more than 6 players, and use 'Other' columns to represent other. Now, The total columns became 190. Second step, since the scales for numerical values are different and the values are all positive. I used 'MinMaxScaler' to standardize the magnitude of numeric features. After scaling the data, I made a pair plot of the data to check if there are any unwanted changes

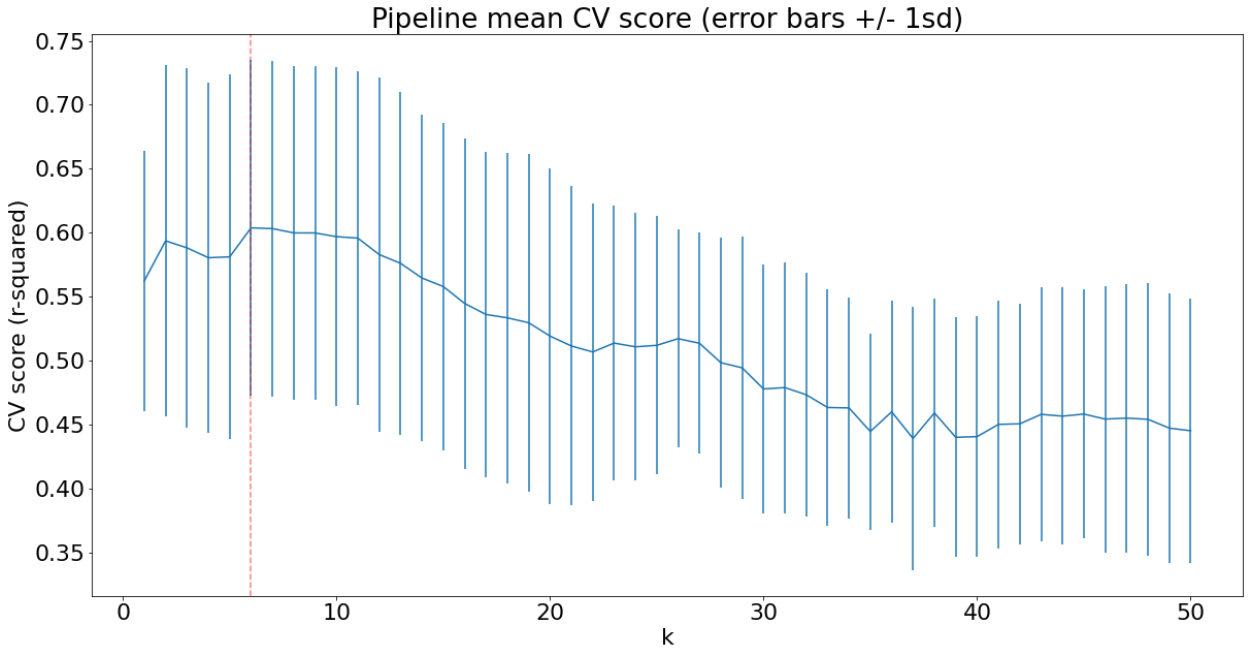


The plots look the same as before scaled. And then, I spared the data into training set, and test set. And I used 20% of the data for test.

Model Selection

For machine learning, the goal was to build models to predict player salary. Since salary is numerical value, I am going use regression models. So tested it with the three most popular machine learning regression models: Linear Regression, Random Forest Regressor and KNN regression. And the metric I am going to evaluate the models was mean absolute error because it is easier to interpreter the results.

For the linear Regression, I first fit it with all the features. But it turned out overfitting. The mean absoluter error for train set is 0.0077, but for the test set is 4.7e+19. The evaluation metric shows the model doing good to predict train data, but very poor on the test data. It is clearly overfitting. The model needs to reduce features. To do that, I created a pipeline including 'Seleckbest' as a step. Then, I used grid search cross validation to get the best K is 6 which means use the top 6 features to fit model will be the best.



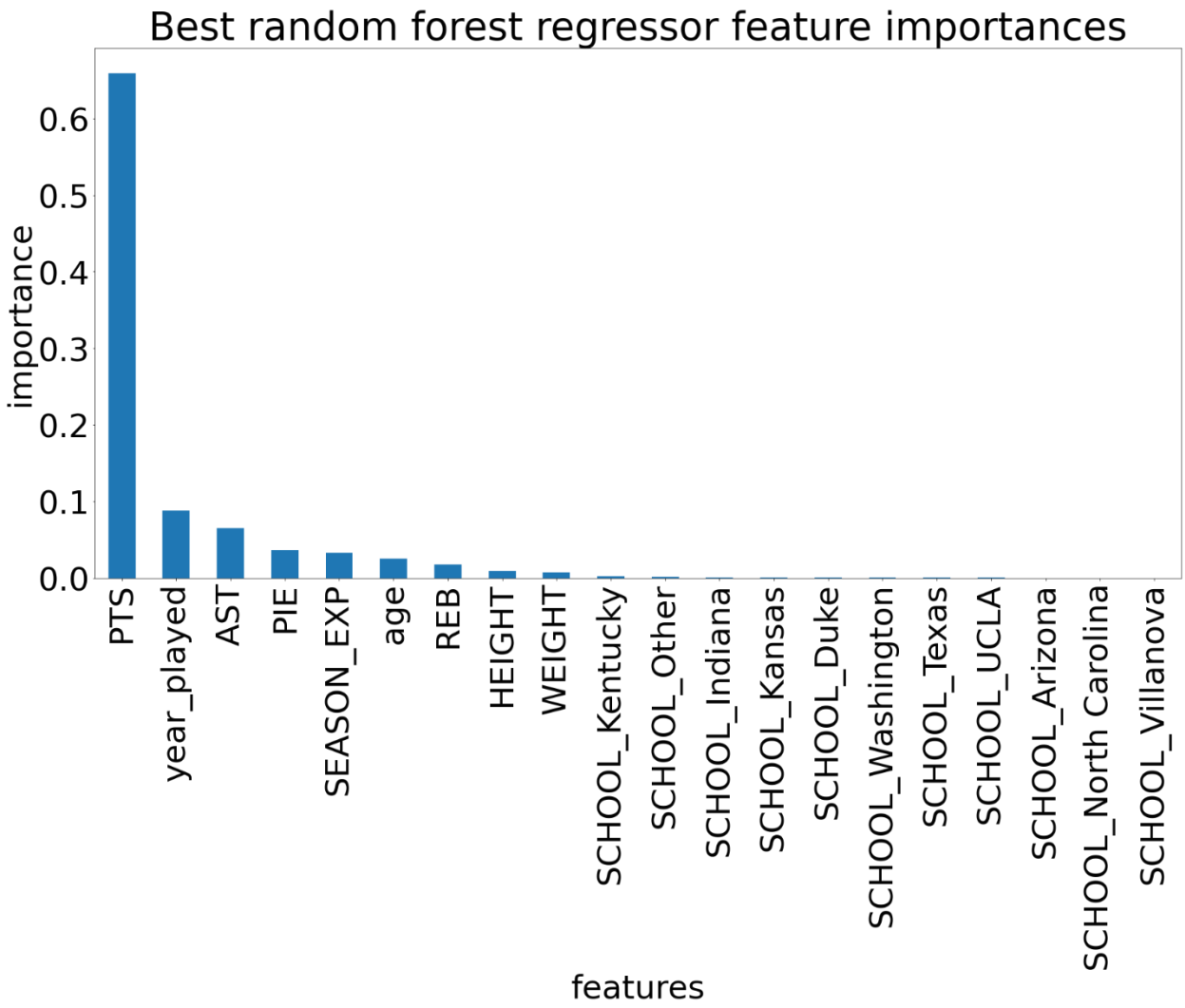
The above suggests a good value for k is 6, but there was decline with k, follow by a rapid increase with k. After the best k, there is a very slow decline. It is interesting that the variance of the results does not change much as k increased.

Which features were most useful? Step into best model. I got the following:

```
PTS              0.579280
AST              0.364178
year_played      0.204195
REB              0.058857
DRAFT_ROUND_1    0.024058
PIE             -0.108458
```

The results suggest that PTS (total number of points) is the biggest positive feature. Thus, makes intuitive sense and match what I found during data exploring. Player who has higher total number of points get pays more.

The second model I tested was Random Forest regression. The best params I found the model is max depth is 5 and n_estimators is 112.



Total number of points is the most importance feature same as what linear model suggest. However, the second most importance feature is year played instead of Assist to turnover ratio.

The last model I tested was the KNN regression model. The best parameter I found for the model is 13 neighbors and uniform weights.

Compared those three models, Random Forest regression model has the best performance. It has mean absolute error 0.0796 which scaled it back to salary is about 3.5 million. Given that the average salary is about 10 million, the error is about 35%. It is not the best model, but it does guide the direction. Its performance is 47% better than a random guess dummy model.

Model	Mean absolute error	Error in dollar
Linear regression	0.0978	4341291
Random forest regression	0.0796	3535549
KNN regression	0.1191	5287359
Dummy	0.1862	8265866

Extract Step

Since models did not output the best result, I thought it could the data I used to fit the model cause the problem. I think the outlier maybe the problem. Recall that outlier are the star players who have extremely high salary compared to their teammates. My hypothesis is that only star players' salaries are not only depends on their course performance and attribute but affected by the player's popularity and non-game issues.

To test my hypothesis, I drop the star players from the dataset. Since the salary is right skew distributed, I used 90% confidence interval to separate the data. Within 90% is non-star player, outside of 90% is star player. After dropping the 5%-star player, the data frame has total 317 rows, 36 players are dropped.

Then, I fit non star data into the model. However, the model outcome did not improve much.

With Non-Star Data

Model	Mean absolute error	Error in dollar
Linear regression	0.0898	3986755
Random forest regression	0.0730	3241510

The best model still random forest regression. It looks like the error became smaller from 3.5 million to 3.2 million, but average salary also decreased as the outlier are dropped to about 7 million. So, the error is about 46%. Therefore, the model did not become better, but worse. One reason could be the less data were used to fit the model.

Future Research

I think one of reason that the model performance is so good is because I do not have enough data. The data set I used to fit the model only have 352 rows which is not too much. In additional, the data set do not contain the player passed year salary. I think more salary information will help improve the model. Therefore, I would like to search for more data such as the last ten years salary of each player along with they are performance like PTS, AST, and REB for each year for the dataset it only contains the player' current salary and overall performance data.

In addition, I think also other factors that will affect player salary and did not include in the data set I am using. I did some more research after this project and noticed salary cap could be a factor that affect player salary. The NBA has salary cap that places a limit on the total money that teams can spend on salaries for their players. This limit is dependent on the NBA's total income as a league. The NBA salary cap is very flexible, however. It is termed a soft cap in that there are a number of exceptions that allow teams to go over the cap limit without penalty. I think that could a reason why the star play gets pay so much higher that non-star. And their salaries could be affected by how populated they are because that will determine how much they can bring to team.

Furthermore, I would like tested categorical model in the future. Which I will separate the salary into difference categorical such lower, below average, average, above average, high and super high.