Analysis of Peak Age and Effects of Aging on NBA Players

Rahul Chandra
Yunkyu Song

Summary of Research Questions and Results

1. In the first part of the project, we will examine at what age players typically become most efficient in the NBA, and "hit their prime." We would like to use factors such as height, weight, and position played to determine the different peaking age for different types of players. We will train a machine learning model using past player's age where their effectiveness was the highest, and use that to predict future player's peak ages that take into account the player's height, weight, and position when entering the league. Since effectiveness is very subjective, we will use Player Efficiency Rating will be used to determine efficiency of a player. Since there may be other things we may not be taking into account in terms of peak efficiency, we will also compute the average age of players who were selected as All-Stars, and the average age of players who were chosen as MVP's. This gives, without taking into consideration any factors, the average age of previous All-Stars and Most Valuable Players.
    a. Average age of All-Star: 26.5 Years
    b. Average age of MVP: 27.9 Years
    c. ML Model Mean Squared Error: Approximately 8(Varies on training/test data)

2. We determine how dramatically a player's performance declines after hitting their primes, by looking at performance 4 years after the player reaches their prime. We will be investigating Player Efficiency Rating(PER) and the factors that go into computing it to figure out the rate of decline of a player. Note that 4 years is not an arbitrary choice -- NBA contracts for high-caliber players often tend to be that length.
    a. ML Model Mean Squared Error: Approximately 32(Varies on training/test data)

Motivation and Background

The average NBA player salary is 6.4 million dollars, however, top NBA players get huge contracts, such as Russell Westbrook's 5-year, 206 million dollar contract with the Thunder, or Lebron James' 4 year, 153 million dollar deal with the L.A Lakers. Since teams have a limited budget for player salaries, it is crucial that they choose to offer these max-contracts to players who will provide them with great value through the duration of the contract. This was exemplified exceptionally well in Lebron James who became injured at a critical point during the season, resulting in 18 missed games. The previous season, he was able to play all 82, but at 34

years old, analysts have attributed attrition for his missed games. Similarly, since getting the contract, Russell Westbrook has lost in the first round of the playoffs every time. Our hope is that our model will help teams assess risk in signing these superstars, so teams will be able to look at drop off rates of players after reaching their primes, and see if they are truly worth the money. Note that although we speak of value here, we are simply talking about Player Efficiency Rating. Although there are other factors to consider when assessing the value a player brings to a team, we wanted a metric that measured value well. Thus, we say that the year a player has reached their prime is the year that their PER is the highest.

Dataset

The first dataset, found in https://www.kaggle.com/drgilermo/nba-players-stats, is scraped from basketball-reference.com. It has, starting from 1950, every single player's height, weight, year drafted, etc. It also has "season" stats, which has, for each player, in each season they played, all the statistics of their season. This includes points per game, minutes/games played, assists, turnovers, player efficiency rating, etc. This will be invaluable to us, as we will use these factors to determine the primes of players, as well as how well they are aging.

The second dataset is from https://www.kaggle.com/open-source-sports/mens-professional-basketball, and has, for each player, the awards they have won, including MVP's and All-Star Selections, which can help us get a general sense about when players "hit their prime," as we can investigate the average age players win these awards/honors.

Statistics recorded in season data: Year ,Player, Pos, Age, Tm, G, GS, MP, PER, TS%, 3PAr ,FTr, ORB%, DRB%, TRB%, AST%, STL%, BLK%, TOV%, USG%, blanl, OWS, DWS, WS, WS/48, blank2, OBPM, DBPM, BPM, VORP, FG, FGA, FG%, 3P, 3PA, 3P%, 2P, 2PA, 2P%, eFG%, FT, FTA, FT%, ORB, DRB, TRB, AST, STL, BLK, TOV, PF, PTS

Statistics recorded in player data: name, year_start, year_end, position, height, weight, birth_date, college

Statistics recorded in awards data: "playerID","award","year","lgID","note","pos"

*We will only use data on players from 1980, after the three-point line was introduced, to accurately compare players.

Methodology

First, we read and stored all the datasets. Joined the awards data with the player data to get the birth-date and award column in the same table. We subtracted the birth-date from the year of the

award for each player, to get the age of their selection. Second, we estimated the average age of an All-Star by computing the average age of the players in our dataset who received the award 'All-Star'. We did the same thing for the award 'Most Valuable Player.'

It is important to note that both these awards are somewhat arbitrary, as they are determined by votes rather than any single statistic. However, these computations provide us a good sanity check on when players reach their prime. For example, if we found that the average All-Star tends to be 27 years old, we know we (probably) messed up somewhere if in our actual calculations, we found that the average player's peak is 33.

Next, we trained a DecisionTreeRegressor model to figure out the peak age of an NBA player(in terms of Player Efficiency Rating) when given height, position, age, and weight when entering the league. To do this,we read and store the data again. We read the season statistics, and make sure to only add the seasons where a player plays over 15,000 minutes to the dataframe. This is because we don't want to choose seasons of a player where their PER is artificially inflated because they played few minutes. We joined with player data, and filtered after 1979, the three-point line was added in 1980, and we would like to compare NBA players consistently. We named this dataframe player_data. We added a peak_age column and a peak_per column to player_data. Then, we created a DecisionTreeRegressor. We filtered player_data into the input columns(height, position, age, and weight when entering the league). We called this X, and similarly, we filtered player_data into results(peak_age) and called it Y. Split X and Y into a training set and a test set in a 80:20 ratio. We fit the training data and training results to the DecisionTreeRegressor, and inputted the test data into the model. Once we received the results, we calculated the mean-squared error between our predicted peak age and the actual peak age.

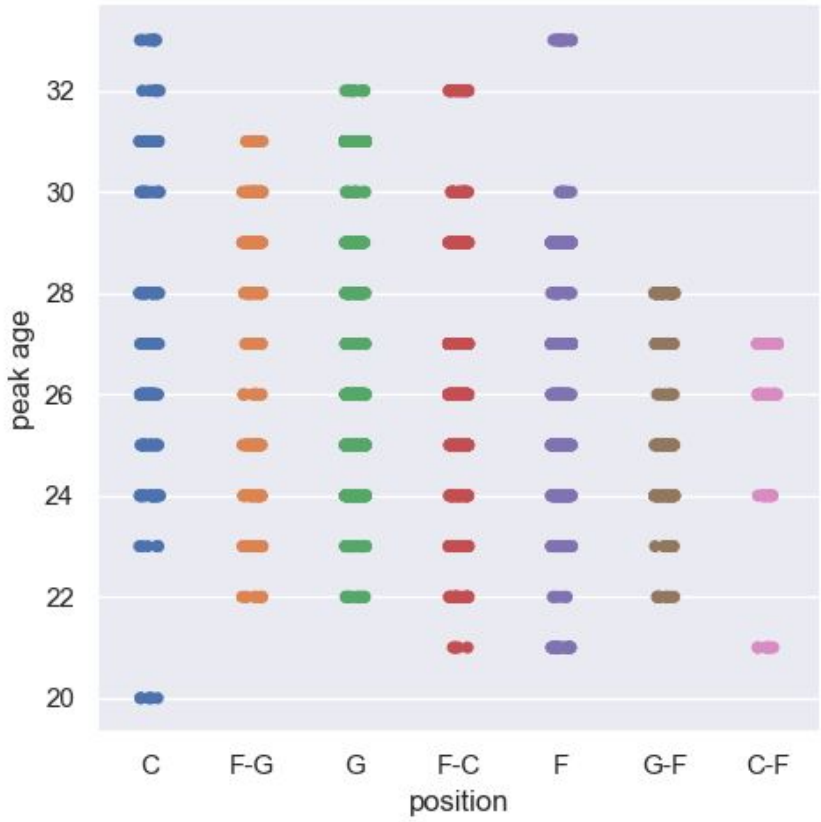
Then, we plan to again use the DecisionTreeRegressor to find the player's decline. Again, we used the player_data dataframe. We add columns that have PERs 1 year after the peak age, 2 years after the peak age, 3 years after the peak age, and 4 years af ther the peak age. We look at the post 4 years of the peak age, since it is the typical contract period for high-performing players. Using those estimated peak ages, we construct another column that estimates the player's decline in performance using this formula: 4 * peak PER - (
peak PER'] / 2 + peak_1 PER'] + peak_2 PER + peak_3 PER + peak_4 PER / 2)

We decided to use this measure, rather than the average decline, to account for the cases in which players play consistently well for a while, and suddenly decline, which is better than suddenly decline, and play consistently for the rest of the career. Using the DecisionTreeRegressor, we again follow the steps outlined above when predicting peak age to filter the dataframe into an input and output columns, this time with attributes 'height', 'weight', 'position', 'beginning_age', 'peak age', 'peak PER', 'decline measure', 'USG%', 'FG%', 'FG', '3P',

'AST', 'STL', 'BLK', 'VORP', '3PA','WS', 'BPM,' all of which are from the player's peak year. The output is the column 'player decline.' All these statistics are from the player's peak year, as realized that the statistics from a player's peak year will shed insight into how they will decline. Then we compute the mean squared error from our predicted values and the actual values when we pass in the inputs from the test set.

Dataset

Our goal was to find the relationship between position, height, weight, and starting age of an NBA player to the age they will be the most effective at, and make a machine learning model to predict future players. Before we did that, we found the average age of an MVP and an All-Star, which was 27.9 years old and 26.5 years old respectively. This shows that the peak-age of most players is fairly young, and the graphs we made that plot height/weight on the x-axis and the peak-age on the y-axis mostly reflect that. What was interesting was the position vs peak-age

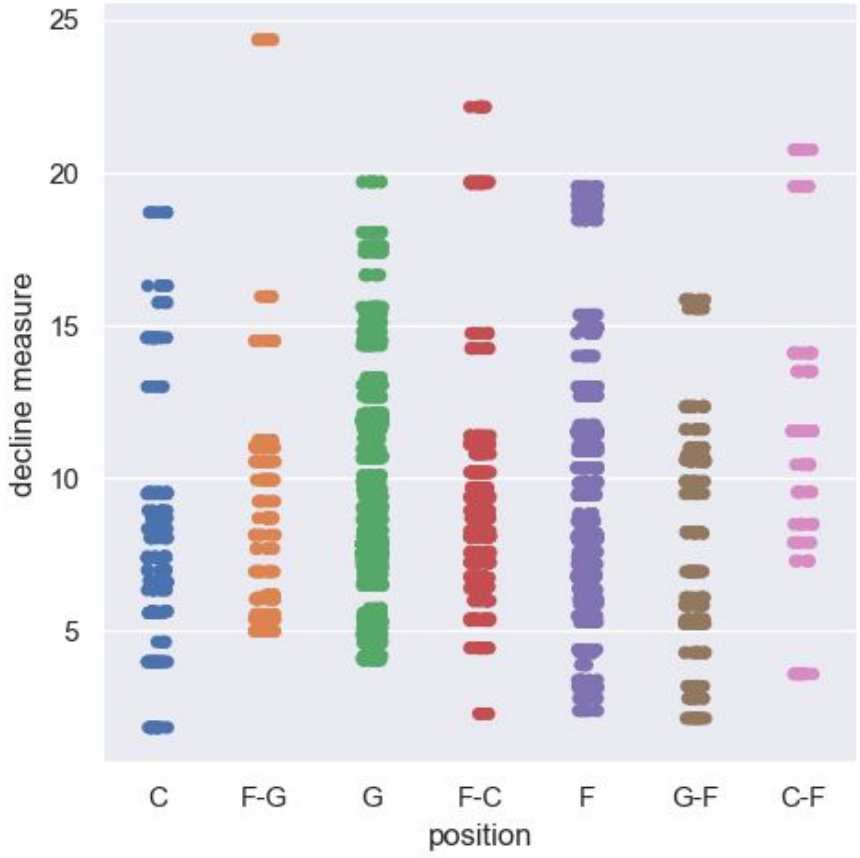


graph.

Here, we can see the range of ages in which players peak in relation to the position they play. This is very interesting, as we see that centers(C) have a very large range of peaking values, which makes sense, as centers tend to be the largest so injuries could make them peak early, but their large stature might keep them relevant enough to peak late. However, guards(G) seem to

peak later in their careers, which makes sense, as they are the skill positions. When we trained our machine learning model on the training dataset with the position, height, weight and age when starting in the NBA, our mean-squared error on the test dataset is around 8 to 9. This means, given height, weight, age, and position when entering the league, the model can give an age that is accurate +-3 years. This is obviously not great, so in the future, we would like to input player's college statistics or high school statistics as well, and see if our model performs better. It is possible that there might not be a large enough correlation between our factors and the peak of a player, as development may also play a large role, which we can't measure with the data we have.

Our machine learning model predicted decline measure better than it predicts peak age. The mean squared error for decline measure is usually under 36, which means that our model is off by +- 6 in the decline measure according to the way we calculated decline measure. Although we have tried many different columns to see if that would give us a better MSE, it seems like there simply is not enough correlation between the statistics of a player's peak year and the decline they will experience the next year. However, when we plotted decline measure vs position, this is the graph we received. It is important to note that the higher the decline measure, the worse it is for the team, as it means that the player had a big drop off following their peak year. According to the graph, it seems like guards(G) and and F(Forwards) tend to have the least drop off.



Reproducing Results

To reproduce the results, simply make sure you have the entire folder with all the datasets. Then, simply run the program. It should print 6 things: Average age for All-Star Selection, Average age for Most Valuable Player, Training MSE for Peak Age, Test MSE for Peak Age, Train MSE for Decline Measure, and Test MSE for Decline Measure. It should also create 9 graphs: position vs PER, weight vs PER, height vs PER, position vs peak age, weight vs peak age, height vs peak age, position vs decline measure, height vs decline measure, and weight vs decline measure.

Working Plan Analysis

Clean Data and Make Initial Computations | Predicted : 2 Hours| Actual: 2

- Convert the datafiles in csv form to the dataframe.

- Join two dataframes with the players' name.
- Compute the average MVP/All-Star Selection age.
- Split the data into the training set and test set as described above.

- Report findings with visual aids.

Make Models to Predict Peak Age | Predicted: 10 Hours| Actual: 12(More Data Cleaning)

- Use a regression model to find a model that predicts players' peaking age with different responsive variables.

- Repeat with different hyperparameters.
- Do a sanity check -- does the peaking age somewhat matchup with the computation we performed first? Are there any outliers? For example, a position where the peaking age is significantly older than other positions?

Compute Decline For Different Players | Predicted: 15 Hours| Actual: 15

- For each player, find the peaking age using the model found previously, and add P_1, ..., P_4 (described above) columns to the dataframe, and add another column named decline measure using the formula mentioned above.
- Use a regression model to find a model that predicts decline measure.
- For the model, store the value for the Mean-Squared-Error.
- Replicate the process with different hyperparameter settings.

Prepare Report | Predicted: 10 Hours | Actual: 12(More Testing)

- Analyze the results. Is there a certain type of player that declines slower than other players?

- Create graphs to demonstrate different decline rates for different players.

Our predictions for how many hours we would spend were decently accurate, but we didn't really expect our model to not work very well, so we didn't expect there to be so much testing. So that added a few hours to the project. Also, we didn't realize how much of the work would just be cleaning data. We expected that actually making the models would be the tough part, and that may have been true if we had chosen to do this using TensorFlow or PyTorch, but we chose to use scikit learn, and the majority of the work was cleaning data. For example, we thought we were completely done with the project, and then we realized that PERs sometimes might be artificially inflated if the player plays few minutes, so we had to do more dataframe manipulations to get the correct data. So our estimates were good, but the distribution of the work we expected did not follow the plan.

Testing

For the machine learning models, we computed the mean-squared error, which tells us how accurate our model is on the test set. We tested how well our model performs with different inputs numerous times. For example, in predicting the peak age, we also added statistics from the first year, and see if the model becomes more accurate, but it does not. In fact, the model becomes worse when we include any part of the first-year statistics. That is how we determined that there might be very little correlation between the first-year and players peak-age. For the model to predict decline measure, we again used the mean-squared error, and tested many different statistics of the peak-year. Again, there was no significant difference, but our graphs that plot position vs decline measure show that skilled position have a lower drop off. This would make sense, since skill “ages” well, rather than athleticism, so we believe the way we computed decline measure is correct.

Live Video or Presentation

We would like to do a live-video presentation.

Collaboration

We did not collaborate with anybody.