27 April 2022

Superteam

A basketball analytics framework by Marc Sperzel

https://github.com/marcoloco23/super_team

Traditional basketball analytics methods focus mainly on understanding the strengths and weaknesses of individual players and then aggregating these to understand the strengths and weaknesses of teams. The central assumption of this project is that the most valuable information in determining who will win or lose a game lies in the interaction of individual players on a given team. As a result, any attempt to understand why one team is better than another will require knowledge of this complex interaction. Moreover, any team's performance will also depend on the opponent's team performance and thus the interaction between the two teams. Therefore, the goal of this project was to understand and predict these interactions based on the combined performances of individual players.

Data

To achieve this, I scraped over 10000 games worth of data dating back to 2011 from NBA.com using the nba_api python library (https://github.com/swar/nba_api). For each game, 105 individual player statistics are collected for each player, as well as each team's plus-minus result indicating how much the team won or lost. This data is then stored in an internal database (Mongo DB), allowing fast fetching and easy manipulation of the data.

The training data is then created by stacking the individual player performances of both teams for a particular game. As a result, a fixed team size must be specified to guarantee a constant number of training data columns. To ensure consistency, the player performances are sorted by minutes played such that the player's statistics with the most minutes played will always represent the first 105 columns of training data. A team size of 13, therefore, represents training data with 13 (players) x 105 (features) x 2 (teams) = 2730 columns, where the first 1365 columns represent the home teams features and the second half describes the away team's features. The plus-minus score is then always given with respect to the home team. This way, we can double the size of our dataset by using each game twice and switching the order of the two groups as well as the sign of the plus-minus score.

Model

A regression model is then trained to predict a team's plus-minus score based on the combined player performances of both teams. The model achieves remarkable accuracy of 97% in predicting which of the two teams won the game given the individual player statistics and can predict by how much a team won or lost down to a 2-point differential. As the number of players on each team is decreased, the model's error increases.

As the model requires individual player performances as input, we calculate the average performance statistics of individual players over a given amount of time. We can then deploy the model to build all kinds of practical applications using this.

27 April 2022

Applications

The most basic use case would simply be to match up two existing NBA teams. We're assuming all players are fit to play without any injuries in this scenario. We can introduce injuries by changing the minutes any player is expected to play or removing him entirely from the lineup. We are not limited in selecting existing NBA teams but can build imaginary teams of players and let them play each other.

We can then simulate the regular-season standings by letting each NBA team play against every other NBA team and recording the win-loss statistics. Averaging over this season's player performances, the win-loss results show that Boston is the best team in the NBA, winning against all other teams. At the same time, Utah and Pheonix are their most substantial competition this year. On the other end, Detroit and Orlando have the lowest win-to-loss ratios, with Detriot unable to win against any team. By comparing this with the actual standings of the current NBA season, we see a clear correlation.

The use of this season's average player performances should intuitively reflect this season's team standings, and thus the previous simulation may seem trivial. For example, a player on a winning team such as the Boston Celtics will have a higher individual average plus-minus score than a player playing for the Detroit Pistons. Instead, we can average player statistics over multiple seasons dating back to 2011. Simulating the season with this new player data, the Golden State Warriors are predicted to come out on top, followed by Utah and Miami, while Boston only ranks 6th.

Having simulated the regular season and ranked all existing NBA teams, let's simulate the 2022 playoffs. Only using player performance data from the 2022 regular season and a team size of eight, the Boston Celtics are predicted to be the 2022 NBA Champions. The bracket below shows the individual matchups.

Figure 1: Predicted playoff bracket using 2022 regular-season player performance data and a team size of 8 and assuming no injured players.

27 April 2022

The playoff elimination tournament is set up to identify the strongest team in the league. Using the same elimination tournament approach, let's turn our attention to methods of building great teams that don't (yet) exist.

Tournament

To find great teams, an identical tournament is set up where random teams are drafted and matched up against each other until a winner emerges. The winner is then allowed to compete in the next iteration of the tournament. This way, strong teams are naturally selected as they will persist over many tournament iterations (like how the Bulls or Warriors were able to claim back-to-back championships). Running an example tournament for 100 iterations using average player performances since 2011 and a team size of eight, we find a super team consisting of:

Donovan Mitchell, Chris Paul, Blake Griffin, Scottie Barnes, Jeff Green, Michael Porter Jr., Immanuel Quickley, and Isaiah Roby,

winning 97 back-to-back championships. Compared with some existing NBA teams, this team may not even be that unrealistic to assemble. We can then test this team against current NBA teams and find that it beats all of them.

To take this one step further, we could introduce a salary cap by getting all individual players' expected salaries and only allowing teams which fulfill some salary cap conditions. Instead of scraping salary information data from the web, I decided to build my own player scoring systems which rank players according to their performance statistics. A player's value is then determined by his rank, while a salary cap would translate to a maximum total score for players on a team. Moreover, this scoring has true predictive power in predicting the season's MVP.

MVP Ranking

This ranking works by taking the average player performances for a given season, standardizing each feature, and calculating each feature's correlation with the average plus-minus. Each feature is then multiplied by its plus-minus correlation, and the average across all features is calculated. This score is then squared to account for a player's exponential value increase with skill. Following this procedure for the 21/22 NBA season, we find the following top 5 candidates:

1. Nikola Jokic
2. Giannis Antetokounmpo
3. Luka Doncic
4. Jayson Tatum
5. Joel Embiid

We can back-test this by predicting past season MVPs and find 4/6 accurate predictions.

Season	Actual MVP	Predicted MVP
20/21	Nikola Jokic	Nikola Jokic
19/20	Giannis Antetokounmpo	Giannis Antetokounmpo
18/19	Giannis Antetokounmpo	James Harden
17/18	James Harden	James Harden
16/17	Russell Westbrook	James Harden
15/16	Stephen Curry	Stephen Curry

In the two cases where our ranking did not predict the MVP, the real MVP finished second. We can also do this analysis by combining performances dating back to 2011 and find the following ranking:

1. LeBron James
2. Stephen Curry
3. Kevin Durant
4. James Harden
5. Luka Doncic

indicating that LeBron James is the best player of his time.

Looking at current NBA teams, we can calculate the average cumulative score of each team. This season's team's average score for 13 players is 7.78, with the highest-scoring teams being Golden State (9.0), Memphis (8.9), and Miami (8.7). For eight players, the average summed score is 5.9. Of course, this is substantially different from the actual salary cap rankings, but it gives a good indication of what a team should be worth if players were paid by their performances. The team identified earlier has a combined score of 5.6, which is less than the league average for eight players of 5.9 and, therefore, would be an allowed team in the NBA.

Super Team

Another, and perhaps more powerful way to search for super teams, would be to start with a random team, find another team that beats that team, and then iterate, always keeping the team that won last until any randomly selected team cannot beat it. Setting the number of iterations to 10000 and running the program without a salary cap, we find the following team of 8 based on historical performances dating back to 2011:

Kevin Durant, Stephen Curry, Jimmy Butler, Nikola Jokic, Jaylen Brown, Myles Turner, Andre Iguodala, Jae Crowder.

This team's value score is 7.1 and thus clearly exceeds our cap of 5.9, making it unreasonably strong. Let's run the same program with a value cap. This time we get the following team with a value score of 5.1:

27 April 2022

Stephen Curry, Mikal Bridges, Victor Oladipo, Nikola Jokic, Draymond Green, Monte Morris, Montrezl Harrell, Kai Jones.

Trade Finder

Next, let's build a trade finder that compares a team's performance to its performance if it makes a trade. By iteratively doing this and keeping track of the win-loss statistics for each potential trade, the best trade is found by maximizing the win-loss ratio. Again, we need to calculate an approximate trade value for each player, as some players are worth more than others, and trades need to remain fair for both teams. We will only allow players to be traded if they are within a particular threshold difference in value score.

Let's look at 100 possible trades for the Charlotte Hornets, match them against all other NBA teams, and record their win-loss statistics. Our trade finder suggests that the hornets trade Cody Martin for P.J. Tucker to improve from 0.53% to 0.57%-win percentage on the season. Notice that the trade finder does not yet consider a player's development but only looks at his average performance. However, we can easily track this development by treating players' past statistics as a time series rather than averaging them. As of now, this feature has not been developed.

Building a Team around a Player

Finally, let's build a team around a given player, as this is another common theme in the NBA. As the current predicted MVP is Nikola Jokic, we will build a championship team around him. Like how we determine a super team in the prior section, we will do the same here, only keeping Jokic constant in all teams. Using all-time player performance data, after 10000 iterations with a salary cap imposed, we find the following team:

Devin Booker, Khris Middleton, Nikola Jokic, Ben McLemore, Armoni Brooks, Jericho Sims, Usman Garuba, DJ Stewart.

Testing this team against all NBA teams, we find that it has a win rate of 93%.. Doing the same for LaMelo Ball, we find another team:

DeMar DeRozan, LaMelo Ball, Andre Iguodala, Danny Green, Seth Curry, Alec Burks, Donte DiVincenzo, Oshae Brissett,

with a win-loss percentage of 90%. Notice that this team has no player at the center position but multiple point and shooting guards. Our model does not have these restrictions, and neither does the application I built. We could, of course, artificially require each team to have at least one or two players at each position, but this may limit the predictive power of the model.

There are countless other possible applications to build using this framework, each providing different and unique insights into the game of basketball.

27 April 2022

Limitations

Looking at the feature importance of our model, it is evident that the model relies mainly on the plus-minus scores of individual players. This is a limitation of the model as it would favor players currently playing on winning teams while unfairly penalizing players playing on losing teams. Nevertheless, keeping the individual plus-minus scores of players as features makes sense as they provide valuable information about a player's performance. Aggregating player statistics over multiple seasons should neutralize this bias. Moreover, the model does not provide any indication of why a team may be better than another team.

Future Work

I plan to build a user interface allowing users to interact with and explore the applications described in this paper in future work. There is also the potential of building applications specifically targeted toward Fantasy basketball, allowing users to create custom teams and get advanced insights into their teams. I have only considered NBA games so far, but there is no reason other than simplicity for this restriction. In the future, I plan to incorporate college basketball players and WNBA players. This will allow building applications useful for the NBA draft in which teams have to pick from a selection of players, often guessing at how a given player will fit into a team. To convince people of its power in the real world, it is also essential to understand why some teams are better than others. Applications to demonstrate this are equally important as the predictions themselves.