Superteam

https://github.com/marcoloco23/super_team

What makes one basketball team better than another? Why do some players play better together than others? It all comes down to which group of players scores the most points, concedes the least, and does so more consistently than anybody else. 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 whether a team will win or lose a game lies in the interactions between players. Moreover, any team's performance will depend on the opponent's performance and the interaction between the two teams. This project aims to build a simple model to capture these interactions and predict the winning team given the box score containing the individual players' statistics.

To achieve this, I scraped over 10000 games worth of data dating back to 2011 using the nba_api python library (https://github.com/swar/nba_api). For each game, 105 different statistics are collected for each player, in addition to each team's plus-minus result, which indicates how much the team won or lost by. The goal is to build a model that maps all these player statistics onto a single plus-minus result indicating which set of players will win the game.

After trying multiple approaches, the best way was to simply stack the individual player statistics of both teams for a particular game into a single row. As a result, a fixed team size must be specified to guarantee a constant number of features. To ensure consistency, the player performances are sorted by minutes played such that the player with the most minutes played will always represent the first 105 columns of the training data. A team size of 13 players, therefore, represents training data with 13 (players) x 105 (statistics) x 2 (teams) = 2730 columns, where the first 1365 columns represent the home teams features and the second half represents the away team's features. Using a standard unsupervised dimensionality reduction technique known as principal component analysis (PCA), these 2730 columns are reduced to just 100 features that capture the most essential information. The plus-minus score is then always given with respect to the home team.

An xgboost 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 an accuracy of 97% in predicting which of the two teams won the game and can predict how much a team won or lost by, down to a 2-point differential. As the number of players on each team is decreased, the information contained in the input is reduced, and the model's error increases. Essentially, the model learns to predict the outcome of a game by simply looking at its box score of player statistics.

As the model requires individual player performances as input, the average performance statistics of individual players over a given amount of time are calculated. The assumption is that a player will continue to perform according to his average performance even as he switches teams. The model can then be used to build all kinds of practical applications.

The most basic use case would simply be to match up two existing NBA teams and see who comes out on top. Let's simulate the regular season by allowing each NBA team to play against every other NBA team and recording the win-loss statistics. Using the averaging player performances for the 2022 season, 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 seem to be 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. Comparing this with the actual standings of the current NBA season, there is a clear correlation.

The use of this season's average player performances should intuitively reflect this season's team standings. 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, and this will, in turn, result in the Boston Celtics winning against Detroit. Instead, the player statistics can be averaged 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 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 expected 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.

Using the same playoff elimination tournament approach, let's turn our attention to methods of building strong teams that don't yet exist. One way would be to randomly draft teams and match them up against each other in an elimination tournament until a winner emerges. The winning team is allowed to compete in the next tournament iteration, naturally selecting teams that persist over many tournaments. Running an example tournament for 100 iterations using average player performances since 2011 and a team size of eight, we find a 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. Testing this team against current NBA teams, the model predicts that it beats all of them. However, some of the suggested teams are unrealistic and would be impossible to assemble.

We could introduce a salary cap by getting all individual players' expected salaries and only allowing teams that fulfill the salary cap conditions. Instead of scraping salary information data from the web, I decided to build a player ranking metric. 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.

This ranking works by taking the average player performances for a given season, standardizing each feature across all players, and calculating each feature's correlation with the average plus-minus. Each feature is then multiplied by this plus-minus correlation, and the average across all features is calculated. Following this procedure for the 2022 NBA season, we find the following top 5 ranked players:

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

We can back-test this scoring method by predicting past season MVPs and find four out of six 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. Doing this analysis by combining performances dating back to 2011, we 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 now calculate the average cumulative score of each team. This season's average score for a team of 13 players is 7.8, 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 and, therefore, would be an allowed team in the NBA.

Another, and perhaps more powerful way to search for 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:

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

Next, let's build a trade finder that compares a team's current performance to its performance after making 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.

Finally, let's build a team around a given player, as this is another common theme in the NBA. The current predicted MVP is Nikola Jokic, so we'll try to create a championship team around him. In the same way 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 and imposing a salary cap, we find the following team after 10000 iterations:

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, nor does the application built. One could, of course, artificially require each team to have at least one or two players at each position, but this may limit the model's predictive power and introduce unnecessary complexity. There are countless other possible applications to build using this framework, each providing different and unique insights into the game of basketball.

A limitation of the model is that it does not indicate why a team may be better than another team, but it simply gives you a plus-minus score. Looking at the feature importance of our model, it is evident that it relies in large part on the plus-minus scores of individual players. This may be 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. Another model was created without the players' plus-minus scores to test this limitation. The model achieved nearly identical performance, indicating robustness against this plus-minus bias. However, understanding why one set of players is better than another requires further analysis, which has not been studied so far.

So far, I have only considered NBA games, but there is no reason other than simplicity for this restriction. In the future, I plan to add college basketball, WNBA, and BIG3 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 an existing team. Moreover, because we can vary the team size of our model, we can specifically build a model for the BIG3 league in which the team size is restricted to three players.

To convince people of the model's usefulness, it's essential to understand why some teams are better than others. As is typical with modern machine learning models, they lack explainability. By building applications using the model as I have outlined in this article, I believe we can better understand the model and make its predictions more transparent. The question now is whether we can build the tools to get the model to a point where coaches and general managers can trust it, learn from it, and use it to make better decisions. At the very least, it can be used to earn some money by making informed betting decisions.