Machine Learning and the NBA All-Star Game

By Kian Kaas & Harry Nguyen

The Problem

Every year the NBA (National Basketball Association) hosts the All-Star game, where 24 of the league's best players are featured, and every year fans are wondering "Who will be an all-star this year?". In this project we hope to utilize machine learning techniques to answer this question and find out what it takes, statistically, to become an NBA All-Star.

The Data

The data we used for this project was fetched from a publicly available API library. This library provided high-level functions that allowed us to gather large amounts of data from the NBA's official statistics website. Specifically, we gathered game logs, player regular season statistics, and team statistics from 1996 to 2023. Furthermore, the data we gathered is all prior to the respective season's all-star game, as using data after the all-star game would not be entirely suitable data for this project topic.

The first thing we did was make API calls to gather team and player stats for each year. Luckily, we were able to get an entire season's worth of stats using a single API call and specify that games after the all-star break were not to be included, so this step was relatively quick.

However, an issue arose when we wanted to gather additional advanced stats such as usage and true shooting percentage, as these stats were only available for individual games. These extra stats were not only important as they could be used in our eventual machine learning model, but that they were also necessary to calculate other advanced statistics that are not available from our API.

To address this, we were required to gather box scores for every game from 1996 to 2023, and then perform a group-by and aggregate to retrieve advanced stats for players, teams, and the league for each season. Some stats that we gathered from this step include: true shooting percentage, usage percentage, PIE (player impact estimate), pace, number of possessions, and offensive rebound percentage.

With this data, additional advanced statistics were calculated. That is, statistics that go beyond basic stats such as points or minutes played, and provide additional insight into a player's impact. To do so, we created a step in our pipeline to calculate any advanced statistics that we felt were valuable such as PER (player efficiency rating) and WS (win shares).

PER is, as described on basketball-reference.com, “a per minute rating of a player’s performance” that attempts to collect all of a player’s contributions into one number. On the other hand, WS is a statistic used to measure how many wins a player contributes to their team, and takes into account player, team, and league-wide stats.

Lastly, we had to generate labels for our model’s training data. In this case that would be whether a player was selected as an All-Star for a particular season. Luckily someone had solved this problem before us, so we chose to adapt their solution to our project to save us some time and effort. There were some things that we changed, such as better variable naming, code organization, and to output the results as a csv file instead of a pickle file. In essence, this step scrapes NBA All-Star rosters from basketball-reference.com and outputs a csv with 2 columns; player name, and All-Star year. However, we had to be cautious with how we scraped the data, as making more than 30 requests within a minute causes the site to block your session for an hour. To solve this, we simply made the program sleep for 30 seconds in between every year.

In terms of data cleaning there were some things that needed to be done. For instance, there were many fields that were irrelevant to the question we were asking. Some of these fields included player nicknames, number of personal fouls, and stat rankings, such as whether a player is a top 5 rebounder in the league. Stat rankings are likely not important, as we already have the total number of stats, like points or rebounds, for each player and our model can likely infer the rankings on its own.

Machine Learning Techniques

After collecting and cleaning all of our data, our next step is to pick a machine learning model that will accurately predict the NBA All-Star players for the 2023 season. The NBA doesn’t have an official way of calculating, ranking, and selecting all-stars through statistics, but rather uses a voting system. The NBA claims that fans make up 50% of the vote, and NBA players and media each comprised 25% of the vote. Nonetheless, we will do our best to provide an accurate prediction through our model using the data we’ve collected.

We decided to use a Random Forest Classifier model to predict our all-stars. We figured this would be a suitable approach as Random Forest is an ensemble learning method that combines multiple decision trees, thus reducing overfitting and enhancing the generalization ability of the model, making it suitable for the convoluted nature of NBA player statistics. Additionally, an NBA player’s individual performance often involves non-linear relationships between metrics, and Random Forest does a good job at capturing these complexities accurately. Another thing Random Forest does well is handling imbalanced classes, which is relevant as all-star selections are only a small sample compared to the total player pool. Furthermore, given the wide range of features in our collected data, Random Forest efficiently scales to handle the complexity. Although our Random Forest model will hopefully give us an accurate prediction of NBA all-stars, we must remember that predicting all-stars is a difficult task, as player success can be heavily influenced by various factors beyond statistical metrics, such as popularity throughout the league.

When training our datasets, we used a simple 80/20 split between the training and testing data. As we are trying to predict all-stars for the 2023 season, we used only player statistics from 1996-2022 in our training data. After we fit the training data to our Random Forest Classifier model, we used only the 2023 season player statistics to predict our all-stars for the 2023 season. From there, we did some simple data manipulation and were able to compare our results and see how well our model performed.

Findings

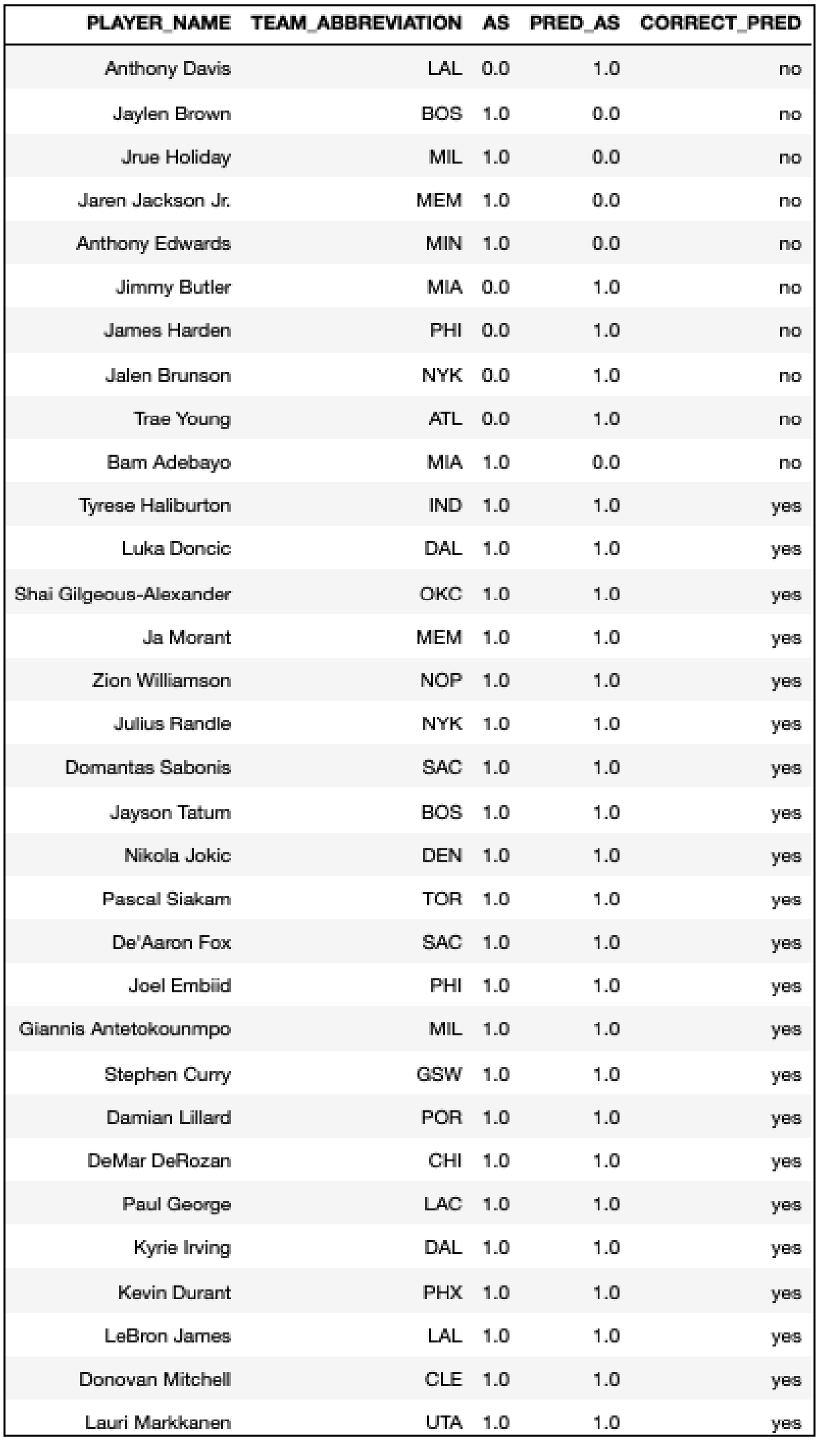
So, how well did our model perform? *Figure 1* below compares which all-stars were correctly predicted, which all-stars were incorrectly predicted, and which all-stars our model missed.

As we can see in *Figure 1* below, our model correctly predicted 22/27 all stars, resulting in an accuracy score of 81.5%. We cannot expect our model to be entirely accurate as there are many other factors contributing to a player's all-star status aside from statistics. For instance, fan votes play a factor, which may lead to popular, but non-impactful players making the all-star team. This was evident when Kobe Bryant made the all-star team in his final year despite being his worst year statistically.

Our model incorrectly predicted 5 players to be all-stars who were not selected for the 2023 all-star game. The most notable player would be James Harden, who has been a very well respected and high level player in recent years, making the previous 10 all-star selections before the 2023 season. Not to mention he even led the league in assists in 2023, thus being considered by many NBA fans as an "all-star snub". The other 4 players, Anthony Davis, Jimmy Butler, Jalen Brunson, and Trae Young, all have a strong argument for being all-star snubs as well, as they have all previously made all-star appearances bar Jalen Brunson. Thus, we believe our model did a good job in predicting all stars.

On the other hand, there were 5 all-star players that our model did not predict to be all stars. We suspect that 3 of the players were not predicted by our model due to the lack of defensive statistics in our data, such as defensive rating and defensive win shares. These players were Jaren Jackson Jr., Bam Adebayo, and Jrue Holiday. Jaren Jackson Jr. won the 2023 Defensive Player of the Year award, while Bam Adebayo and Jrue Holiday both finished in the top 10 of voting for the award. As for the other 2 players, Anthony Edwards and Jaylen Brown, we suspect our model may have missed them due to the popularity of the players off the court. Anthony Edwards is a young player who is exciting to watch, and has gained a large fan base in the younger generation over recent years. The same can be said about Jaylen Brown, who plays for the Boston Celtics, a team with one of the largest fan bases in the NBA.
Another feature we could look to add into our model would be team conference rankings, seeing as how player's like Jaren Jackson Jr. and Jaylen Brown both play on teams ranking top 2 in their

respective conferences. If a player is individually performing well on a successful team it is likely indicative that they are contributing meaningfully to their team's success, and are thus an impactful player.



PLAYER_NAME	TEAM_ABBREVIATION	AS	PRED_AS	CORRECT_PRED
Anthony Davis	LAL	0.0	1.0	no
Jaylen Brown	BOS	1.0	0.0	no
Jrue Holiday	MIL	1.0	0.0	no
Jaren Jackson Jr.	MEM	1.0	0.0	no
Anthony Edwards	MIN	1.0	0.0	no
Jimmy Butler	MIA	0.0	1.0	no
James Harden	PHI	0.0	1.0	no
Jalen Brunson	NYK	0.0	1.0	no
Trae Young	ATL	0.0	1.0	no
Bam Adebayo	MIA	1.0	0.0	no
Tyrese Haliburton	IND	1.0	1.0	yes
Luka Doncic	DAL	1.0	1.0	yes
Shai Gilgeous-Alexander	OKC	1.0	1.0	yes
Ja Morant	MEM	1.0	1.0	yes
Zion Williamson	NOP	1.0	1.0	yes
Julius Randle	NYK	1.0	1.0	yes
Domantas Sabonis	SAC	1.0	1.0	yes
Jayson Tatum	BOS	1.0	1.0	yes
Nikola Jokic	DEN	1.0	1.0	yes
Pascal Siakam	TOR	1.0	1.0	yes
De'Aaron Fox	SAC	1.0	1.0	yes
Joel Embiid	PHI	1.0	1.0	yes
Giannis Antetokounmpo	MIL	1.0	1.0	yes
Stephen Curry	GSW	1.0	1.0	yes
Damian Lillard	POR	1.0	1.0	yes
DeMar DeRozan	CHI	1.0	1.0	yes
Paul George	LAC	1.0	1.0	yes
Kyrie Irving	DAL	1.0	1.0	yes
Kevin Durant	PHX	1.0	1.0	yes
LeBron James	LAL	1.0	1.0	yes
Donovan Mitchell	CLE	1.0	1.0	yes
Lauri Markkanen	UTA	1.0	1.0	yes

Figure 1*:* Final results of the 2023 all-star predictions based on our model.

Visualization of Results

To dive even deeper into our model's prediction, we performed a PCA analysis on our dataset to determine which statistics are the most important for a player's chances in all-star selection. Our

PCA analysis gave us the following results (note that the features are listed in order of decreasing variance):

First Principle Component : Points, Win Shares, Turnovers, Assists, Steals, Usage Percentage
Second Principle Component: Blocks, True Shooting Percentage, Rebounds

It is interesting to note that the features in the second principle component favor big man positions (centers and power forwards).

We were interested in seeing which players were playing at a level comparable to the 2023 all-stars, but were not selected through the NBA's voting system. Thus after performing PCA analysis, we created *Figure 2*, as seen below, of all non-all-stars and where they rank compared to all-star players, and identified where the players our model selected as all-stars lie.

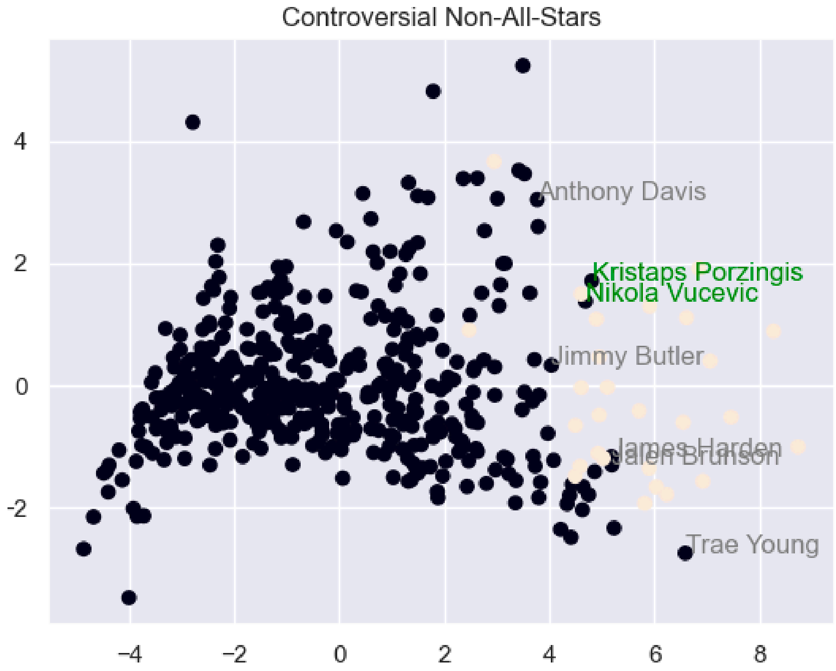


Figure 2*:* Controversial Non-All-Stars and where they rank compared to all-star players.

As we can see from *Figure 2*, our model did a great job at predicting the all-stars, as the predicted players were clearly playing at an all-star level throughout the 2023 season. However, we can also see that Kristaps Porzingis and Nikola Vucevic were also playing at an all-star level caliber, yet our model did not predict either of them to be all-stars.

The last thing we wanted to investigate was which players were potentially controversial all-star selections (according to our PCA analysis). Thus, we created *Figure 3* below which identifies some of the 2023 all-stars who were playing at a lower level than the top players in the league.

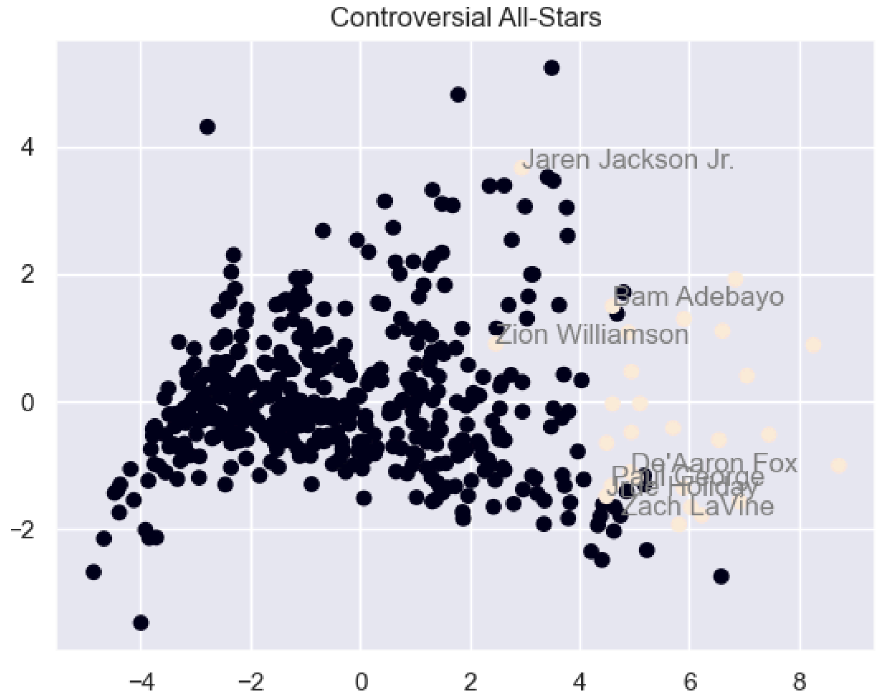


***Figure 3**:* Controversial All-Stars and where they rank compared to the rest of the all-stars.

An interesting case seems to be Zion Williamson. Zion this year only played 29 games out of the 50 or so games prior to the all-star break, meaning that his stat totals suffered as a result. This may explain why he seems to be surrounded by non-all-star players in the *Figure 3* despite playing at a very high-level for the season.

It is also worth mentioning that Jaren Jackson Jr. was the defensive player of the year in 2023, hence why he seems to rank so highly on the y-axis of this plot. The y-axis represents the second principle component which seems to be more oriented towards defensive features such as blocks and rebounds.

The notable names that our model did not select are Jaren Jackson Jr., Bam Adebayo, and Jrue Holiday.

Limitations

The first problem we faced was that the NBA does not provide league statistics prior to 1996. If we had access to that data, we would have over 30 extra years of training data for our model. However, in retrospect this is an acceptable limitation as the NBA has changed so much over the years that what was valued in the 70's is drastically different than the present. For example, today's NBA has an unprecedented emphasis on the three-point shot, so that will have a greater impact on whether a player is selected for the All-Star game. Therefore, it may not be appropriate to go too far back in the past for the purposes of this project.

Another problem we encountered was how long it took to gather the large amounts of data our project required. The API we used bottlenecked our data gathering process, where we were only able to make a request around once every second. Because of this data gathering took enormous amounts of time. For example, the step in our pipeline to gather all game logs from 1996 to 2023 took more than 24 hours to complete.

Another issue we had was gathering advanced statistics. Our chosen API did not seem to have any functions that we could call to gather advanced stats like PER (player efficiency rating) or WS (win shares). At the time, we chose to use the data available to use from the API and manually calculate these advanced statistics. However, in hindsight, it would have likely been better to create a web scraping tool and gather the data from a site like basketball-reference.com where such advanced statistics are publicly available. Not only might this have been a faster way to gather data, but it would guarantee accurate results as manually calculating advanced statistics turned out to be an error prone and tedious process. One tradeoff though is that these would be advanced statistics for the whole season, rather than the portion of the season prior to the all-star game, which is not totally appropriate data for this project.

Relating to advanced statistics, if we had more time we would have likely calculated features other than PER and WS. Some other interesting advanced statistics that may have proved useful include: DWS (defensive win shares) and VORP (value over replacement player).