CSC111 Project Proposal: An Analysis of the Modern NBA "Big Man"

Chris Chan

Friday, April 16, 2021

Problem Description and Research Question

I am a big fan of basketball. Specifically, I am a fan of the NBA, the Toronto Raptors being the team I primarily follow. My favourite players in the league are often power forwards and centers - positions that traditionally take advantage of their size to impact the game. In the basketball world, the term "big men" is often colloquially used to refer to centers (Big Man Definition - Sporting Charts). I, however, will be using it in reference to both centers and power forwards.

In the last decade or so, basketball has seen a shift in play style with the rise in popularity of the three point shot. CBS sports writer Bill Reiter is just one of many who attribute this shift to the revolutionary play of NBA superstar Stephen Curry. In his article, he asserts that players like Stephen Curry have revolutionized the traditional game of basketball when alluding to a list of the top 15 shooters in NBA history: "One sharpshooter after another, each a stepping stone in an evolution of the game that has made the traditional center obsolete and the game a vastly different thing than it was even a decade ago"(Bill Reiter, Jun 1). In a sort of blog, one basketball coach writes, "Today, the NBA is run by [point guards and shooting guards]. The fast paced, 3-point shooting game doesn't allow traditional big men to remain relevant." (Chris Are Big Men Becoming Obsolete?). Just from watching games in the last couple of years and comparing it to highlights of before I started watching, their beliefs are quite evident - I can tell that there has been a clear shift in play-style. As much as I agree, as a fan of traditional big men, I find it difficult to accept. Through this project, **I want to determine whether the modern NBA has truly left-behind the big men of old, or have they simply just evolved with it.** Do they still impact games the way they once did? Are they more effective or less effective? Through this project, I will answer all of these questions.

Dataset Descriptions

Basketball: https://www.kaggle.com/wyattowalsh/basketball

- This is an sql file containing multiple tables of data spanning over 70 years. I chose to use the table titled 'Player_Attributes', and of it, I chose to use the following columns: 'FIRST_NAME, LAST_NAME, HEIGHT, WEIGHT'.

NBA Players stats since 1950: https://www.kaggle.com/drgilermo/nba-players-stats?select=playerdata.csv

- This is a data set containing 3 csv files of every NBA players total stats for each season they played. I chose to use the file titled 'Seasons_Stats.csv' and used all read all columns.

2021 NBA Player Stats: Per Game: https://www.basketball-reference.com/leagues/NBA_2021_per_game.html

- This data set is a csv which contains per game stats for NBA players in the current NBA season.

https://www.basketball-reference.com/leagues/NBA_2021_advanced.html

- This data set is a csv which contains advanced stats for NBA players in the current NBA season.

Computational Overview

My project used data about NBA players. There were two categories: some of the data represents stats for players this NBA season (2020-21 NBA Player Stats: Per game). The rest represents seasonal averages and totals for players

that played between 1947 - 2018 (Goldstein NBA Players stats since 1950). Each unique player is added to a graph and an edge is created between vertices/players based on their 'big man score'. This connection dictates a degree of similarity which seeks to help answer my overarching question.

Firstly, my program reads all the data sets into pandas Dataframe objects. The core of this is done in the module 'data_process.py' and is primarily done by built-in pandas methods. Each data set required a different type of filtering and cleaning. For example, the data set with the table 'Seasons_Stats.csv' had numerous seasons for each player and recorded stat totals. In 'data_process.calculate_average(), I opted to calculate the average over all seasons before calculating the per game numbers. In another data set, I removed rows with players who appeared more than once because they had played for more than one team this season. At the very least, each dataset was filtered to only retain players whose position satisfied "big men': power forwards and centers. This was done by the general purpose function 'df_filter(). After building up a DataFrame for each data set, and then merging where necessary, I then proceeded to build a graph from each, where unique player name was entered as a player vertex. This was the job of the 'create_graph() function in module 'create.py'. I then connected the graph by adding edges between players who were considered "similar". For my project, I came up with a 'big man score' value for each player. It is a number representing the weighted average of scores given to different aspects of basketball: a number is attributed to and is based on defensive stats, another is attributed to and based on offensive stats, etc.I then gave a weight to each category relative to its impact in quantifying a "true big man". Players were connected if their scores were within a certain range from one another. I will expand upon this in the discussion section.

To report my computations, I produced a simple tkinter function which takes the graph, and allows the user to treat it like a data base. Making use of widgets such as tkinter.Button and tkinter.Entry, I created a search functionality which outputs the data stored for the searched player. After a player is searched, I made use of the Radiobutton widget to allow the user to then select a player adjacent, and thus similar to the original player.

Instructions for Obtaining Datasets and Running my Program

- Download the zip file shared with the course email using https://send.utoronto.ca
- Extract the folder called 'data' to the same location as the python modules. It will contain all the datasets.
- Run `run_visual()` in main.py. After a few moments, you should see a 1200 x 800 tkinter window with a search bar. While there are players with accents in their names, the program adjusted this to only ascii characters. The program is only able to read names as they are in the data set. If you need a player to start with, enter 'Chris Boucher', Montreal's very own!
- After typing, you may press the Return key (enter) on your keyboard or the 'enter' button on the screen to search. If nothing happens, the name was incorrectly spelt.
- After searching, buttons will appear denoting players that are adjacent to the current search, you may click these buttons to search that player.

Description of Changes

I had originally planned to determine "similarity" based upon individual stats, however, in practice, I could not come up with a feasible way to do so. Instead, I opted to calculate a number which measured the degree to which they fit the mold of the "traditional big man". Not only was this simple, I also felt that is was more suited to answering my original question.

Discussion

I would say that the results of my computational exploration *do not* help answer my original question. In retrospect, I think there are two main reasons for this. The first lies in one of the data sets, but more so in how I aggregated over it. I had completed the function mentioned previously `calculate_average()` before realizing that it made little sense to do so, particularly because data that had missing values were removed. Take for example NBA legend Kareem Abdul-Jabbar. Kareem started his career in the early 70s, but I found that some of the data is missing from that time (understandbly so, not even considering the fact that the 3pt line and 3pt stats didn't exist then), the

rows of which I then decided to omit. This led to Kareem, someone who stands as the NBA's record holder for most points scored in the history of the NBA, not properly being represented in my program. Instead, he is represented by the latter years of his career. On the other hand, I am comparing these underestimates to inflated stats. The stats representing the current players is exactly that, their stats as of today (or whenever I saved the data), where they have played less games than in a typical season. The second reason it does not help answer my question is because I know that my formula for 'big man score' is *very* flawed. I had no basis for it other than the fact that I wanted the value to be around a scale of 10. When coming up with it, I calibrated the formula around two players, one who I consider the epitome of traditional big men, Shaquille O'neal, and the other being the first of his kind, a passing wizard at the center position, Nikola Jokic. As a result of my narrow calibration, as well as the inflated stats, the players that are compared to each other aren't the greatest. While I've seen instances where I was satisfied with the comparison, I think it still requires a lot of work. For example, the player who I built it around, Shaq, has a much lower score than the current player Joel Embiid, or just the fact that Shaq is adjacent to players who average less than 5 points... Additionaly, the range of similarity - that is, the minimum difference required to be considered "similar" was chosen arbitrarily within reason. With this many external factors to consider, its hard to say that any conclusion can be made towards answering my question with my program.

I feel I was limited in terms of the visualization aspect because tkinter had did not function as I expected, in the limited time I used it. For example, I had to change certain widget choices, which were less than ideal, and the desired formatting of data to be showcased was more difficult to achieve than I had anticipated.

For further exploration, of course the first step would be to improve the similarity algorithms. This could be done by coming up with a better formula, or maybe introducing a clustering algorithm in place of the big man score. I could also make more use of some of the other stats I collected in this project, which I ultimately avoided as 50 columns of data, many of which are 'advanced stats', was too overwhelming.If I did use those stats though, I could perhaps do more computations to determine other ways of determining 'similarity'. Another way I could explore this is by adding a genuine visualization function to this program similar to a3. This could help with improving the similarity algorithm.

References

"2020-21 NBA Player Stats: Advanced." Basketball, www.basketball-reference.com/leagues/NBA_2021_advanced.html.

2020-21 NBA Player Stats: Per Game. www.basketball-reference.com/leagues/NBA_2021_per_game.html.

"Big Man Definition - Sporting Charts." SportingCharts.com, www.sportingcharts.com/dictionary/nba/big-man.aspx.

Bill Reiter Jun 1. "How Stephen Curry Ushered NBA's Greatest Shooting Era and Changed the Perception of Championship Teams." CBSSports.com, 1 June 2020, www.cbssports.com/nba/news/how-stephen-curry-ushered-nbas-greatest-shooting-era-and-changed-the-perception-of-championship-teams/.

Chris, Coach. "Are Big Men Becoming Obsolete?" Medium, Medium, 1 Aug. 2017, medium.com/@TheCoachChris_/are-big-men-becoming-obsolete-c32b8f9bbafe.

Goldstein, Omri. "NBA Players Stats since 1950." Kaggle, 27 Apr. 2018, www.kaggle.com/drgilermo/nba-players-stats?select=player_data.csv.

"Tkinter Dialogs¶." Tkinter Dialogs - Python 3.9.2 Documentation, docs.python.org/3/library/dialog.html#module-tkinter.simpledialog.