Data Science Internship Paper

Through this semester I was able to complete a data science internship under Dr. Che. The scope of this essay is to discuss my goals for my project, the extent to which I was able to accomplish those goals, and key learnings throughout the journey of the project. From the beginning I knew that I wanted to work on a project related to something I am already knowledgeable about. The techniques for parsing and working with large datasets I was unfamiliar with so learning those tools as well as the intricacies of the field relating to a dataset seemed like it would be a monumental task. I have been a basketball fan my entire life and in recent years data analytics have become a huge factor in the way teams are constructed. As such I decided to perform data analysis on NBA team and player statistics to determine a path to building a championship team.

For me, the way that I approach a project where I don't have a strong background in the tools already is to develop a strong foundation first. I have some experience with Python but not very much so by starting with the Applied Data Science with Python Specialization from the University of Michigan (on Coursera) I developed a baseline knowledge with the tools I will need to complete this project. The first three course (Introduction to Data Science in Python, Applied Plotting, Charting, & Data Representation in Python, and Applied Machine Learning in Python) took up most of my time for the first few weeks of the semester but that was time well spent. I developed a fluency in numpy (the Python math library), Pandas (used for data manipulation and cleaning), and MatPlotLib for data visualization. Additionally, I gained a lot of useful experience working with Jupyter Notebooks to work with Python in a live environment which is how I chose to work with the NBA data I'm using on Kaggle.

While working on a foundation in the programming required for this project I decided to build my expertise in professional basketball analytics. Every year MIT Sloan puts on a sports analytics conference with their panels recorded. I watched and took rigorous notes on several panels to figure out where the true experts were focusing their attention and what sort of questions they were asking. NBA offices have huge analytics departments where they work on complex problems so obviously the scale of my findings would be much smaller, but taking the temperature of the industry was useful in steering me to a specific question. I also purchased and read *Basketball Analytics: Objective and Efficient Strategies for Understanding How Teams Win* by Stephen Shea and Christopher Baker. The book has a lot of specifics of how advanced metrics are calculated. While I won't be performing those calculation myself necessarily, I found the material immensely useful in determining which statistical categories I should be focusing on and which of them are mostly useless.

As I neared the end of my time with Coursera and the NBA specific material I started working on Kaggle. At the beginning of the semester I searched Kaggle for a dataset of NBA related statistics and found what was a massive set of statistics. The dataset I started working with encompasses individual total statistics for several basketball leagues (including the NBA) dating back to 1999. The data was massive so at first there was a fair bit of cleaning involved. I started working with the data using Pandas and as I was working with the data the exact purpose of my project came into focus. Watching the panels from the Sloan conference made it clear that the only true goal of every NBA franchise is to win a championship. Of course this is intuitive to sports fan but the way it was discussed by the professionals revealed to me that any work I would do with this data that revealed anything less than how to win the title would be essentially useless.

The issue that I ran into almost immediately was that my dataset was composed of only individual statistics and to win a championship you need a team. Within the dataset each player was associated with a team for each season so I attempted to work around this by manually calculating team totals. I separated the dataset into playoffs and regular season statistics, grouped by team and summed up each individual column. This seemed to be ok but after completing these calculations I went and sought out known team averages to compare and see how accurate I was. In general my method was pretty effective but in the case that a player changed teams midway through the season it created some inaccuracies. I also ran into the problem when calculating per/game statistics that I needed to know exactly how many games were played that season. It's rare any player plays every game in a year so choosing the max games played value wouldn't work and with the lockout in 2012 and the COVID shortened 2019 season I found myself adding magic numbers to the program. It became clear that I was working with incomplete information and would need to look for something else.

After some searching I came across the website basketball-reference.com. Basketball Reference is a phenomenal resource with tables for every sort of statistical breakdown of the sport you could possibly want. There I found everything I needed. In the end I created six .csv files, one for player salaries, one for league average per-game statistics over the years, one for the individual player per-game statistics from the 2019-20 season, one for 2019-20 individual player advanced metrics. The final two .csv files were team per-game statistics for both regular season and playoffs for every season since the 2009-10 season. Unfortunately, Basketball Reference only has team statistics in tables for each individual season. As such I had to gather the table for each individual season in comma separated format and combine them, adding a

season column to differentiate between years. With all of these files I was now armed with all of the data I needed to answer the task of building a champion.

After finding all of the data and creating them as datasets in Kaggle I was ready to start solving the problem. The first thing I decided to do was to visualize where the league currently stands statistically. According to several panelists from the Sloan panels that I watched it's considered common knowledge that regular season and playoff basketball are very distinct. As such I decided to focus in on what I deemed to be key indicators of team success and plot those values for playoffs and regular season side by side. For this visualization I chose a box plot because that best represents outliers and a median. The result is shown in Figure A. This chart represents team average points, assists, rebounds, blocks, turnovers, steals, free-throw attempts and threes made per game for the 2019-20 season (broken out regular season vs. playoffs).

The specific values of the 2019-20 season averages are not too helpful on their own in projecting what we need to build a championship roster. In order to determine what we need I want to take a two part approach. First, I want to know what statistical categories are most relevant in a straight comparison between past champions and the league averages the year they won the title. What results is Figure B. Here I went and found the last 11 NBA champions. I grabbed their team averages for the regular season and charted it against the league median average for each category. What becomes clear immediately are a couple of things: with the exception of the 2019-20 Championship Lakers, champions always average more free throws and threes than the league (which supports background research I've done before). The other major takeaway from that overview is that champions are almost always at least even with the league median in every single one of those categories.

After getting a clearer picture of how a prospective champion needs to compare to the league they are competing against, the next logical step is to make a prediction of what that league will look like. To do this I wanted to have a clear picture of the trends across the league in the eight statistical categories that we have been looking at. I took the dataset that I had of all league averages and plotted the averages for those statistics for every season since the 2002-03 season. I detail some of this in the final notebook but I chose to start in the 2002-03 season because that's widely regarded as the first season since the last major rules change. Between the 2002-03 season illegal defense rules were changed, opening the lane more widely for offensive output than it had ever been before. This was a relatively simple procedure to get all of the statistics together in one plot (with multiple subplots) and can be seen in Figure C. After viewing the figure it becomes clear that there aren't substantial spikes or dips between seasons in any of the categories (though there is a general upward trend in almost every category). Given that I chose a simple algorithm to predict the 2020-21 stats. I took the difference between the last two seasons and added it to last year's stats. Now I have my baseline. If I'm going to build a contender I need to be at least league average in every category next season.

The last dataset I brought in was player salaries. There was some more cleaning involved in using this data than there was with any other set (i.e. salaries were strings with commas and dollar signs that needed to be converted to numbers). This is explained in more detail in the notebook but in order to build a champion it's well known that you not only need to hit the statistical goals I have set but you need a top 15 player to lead the team. According to Shea and Baker in *Basketball Analytics* the clearest way to determine the relative ranking of individual players is PER (Player Efficiency Rating). This number was available in the advanced metrics dataset that I included in the project. For simplicity's sake I chose the player with the highest

rating in the league, Giannis Antetokounmpo. I added him and included a rookie on our roster. This means that I needed to fill 13 slots out our 15 man roster. The hard salary cap for the 2020-21 season is $145,470,158. To start I added the stats Giannis produced last year to our team totals and subtracted his salary from the cap. Finally, I created a nested loop structure to iterate over every possibility of 13 players from the 650 players in the league last year. Unfortunately this is where I ran into a snag. I had a lot of issue getting just the basic concept off the ground but eventually followed these steps:

1. Grab 13 rows from the players dataframe
2. Take each of their names, points, assists, blocks, rebounds, steals, turnovers, free throws, and threes and sum them up.
3. If each team total is at least equal to our calculated statistical league average for next year then:
    a. Sum the player's salaries by grabbing them from the salary dataframe
    b. If the total salary (including Giannis' salary) is less than the cap, then this is our team. We exit the loops and output a dataframe of the 14 players (not including the rookie who doesn't exist as a row anywhere)
4. Else:
    a. Iterate an indexer and change the 13^{th} player.
5. After checking every possible 13^{th} player do the same with the 12^{th} slot, 11^{th} slot, etc.

The key problem I ran into was the inability to test. Looping through the entire dataframe was something I tried to avoid as much as possible but was something I had to do if I thought it was possible I had a solution. To complete the loop took a couple of hours so to test I set a cutoff point of 30 minutes but even so it became too unwieldy and I ran out of time. It is also possible

that there is no combination of players that meets the criteria. That would be interesting because it actually means it's easier to win a championship than I had been assuming. I spent a fair bit of time scouring the web for a better algorithm to test possible combinations but I came up empty. I believe my tool can be perfected and I believe, if it worked, it would find a roster that could genuinely compete. There are also considerations to team chemistry but those cannot be quantified, and that's why there will always be human general managers making the final personnel decisions. Given infinite time I'm sure I could make this work but I spent 16 hours the last two days of the semester working on it and made little to no progress. My time with the project is done.

Taking a step back I found this experience to be extremely valuable. This is the most interested I have been in any school related project in my entire academic career and it has opened my eyes to possibility of bringing data science and sports together into a potential career. I also gained a true appreciation for Python as a language. Python has a vast amount of libraries available and works so efficiently it's incredible the things you can accomplish in one line. Not only have I increased in my confidence with the language, but I will be counting it as my preferred language for completing projects in the future. Finally I would like to thank Dr. Che for agreeing to oversee my project. This has been an incredibly unusual semester for everybody and without this opportunity I would have had a much more difficult time graduating this December.

Figure A:

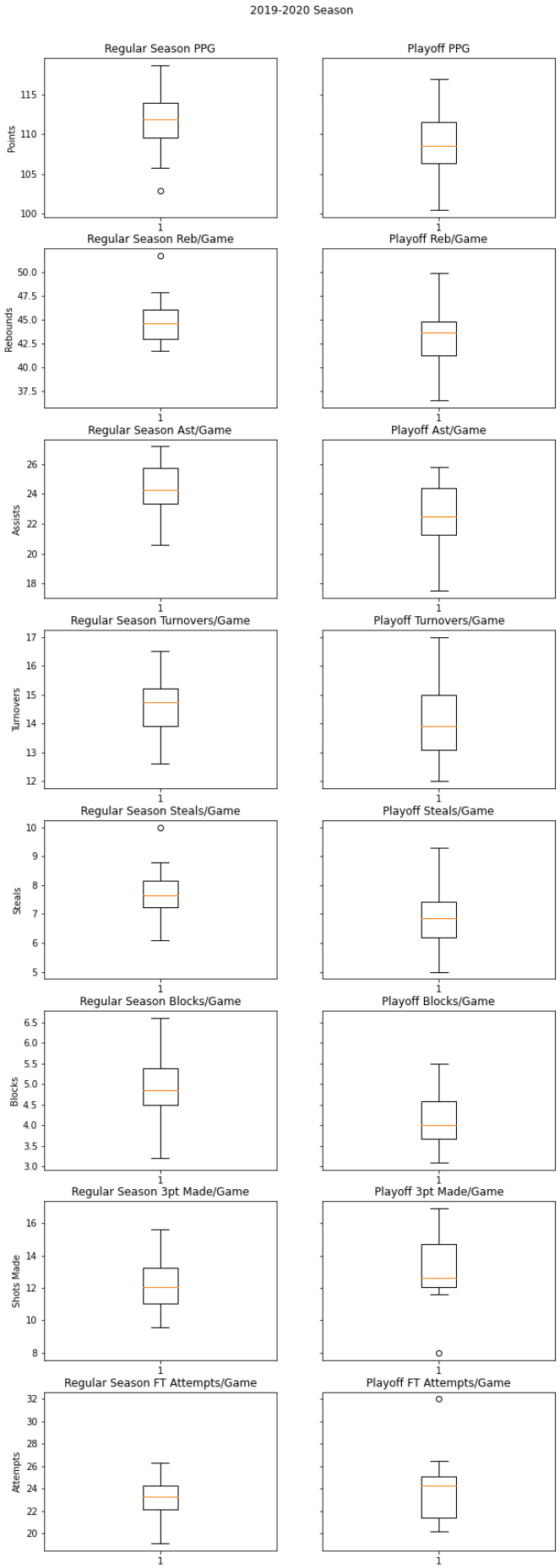

2019-2020 Season
Regular Season PPG
Playoff PPG
Points
Regular Season Reb/Game
Playoff Reb/Game
Rebounds
Regular Season Ast/Game
Playoff Ast/Game
Assists
Regular Season Turnovers/Game
Playoff Turnovers/Game
Turnovers
Regular Season Steals/Game
Playoff Steals/Game
Steals
Regular Season Blocks/Game
Playoff Blocks/Game
Blocks
Regular Season 3pt Made/Game
Playoff 3pt Made/Game
Shots Made
Regular Season FT Attempts/Game
Playoff FT Attempts/Game
Attempts

Figure B:

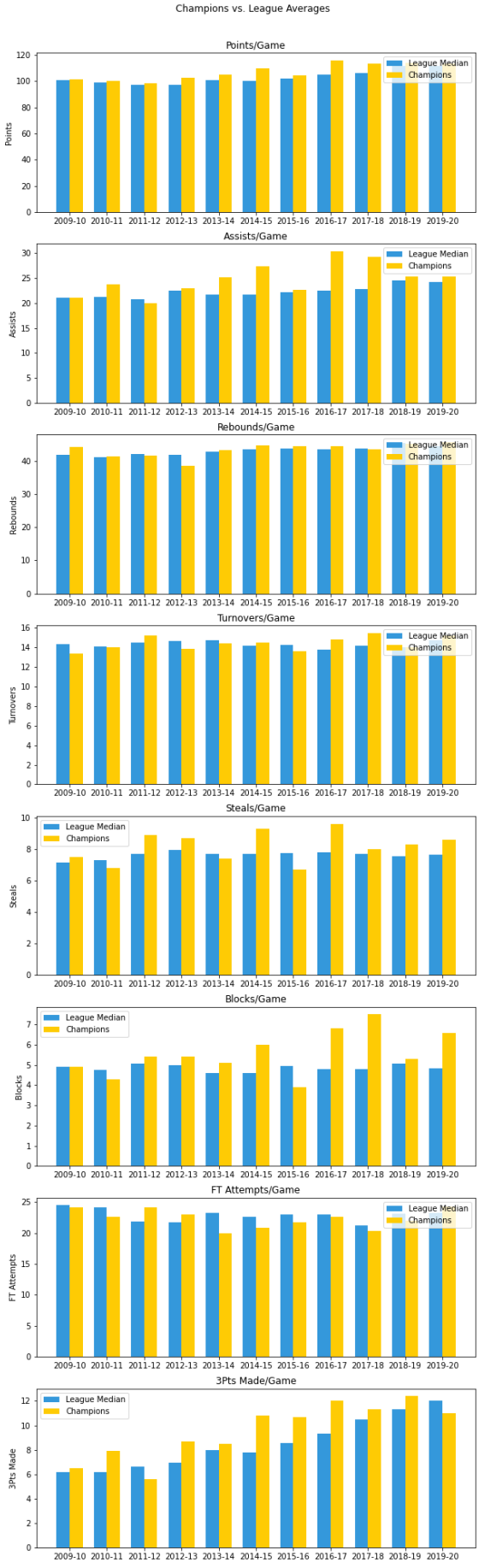
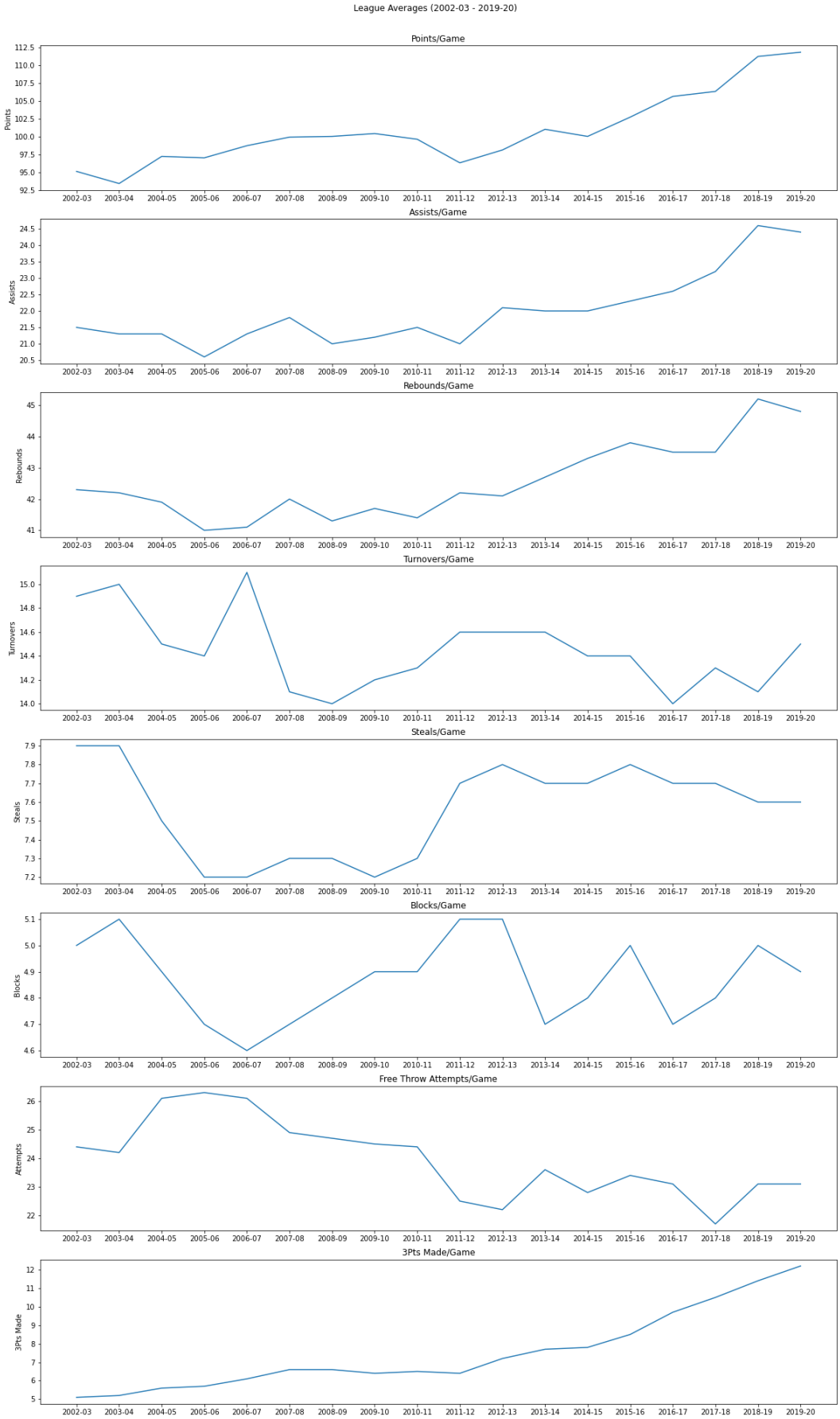

Champions vs. League Averages
Points/Game
Points
Assists/Game
Assists
Rebounds/Game
Rebounds
Turnovers/Game
Turnovers
Steals/Game
Steals
Blocks/Game
Blocks
FT Attempts/Game
FT Attempts
3Pts Made/Game
3Pts Made
League Median
Champions
2009-10
2010-11
2011-12
2012-13
2013-14
2014-15
2015-16
2016-17
2017-18
2018-19
2019-20

Figure C:


League Averages (2002-03 - 2019-20)
Points/Game
Points
Assists/Game
Assists
Rebounds/Game
Rebounds
Turnovers/Game
Turnovers
Steals/Game
Steals
Blocks/Game
Blocks
Free Throw Attempts/Game
Attempts
3Pts Made/Game
3Pts Made
2002-03
2003-04
2004-05
2005-06
2006-07
2007-08
2008-09
2009-10
2010-11
2011-12
2012-13
2013-14
2014-15
2015-16
2016-17
2017-18
2018-19
2019-20