Using Machine Learning for NBA Game Outcome Prediction

Rohan Kosalge, Rida Assaf

9 July 2021

1 Abstract

Predicting the outcomes of games in the National Basketball Association (NBA) has been an abiding challenge throughout the league's history. However, since its widespread population this last decade, it has shown importance in the vast business of sports betting and emerged as a tool for teams and their coaches. NBA games are difficult to predict, due to the high volume of points normally scored by both teams in a match, as well as extra factors such as injuries and underwhelming player performances. Machine Learning is useful for data mining and analysis, and is efficient in recognizing patterns within large sets of statistics. Similar to the emergence of the NBA and sports betting, Machine Learning has become one of the most preferred ways to solve some of the most perplexing problems. In this paper, we utilize different machine learning algorithms such as Linear Regression, Logistic Regression, Support Vector Machines, Random Forest, and Multilayer Perceptrons to achieve a high accuracy of NBA game predictions. We use data scoured from certified sources such as Basketball Reference that include advanced features such as True Shooting Percentage, Usage Rate, and Box Plus/Minus. Overall, we were able to achieve a maximum accuracy of 70.569% of games predicted successfully, finishing marginally close to the state-of-the-art models.

2 Introduction

The National Basketball Association (NBA) is one of the most affluent and popular sports leagues in the world, generating billions of dollars every season and attracting millions of fans on a global scale. With the advent of sports betting, an industry now worth over $50 billion, the importance of predicting the outcomes of NBA games as precisely as possible has grown among teams and fans alike [1]. Fans participate in 'fantasy' leagues and NBA 2K, the leading video game market of the NBA, where they are able to construct their own teams and simulate games against others in hopes of winning. Accurate predictions may assist teams with player trades and improved coaching in the off-season.

Overall, the utilization of game predictions has rapidly increased within the last decade of the NBA itself [2].

Predicting NBA games is a challenging task, and many related models score within the 65-72% mark [3][4]. Figuring out the most ideal features to apply and methods to implement creates another challenge to resolve, contributing to the abiding impact of sports betting.

This paper aims to utilize machine learning methods for predicting the outcomes of NBA games, incorporating detailed statistics that are more capable of describing the impact of an NBA game than individual team statistics and 'standard' box score statistics. We explain why these statistics were chosen to be utilized for the models and identify the most important features required to most accurately identify the victor. Then, we compare the accuracies of different machine learning methods and keep track of the most error-free ones. Puranmalka heavily relied on the use of SVMs to get accurate results, while also using more unorthodox statistics to headline his model [5]. Besides SVMs, Cao incorporated another algorithm in Artificial Neural Networks [6]. Beckler, Wang, and Papamichael additionally utilized Linear Regression and Logistic Regression algorithms, mainly testing player and game stats to obtain precise results [7]. We utilize all of these algorithms in our project with the goal of generating accuracies that come close to leading models in the basketball betting business.

3 Methodology

The first part of the project was scraping all the data we needed online and processing it accordingly. The data consists of a large number of features, any of which could be easily taken out or replaced. After saving this data, the second part was to select the most important features and process it through a machine learning model. With several tweaks in the data features, model type, and the number of seasons that are trained, the last part, as well as the goal of this project, was obtaining the best results possible from the model.

3.1 Data

All data used for this project has been scraped from a site called Basketball Reference (basketball-reference.com) [8], an extensive and detailed NBA database which provides statistics of players, teams, games, standings, records, and much more. Compared to other sites such as NBA (nba.com) and ESPN (espn.com), Basketball Reference provides more advanced statistics, displays statistics in a simpler and downloadable format, and offers data from every single season since the league's inauguration in 1946. We show an example page from the site in Figure 1.

Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	FG%	3P	3PA	3P%	2P	2PA	2P%	eFG%	FT	FTA	FT%	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS
1	Precious Achiuwa	PF	21	MIA	61	4	12.1	2.0	3.7	.544	0.0	0.0	.000	2.0	3.7	.546	.544	0.9	1.8	.509	1.2	2.2	3.4	0.5	0.3	0.5	0.7	1.5	5.0
2	Jaylen Adams	PG	24	MIL	7	0	2.6	0.1	1.1	.125	0.0	0.3	.000	0.1	0.9	.167	.125	0.0	0.0		0.0	0.4	0.4	0.3	0.0	0.0	0.0	0.1	0.3
3	Steven Adams	C	27	NOP	58	58	27.7	3.3	5.3	.614	0.0	0.1	.000	3.3	5.3	.620	.614	1.0	2.3	.444	3.7	5.2	8.9	1.9	0.9	0.7	1.3	1.9	7.6
4	Bam Adebayo	C	23	MIA	64	64	33.5	7.1	12.5	.570	0.0	0.1	.250	7.1	12.4	.573	.571	4.4	5.5	.799	2.2	6.7	9.0	5.4	1.2	1.0	2.6	2.3	18.7
5	LaMarcus Aldridge	C	35	TOT	26	23	25.9	5.4	11.4	.473	1.2	3.1	.388	4.2	8.3	.505	.525	1.6	1.8	.872	0.7	3.8	4.5	1.9	0.4	1.1	1.0	1.8	13.5
5	LaMarcus Aldridge	C	35	SAS	21	18	25.9	5.5	11.8	.464	1.3	3.6	.360	4.2	8.2	.509	.518	1.5	1.8	.838	0.8	3.7	4.5	1.7	0.4	0.9	1.0	1.7	13.7
5	LaMarcus Aldridge	C	35	BRK	5	5	26.0	5.0	9.6	.521	0.8	1.0	.800	4.2	8.6	.488	.563	2.0	2.0	1.000	0.4	4.4	4.8	2.6	0.6	2.2	1.4	2.2	12.8
6	Ty-Shon Alexander	SG	22	PHO	15	0	3.1	0.2	0.8	.250	0.1	0.6	.222	0.1	0.2	.333	.333	0.1	0.1	.500	0.1	0.5	0.7	0.4	0.0	0.1	0.2	0.1	0.6
7	Nickeil Alexander-Walker	SG	22	NOP	46	13	21.9	4.2	10.0	.419	1.7	4.8	.347	2.5	5.2	.485	.502	1.0	1.4	.727	0.3	2.8	3.1	2.2	1.0	0.5	1.5	1.9	11.0
8	Grayson Allen	SG	25	MEM	50	38	25.2	3.5	8.3	.418	2.1	5.5	.391	1.3	2.8	.471	.547	1.6	1.8	.868	0.4	2.8	3.2	2.2	0.9	0.2	1.0	1.4	10.6
9	Jarrett Allen	C	22	TOT	63	45	29.6	4.7	7.7	.618	0.1	0.3	.316	4.6	7.3	.631	.624	3.2	4.6	.703	3.1	6.9	10.0	1.7	0.5	1.4	1.6	1.5	12.8
9	Jarrett Allen	C	22	BRK	12	5	26.7	3.7	5.4	.677	0.0	0.0		3.7	5.4	.677	.677	3.8	5.1	.754	3.2	7.3	10.4	1.7	0.6	1.6	1.8	1.8	11.2
9	Jarrett Allen	C	22	CLE	51	40	30.3	5.0	8.2	.609	0.1	0.4	.316	4.9	7.8	.623	.616	3.1	4.5	.690	3.1	6.8	9.9	1.7	0.5	1.4	1.5	1.5	13.2
10	Al-Farouq Aminu	PF	30	TOT	23	14	18.9	1.7	4.3	.384	0.3	1.6	.216	1.3	2.7	.484	.424	0.8	1.0	.818	1.0	3.8	4.8	1.3	0.8	0.4	1.2	1.3	4.4
10	Al-Farouq Aminu	PF	30	ORL	17	14	21.6	2.1	5.2	.404	0.4	1.8	.226	1.7	3.4	.500	.444	0.8	1.0	.824	1.2	4.2	5.4	1.7	1.0	0.5	1.5	1.3	5.5
10	Al-Farouq Aminu	PF	30	CHI	6	0	11.2	0.3	1.7	.200	0.2	1.0	.167	0.2	0.7	.250	.250	0.7	0.8	.800	0.3	2.8	3.2	0.3	0.3	0.0	0.5	1.2	1.5

Figure 1: First Ten Players (listed alphabetically) of the Per Game Stats Page from the 2020-2021 NBA Season

A group of data we processed is team standings, most importantly the number of wins and losses that both teams have prior to the start of the game. For example, a 50-win team that is matched up to play against a 20-win team has a higher chance of winning, as they have won more games in nearly the same amount of games played in that season so far. To process these features, we obtained data from the 'Schedules' Section of Basketball Reference, where we could access general info from every game played in a season. These attributes include the date of the game, the names of the Away and Home teams, and the amount of points scored by each team. Figure 2 shows an example page from the 'Schedules' Section.

This group of data gave us the ability to iterate through every game and keep a tab of different categories of standings for every team. We kept a record of five different types of standings: Overall, Place (Home/Away), Conference, Division, and Last Ten Games. If a team has played less than 10 games prior to the match, then it is replaced with the 'Overall' standing. These features proved useful as they added other dimension to the strength of the team. For example, the 2015-16 San Antonio Spurs had an overall record of 67 wins and 15 losses at the end of the season. However, they had a record of 40 wins and 1 loss when playing games at their home stadium. A total of 15 features were saved, as for every standing the number of wins, losses, and win percentage were recorded.

Another group of data that was utilized for the model was the individual game data for every game a team played. These types of statistics were taken from the 'Box Scores' section from Basketball Reference. A box score is a group of statistics recorded immediately after the closure of a game. it provides basic info such as the number of points that were scored and the number of shots that were made. Basketball Reference provides more advanced statistics aimed to measure a team's performance more accurately. A full list of these statistics with corresponding descriptions can be found in Appendix A.

Date	Start (ET)	Visitor/Neutral	PTS	Home/Neutral	PTS			Attend.	Notes
Tue, Dec 22, 2020	7:00p	Golden State Warriors	99	Brooklyn Nets	125	Box Score		0	
Tue, Dec 22, 2020	10:00p	Los Angeles Clippers	116	Los Angeles Lakers	109	Box Score		0	
Wed, Dec 23, 2020	7:00p	Charlotte Hornets	114	Cleveland Cavaliers	121	Box Score		300	
Wed, Dec 23, 2020	7:00p	New York Knicks	107	Indiana Pacers	121	Box Score		0	
Wed, Dec 23, 2020	7:00p	Miami Heat	107	Orlando Magic	113	Box Score		3,396	
Wed, Dec 23, 2020	7:00p	Washington Wizards	107	Philadelphia 76ers	113	Box Score		0	
Wed, Dec 23, 2020	7:30p	New Orleans Pelicans	113	Toronto Raptors	99	Box Score		3,800	
Wed, Dec 23, 2020	7:30p	Milwaukee Bucks	121	Boston Celtics	122	Box Score		0	
Wed, Dec 23, 2020	8:00p	Atlanta Hawks	124	Chicago Bulls	104	Box Score		0	
Wed, Dec 23, 2020	8:00p	San Antonio Spurs	131	Memphis Grizzlies	119	Box Score		0	
Wed, Dec 23, 2020	8:00p	Detroit Pistons	101	Minnesota Timberwolves	111	Box Score		0	
Wed, Dec 23, 2020	9:00p	Sacramento Kings	124	Denver Nuggets	122	Box Score	OT	0	
Wed, Dec 23, 2020	10:00p	Utah Jazz	120	Portland Trail Blazers	100	Box Score		0	
Wed, Dec 23, 2020	10:30p	Dallas Mavericks	102	Phoenix Suns	106	Box Score		0	
Fri, Dec 25, 2020	12:00p	New Orleans Pelicans	98	Miami Heat	111	Box Score		0	
Fri, Dec 25, 2020	2:30p	Golden State Warriors	99	Milwaukee Bucks	138	Box Score		0	
Fri, Dec 25, 2020	5:00p	Brooklyn Nets	123	Boston Celtics	95	Box Score		0	
Fri, Dec 25, 2020	8:00p	Dallas Mavericks	115	Los Angeles Lakers	138	Box Score		0	
Fri, Dec 25, 2020	10:30p	Los Angeles Clippers	121	Denver Nuggets	108	Box Score		0	
Sat, Dec 26, 2020	5:00p	Atlanta Hawks	122	Memphis Grizzlies	112	Box Score		0	

Figure 2: First 20 games of the 2020-21 NBA season

Overall, a total of 35 general features were used. We further divide these features into two subgroups of statistics. These subgroups eventually come together as a large group of features for the model to train on. The first subgroup is the team's starters' game data. The starters of a team are the five players that start off a basketball game, and are considered to be the best players on the team, normally playing more minutes than any other players on the team. Figures 3 and 4 both display data from the starters of the opening night game of the 2020-21 season between the Golden State Warriors and Brooklyn Nets.

Starters	MP	FG	FGA	FG%	3P	3PA	3P%	FT	FTA	FT%	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	+/-
Andrew Wiggins	31:14	4	16	.250	2	6	.333	3	4	.750	0	2	2	1	0	1	4	4	13	-28
Stephen Curry	30:19	7	21	.333	2	10	.200	4	4	1.000	3	1	4	10	2	0	3	1	20	-23
Kelly Oubre	25:39	3	14	.214	0	6	.000	0	0		4	3	7	2	1	2	3	1	6	-28
James Wiseman	24:17	7	13	.538	1	1	1.000	4	8	.500	1	5	6	0	2	0	1	2	19	-10
Eric Paschall	21:33	2	6	.333	1	1	1.000	1	2	.500	1	0	1	0	0	1	0	1	6	-28

Figure 3: Basic Starter Statistics

The other subgroup of data compiled was the Team Totals. These include the starters as well as the 'bench' players, who play less minutes and are not as involved in the game compared to the starters. Team Totals can be seen within the statistics of the same game in Figures 5 and 6, highlighted in yellow at the bottom.

Starters	MP	TS%	eFG%	3PAr	FTr	ORB%	DRB%	TRB%	AST%	STL%	BLK%	TOV%	USG%	ORtg	DRtg	BPM
Andrew Wiggins	31:14	.366	.313	.375	.250	0.0	6.5	3.0	5.0	0.0	2.7	18.4	26.3	63	118	-14.0
Stephen Curry	30:19	.439	.381	.476	.190	8.3	3.4	6.1	61.1	2.9	0.0	11.6	32.1	99	114	6.0
Kelly Oubre	25:39	.214	.214	.429	.000	13.1	11.9	12.6	11.9	1.7	6.6	17.6	25.0	54	110	-11.0
James Wiseman	24:17	.575	.577	.077	.615	3.5	21.0	11.4	0.0	3.6	0.0	5.7	27.2	107	107	3.3
Eric Paschall	21:33	.436	.417	.167	.333	3.9	0.0	2.1	0.0	0.0	3.9	0.0	12.1	96	119	-6.9

Figure 4: Advanced Starter Statistics

	Basic Box Score Stats																		
Starters	MP	FG	FGA	FG%	3P	3PA	3P%	FT	FTA	FT%	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	+/-
Andrew Wiggins	31:14	4	16	.250	2	6	.333	3	4	.750	0	2	2	1	0	1	4	4	13	-28
Stephen Curry	30:19	7	21	.333	2	10	.200	4	4	1.000	3	1	4	10	2	0	3	1	20	-23
Kelly Oubre	25:39	3	14	.214	0	6	.000	0	0		4	3	7	2	1	2	3	1	6	-28
James Wiseman	24:17	7	13	.538	1	1	1.000	4	8	.500	1	5	6	0	2	0	1	2	19	-10
Eric Paschall	21:33	2	6	.333	1	1	1.000	1	2	.500	1	0	1	0	0	1	0	1	6	-28
Reserves	**MP**	**FG**	**FGA**	**FG%**	**3P**	**3PA**	**3P%**	**FT**	**FTA**	**FT%**	**ORB**	**DRB**	**TRB**	**AST**	**STL**	**BLK**	**TOV**	**PF**	**PTS**	**+/-**
Brad Wanamaker	21:40	0	2	.000	0	1	.000	3	4	.750	0	1	1	3	0	0	1	2	3	+1
Jordan Poole	17:55	1	4	.250	0	1	.000	0	0		0	2	2	3	0	0	2	2	2	-3
Juan Toscano-Anderson	13:15	2	2	1.000	0	0		0	0		0	4	4	2	0	2	0	3	4	-9
Marquese Chriss	12:26	4	10	.400	1	3	.333	0	0		3	5	8	1	0	0	0	2	9	-6
Damion Lee	12:14	1	2	.500	1	1	1.000	0	0		1	6	7	2	0	0	0	0	3	+8
Kent Bazemore	11:35	1	2	.500	0	1	.000	0	0		0	2	2	2	1	0	2	3	2	-3
Kevon Looney	11:17	2	4	.500	0	0		0	1	.000	0	2	2	0	0	0	2	3	4	-10
Mychal Mulder	6:36	3	3	1.000	2	2	1.000	0	0		0	1	1	0	0	0	0	0	8	+9
Team Totals	**240**	**37**	**99**	**.374**	**10**	**33**	**.303**	**15**	**23**	**.652**	**13**	**34**	**47**	**26**	**6**	**6**	**18**	**24**	**99**	

Figure 5: Team Totals for Basic Box Score Statistics

	Advanced Box Score Stats															
Starters	MP	TS%	eFG%	3PAr	FTr	ORB%	DRB%	TRB%	AST%	STL%	BLK%	TOV%	USG%	ORtg	DRtg	BPM
Andrew Wiggins	31:14	.366	.313	.375	.250	0.0	6.5	3.0	5.0	0.0	2.7	18.4	26.3	63	118	-14.0
Stephen Curry	30:19	.439	.381	.476	.190	8.3	3.4	6.1	61.1	2.9	0.0	11.6	32.1	99	114	6.0
Kelly Oubre	25:39	.214	.214	.429	.000	13.1	11.9	12.6	11.9	1.7	6.6	17.6	25.0	54	110	-11.0
James Wiseman	24:17	.575	.577	.077	.615	3.5	21.0	11.4	0.0	3.6	0.0	5.7	27.2	107	107	3.3
Eric Paschall	21:33	.436	.417	.167	.333	3.9	0.0	2.1	0.0	0.0	3.9	0.0	12.1	96	119	-6.9
Reserves	**MP**	**TS%**	**eFG%**	**3PAr**	**FTr**	**ORB%**	**DRB%**	**TRB%**	**AST%**	**STL%**	**BLK%**	**TOV%**	**USG%**	**ORtg**	**DRtg**	**BPM**
Brad Wanamaker	21:40	.399	.000	.500	2.000	0.0	4.7	2.1	18.0	0.0	0.0	21.0	8.3	90	120	-11.2
Jordan Poole	17:55	.250	.250	.250	.000	0.0	11.4	5.2	23.4	0.0	0.0	33.3	12.6	54	118	-13.9
Juan Toscano-Anderson	13:15	1.000	1.000	.000	.000	0.0	30.8	13.9	24.4	0.0	12.7	0.0	5.7	207	104	5.8
Marquese Chriss	12:26	.450	.450	.300	.000	20.3	41.1	29.7	17.9	0.0	0.0	0.0	30.4	103	109	2.3
Damion Lee	12:14	.750	.750	.500	.000	6.9	50.1	26.4	23.7	0.0	0.0	0.0	6.2	169	106	4.6
Kent Bazemore	11:35	.500	.500	.500	.000	0.0	17.6	8.0	25.2	3.7	0.0	50.0	13.0	63	107	-9.1
Kevon Looney	11:17	.450	.500	.000	.250	0.0	18.1	8.2	0.0	0.0	0.0	31.1	21.6	54	116	-18.8
Mychal Mulder	6:36	1.333	1.333	.667	.000	0.0	15.5	7.0	0.0	0.0	0.0	0.0	17.2	267	117	27.8
Team Totals	**240**	**.454**	**.424**	**.333**	**.232**	**22.8**	**72.3**	**45.2**	**70.3**	**5.4**	**10.5**	**14.2**	**100.0**	**89.6**	**113.1**	

Figure 6: Team Totals for Advanced Box Score Statistics

Team Totals do not include +/- (Plus/Minus) and BPM, as those are player-based statistics that measure a player's strength in relation to the average team player during that game. Additionally, the 'Minutes Played' feature was not included, as the values for both teams will always be the same (as both teams play the same amount of minutes at the end of the game).

All in all, there are 35 'sub-features' for starter statistics, and 32 'sub-features' for overall team statistics, which add up to 67 total game features. This leads to a culmination of 81 total features used as inputs for the machine learning model.

After obtaining all the features, 'cumulative' game data (data incorporating all data from all the previous games played before a certain game) was generated. If a team plays its Nth game of the season, then the data processed for that game should be a cumulative mean of all game data from the last N-1 games played. A simple Python program was created to iterate through all game data of a team and produce cumulative data to replace the 'raw data' originally obtained.

One of the problems generated from this was that the cumulative data for the first few games were not fully representative of the team's actual strength that season. If a team is playing its very first game of the season, then there is no game data available as they have not played any previous games. This 'hole' in the data does not improve the model results and is easily fixed if cumulative data from the team's previous season is used to replace the cumulative data for the first games of the season. We came to the conclusion that the model worked best when we utilized the previous season's cumulative data for the first 75 games of the season, which is around 5 games played for each team. The 76th game would be the first to use cumulative data from only the games played in the current season, and the rest would follow. As there are normally 1,230 games played in an NBA season, this means that the first 6% of games are determined by team performance from the last season.

With the cumulative data produced, the final array of data that is passed into the model is the difference of the two teams' cumulative team data. We experimented with two ways to represent the data, which were the difference of the two teams' statistics and the ratio of both; we came to the conclusion that using a different of both teams' data was found to be better as it generates a list of net values. To get the difference, we subtract the statistics of the away team from the statistics of the home team. This indicates that if the net values are positive, the home team has higher numbers, and if the net values are negative, the away team has higher numbers.

As the features of the model are represented by the cumulative data, the target variable of the model is represented by the outcome of a game itself. A simple

program iterated through the "Schedules" section of Basketball Reference, and the outcomes were mapped to binary numbers, in which:

- 0: Home Team is the Winner
- 1: Away Team is the Winner

A total of 11 NBA seasons of data were saved, from the 2009-2010 season to the 2019-2020. The 2009-2010 season was solely used for generating cumulative data for the 2010-2011 season, which means it was not used for training. The last season (2019-2020) was only used for testing. This means that 9 seasons, from 2010-2011 to 2018-2019, were available for training, and 9 seasons, from 2011-2012 to 2019-2020, were available for testing.

3.2 Feature Selection

The 81 features were split into three separate groups: Team Standings, Team Data, and Starter Data. We used the Python library sci-kit learn to get the feature importance based on the models used for training, according to the library's permutation importance method. After running through every machine learning method on all 81 features, the permutation importance method was run on the most efficient model, which returned a list of sorted features, from most to least important in units of importance (0 - 1) and importance share range (0 - 0.1). Table X displays the 10 most important features from this list:

Feature	Importance Share	Importance Share Range (+/-)
sBPM	0.157	0.014
sORtg	0.018	0.008
tDRtg	0.017	0.006
tSTL	0.016	0.005
tBLK%	0.015	0.004
sORB	0.011	0.003
t3PA	0.011	0.005
tTOV	0.010	0.005
tFTA	0.010	0.004
sDRB	0.010	0.004

When training machine learning models on solely these 10 features, no significant change in performance was observed, but a 0.5% increase in accuracy was obtained instead.

We experimented with different groups of features with the purpose of detecting variables with no correlation to the game outcomes. Several randomized groups of N number of features were used to train models and return the least important features. In the end, we discarded the following Team Standings features:

- Overall Wins
- Home Win Percentage
- Conference Win Percentage
- Division Wins
- Division Losses
- Last 10 Losses
- Last 10 Win Percentage

When it came to the Team Data and Starter Data, we ended up adding every feature except for Team Personal Fouls (tPF) and Starter Personal Fouls (sPF). We realized that they were the only features that seemed to have no correlation to the game outcomes, as having more of less fouls does not affect the play of the team.

Overall, we shortened the length of the list of features from 81 to 72, discarding 9 features in the process.

3.3 Model

For this project we utilized five different machine learning models: Linear Regression, Logistic Regression, Support Vector Machine Classifier (SVC), Random Forest Classifier, and Multi-Layer Perceptron. We have simulated these models through sci-kit learn.

All of the models are passed training data from a certain number of NBA seasons. We observed that the model generated its best results if it was passed 4 NBA seasons. Obviously, since a team gets better or worse in chronological order, the previous 4 seasons would be passed, and not 4 random seasons. For example, if we wanted to generate predictions for the 2015-2016 season, we would pass training data from the 2011-2012 season to the 2014-2015 season.

Linear Regression is a ubiquitously used algorithm that is mainly used for regression problems, where the model predicts a value instead of a class. It is utilized for predicting real-valued numbers (applications include stock price predictions, weather forecast, number of calories burned after exercise, etc.) Since the model must return either a 0 or a 1 to indicate the winner, a threshold variable was created to determine the outcome, where:

- if Y ¡ *THRESH*, Y = 0
- if Y *THRESH*, Y = 1

The threshold variable varies depending on the season that the algorithm is testing on. It rests around the 0.5 mark, with the lowest threshold being 0.488 for 2019 and highest being 0.564 for 2018.

Logistic Regression is a widely used classification algorithm that uses a sigmoid graph rather than a linear graph to sort out an outcome. Like Linear Regression, Logistic Regression predicts a real-valued number, but based on a certain threshold or thresholds, it returns a class (a value representing a group of outcomes) instead of a number.

Support Vector Machines (SVMs) focus on maximizing the margins of the classification boundary to focus on the points closest to the boundary. Unlike Logistic Regression, which treats all points equally, SVMs ignore the points farthest from the decision boundary, potentially producing more accurate results than other algorithms.

Random Forest is an ensemble algorithm that uses several decision trees to produce a majority vote outcome. It is great for classification as a 'committee' of trees decides the target variable, and not by a graph or decision boundary margins.

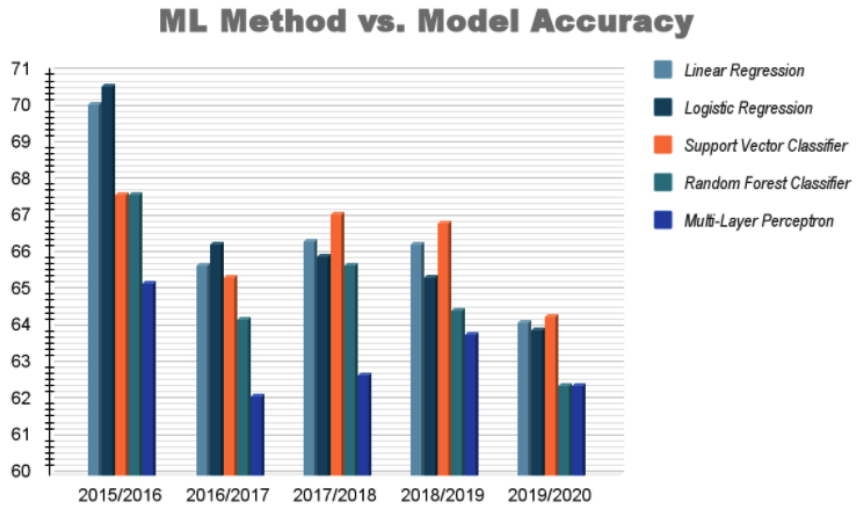
Lastly, Multi-Layer Perceptrons (MLP) were used as well. It is a neural network that requires different layers of nodes to determine the class of the feature variable. The number of layers and the number of nodes for each layer are mutable and the model accuracy can vary based on them.

4 Results

The results were obtained from training on four NBA seasons of data, and testing on one season. We obtained values from the last five seasons of data (2015-2016 to 2019-2020), through all of the five machine learning models explained above. The following table displays the accuracies of every model for each test season and method:

Model Accuracy	Linear Regression	Logistic Regression	Support Vector Classifier	Random Forest Classifier	Multi-Layer Perceptron	Mean
2015-2016	70.081	70.569	67.642	67.642	65.203	68.227
2016-2017	65.691	66.260	65.366	64.228	62.114	64.732
2017-2018	66.341	65.935	67.073	65.691	62.683	65.545
2018-2019	66.260	65.366	66.829	64.472	63.821	65.350
2019-2020	64.117	63.928	64.284	62.416	62.416	63.432
Mean	66.498	66.412	66.239	64.890	63.248	65.457

Here is a graphical representation of the results listed above:



Where the X-axis represents the season, and the Y-axis represents the model accuracy.

From the table and graph above, it is concluded that Linear Regression yielded the most accuracy out of the other machine learning models used. Logistic Regression and Support Vector Classifiers were close runner-ups, and Random Forest Classifier and Multi-Layer Perceptron gave the worst results. The most accurate result was testing Linear Regression on the 2015-2016 season, which produced a 70.569% accuracy.

Figure 8 displays a graphical representation of the 2015-2016 Linear Regression model, comparing class probability to the difference in points from both teams in a game during this season.

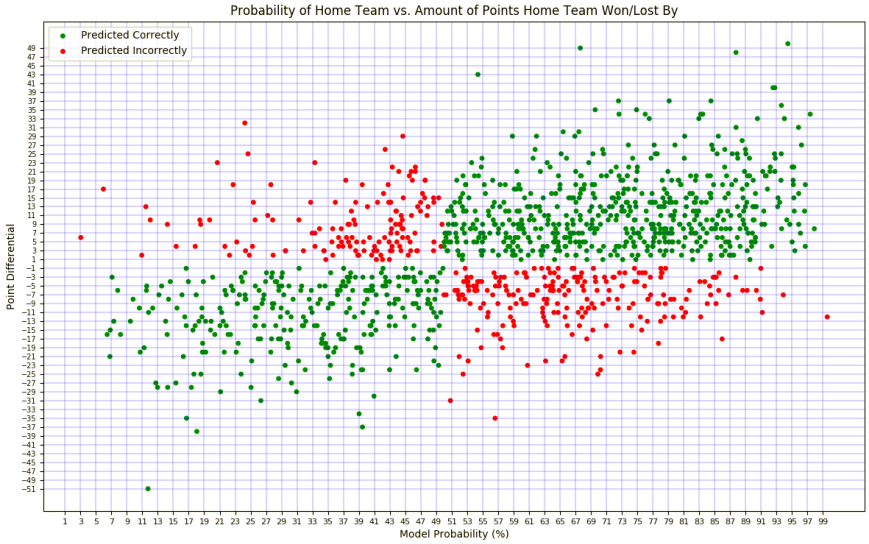


Figure 7: 2015-2016 Linear Regression Model Representation

From the graph it can be concluded that there is a positive correlation between the Model Class Probability and the Point Differential between both teams. Most data points happen to be on the right side of the graph, indicating that the home team won most of the games played. While there are some outliers (for example, the right-most red point, indicating that the model predicted a ¿ 99% chance of the home team winning but they lost by a significant amount of points), most of the games were predicted with a fair probability.

Puranmalka achieved a highest accuracy of 73.45% with his Support Vector Classifier model, surpassing my highest Support Vector Classifier accuracy by nearly 6%. Beckler, Wong, and Papamichael were able to get a highest accuracy of 73% with their linear regression model, surpassing my highest linear regression model accuracy by almost 3%. Cao achieved a highest accuracy of 69.67% with his logistic regression model, falling short of my highest logistic regression model accuracy by 0.9%. Expert predictor models licensed by ESPN and NBA are able to predict 71-72% of games. Overall, my results come close to these other machine learning methods, while falling just 0.5-1% shy of the leading models.

5 Conclusion

The goal of this project was to use machine learning models to generate results capable of surpassing the commercial models utilized by major sports networks. We used a total of 5 machine learning methods, which were each given four NBA seasons of training data, and we obtained results by testing on the next NBA season. The highest accuracy achieved from this process was 70.569% from a Linear Regression model simulated on the 2015-2016 season. While other models may have surpassed this number through other techniques, our model ultimately passed many major models (as most of them lie within the 65%-72% mark), and lay a footstep away from the official brand models.

References

[1] McFarlane, Greg. *How The NBA Makes Money*. investopedia.com, 2020. https://www.investopedia.com/articles/personal-finance/071415/how-nba-makes-money.asp

[2] Yang, Yuanhao. *Predicting Regular Season Results of NBA Teams Based on Regression Analysis of Common Basketball Statistics*. University of California at Berkeley, 2015. https://www.stat.berkeley.edu/ aldous/Research/Ugrad/Stanley_Yang%20_Thesis.pdf

[3] Fang, Ruogu. *NBA Game Prediction based on Historical Data and Injuries*. http://dionny.github.io/NBAPredictions/website/

[4] Weiner, Josh. *Predicting the outcome of NBA games with Machine Learning*. towardsdatascience.com, 2021. https://towardsdatascience.com/predicting-the-outcome-of-nba-games-with-machine-learning-a810bb768f20

[5] Puranmalka, Keshav. *Modelling the NBA to Make Better Predictions*. Massachusetts Institute of Technology, 2012. https://dspace.mit.edu/bitstream/handle/1721.1/85464/870969496-MIT.pdf? sequence=2&isAllowed=y

[6] Cao, Chenjie. *Sports Data Mining Technology Used in Basketball Outcome Prediction*. Technological University Dublin, 2012. https://arrow.tudublin.ie/cgi/viewcontent.cgi?article=1040&context=scschcomdis

[7] Beckler, Matthew, Wong, Hongfei, and Papamichael, Michael. *NBA Oracle*. Carnegie Mellon University, 2009. https://www.mbeckler.org/coursework/2008-2009/10701_report.pdf.

[8] Basketball Reference, 2021. https://www.basketball-reference.com

6 Appendix A: Box Score Data Features

Abbreviation	Feature
MP	Minutes Played
FG	Field Goals
FGA	Field Goals Attempted
FG%	Field Goal Percentage
3P	3 Pointers
3PA	3 Pointers Attempted
3P%	3 Point Percentage
FT	Free Throws
FTA	Free Throws Attempted
FT%	Free Throw Percentage
ORB	Offensive Rebounds
DRB	Defensive Rebounds
TRB	Total Rebounds
AST	Assists
STL	Steals
BLK	Blocks
TOV	Turnovers
PF	Personal Fouls
PTS	Points
+/-	Plus/Minus
TS%	True Shooting Percentage
eFG%	Effective Field Goal Percentage
3PAr	3 Point Attempt Rate
FTr	Free Throw Rate
ORB%	Offensive Rebound Percentage
DRB%	Defensive Rebound Percentage
TRB%	Total Rebound Percentage
AST%	Assist Percentage
STL%	Steal Percentage
BLK%	Block Percentage
TOV%	Turnover Percentage
USG%	Usage Rate
ORtg	Offensive Rating
DRtg	Defensive Rating
BPM	Box Plus/Minus