Final Project Report

Group B5 IST 687

Summer 2018

Scott Snow
Jeffrey Kao
Kendra Osburn
Ben Schneider

TABLE OF CONTENTS

Project Background and Description

This project is an exercise in taking a large data set spread over time with many metrics and elements and using that to confirm or refute past analytical based decisions.

Project Scope and Context of this Analysis

The scope of this project covers NBA player data from 1978 to 2016 as well as the MVP award for each year and the NBA Championship team for each year. The context is to use analytical techniques that we know or learn during this course to address the business questions below. While this is a curiosity and we only use the data mentioned in acquisition, its most likely use case would be in various forms of sports betting or expansion team creation.

Business Questions

- Which player statistics seem to have the most impact on the selection that was made for regular season mvp
- Which team averages seem to have the most impact on the team that wins the Championship?
- Do any random or non-random samples from this data set suggest something contrary?
- Is there a sample of years where the statistics indicate, for either category, that the wrong player was chosen as MVP or there was an upset based on this data in the outcome of that year's finals?
- What similarities exist between the players or teams in those samples?
- Can we draw any conclusions to what statistics about a player or team, might secure them an MVP award or NBA Championship over a more generally higher qualified player or team.
- Did each player earn their salary.
- How much of an impact did Salary expenditures have on season outcome.

Data Acquisition Process

NBA Champions by Year and NBA MVP by year are both available via Wikipedia. The bulk of the player data was obtained from https://data.world which seems to operate as a dataset networking site.

Data Selection Summary

From the original data set, we isolated a set of years and players that we wanted to focus on. Specifically, 2005-2015 and players that contributed at least 1% to their teams cumulative playing minutes. The reason we did only concentrated on the last 10 years of data was because of the state of the NBA and the style of game play nowadays. The playstyle last 10 years has definitely veered towards an analytics-based game, where many teams are adopting the 3pt and layup approach. This is because they have determined through analysis that to optimize the amount points in a game you should take more 3 pointers and layups.

Initial Quality Assessment

Our initial assessment was that this data was large, but we only wanted to concentrate on more modern NBA statistics. Some statistics were not collected in the older years. For instance, the oldest year 1978 did not even have 3 pointers because the NBA did not have 3 pointers yet. In addition, the 3 point line was moved around during the 90's which may have also skewed data. Our data set for 2016 was incomplete so we decided to eliminate that year as well. We ended up deciding on the 11 year subset of 2005-2015 player data.

Final Fields and Variables

The fields/columns we selected were based on data we thought we would need to answer our business questions. Since we knew that we wanted to compare player stats with the Championship team and the MVP, we chose relevant columns like True Salary, Win Shares, and others that most likely evaluate the player's ability to win and their true worth.

Data Dictionary

Index	Column Name	New Names	Definition
1	Year	Year	Year in the NBA
2	Tm	Team	Team Name
3	Player	Player Name	Player Name
4	Age	Age	Age of Player
5	G	Games Played	# of Games Played
6	MP	Minutes Played	# of minutes played

7	PER	Player Efficiency Rating	The player efficiency rating (PER) is famous rating from ESPN's John Hollinger's all-in-one basketball rating, which attempts to boil down all of a player's contributions into one number.
8	TS.	True Shooting %	True shooting percentage is an APBRmetrics statistic that measures a player's efficiency at shooting the ball.
9	X3PAr	3Pt Attempt Rate	Measure of what % of a player's shots come from long-distance.
10	FTr	FT Attempt Rate	Ratio of foul shots to field goal attempts
11	ORB.	Offensive Rebound %	The percentage of a team's offensive rebounds that a player has while on the court
12	DRB.	Defensive Rebound %	The percentage of a team's defensive rebounds that a player has while on the court
13	TRB.	Total Rebound %	The percentage of a total rebounds that a player has while on the court
14	AST.	Assist %	The percentage of a team's assists that a player has while on the court
15	BLK.	Block %	The percentage of a

			team's blocks that a player has while on the court
16	TOV.	Turnover %	The percentage of a team's turnovers that a player has while on the court
17	USG.	Usage %	The percentage of team plays used by a player when he is on the floor
18	OWS	Offensive Win Shares	Share of wins a player contributes to their team from offensive
19	DWS	Defensive Win Shares	Share of wins a player contributes to their team from defense
20	WS	Win Shares	Share of wins a player contributes to their team
21	WS.48	Win Shares Per 48 min	Share of wins a player contributes to their team per 48 min
22	OBPM	Offensive Box +/-	Offensive Box score-based metric for evaluating basketball players' quality and contribution to the team.
23	DBPM	Defensive Box +/-	Defensive Box score-based metric for evaluating basketball players' quality and contribution to the team.
24	BPM	Box +/-	Box score-based metric for evaluating basketball players' quality and contribution to the team.

25	VORP	Value over Replacement Player	**Value over Replacement Player (VORP) converts the BPM rate into an estimate of each player's overall contribution to the team, measured vs. what a theoretical "replacement player" would provide, where the "replacement player" is defined as a player on minimum salary or not a normal member of a team's rotation.**
26	OWS.48	Offensive Win Shares Per 48 min	**Share of wins a player contributes to their team from offensive per 48 min**
27	DWS.48	Defensive Win Shares Per 48 min	**Share of wins a player contributes to their team from defense per 48 min**
28	Shot.	% Shots of Team	Percentage of team's shots the player takes
29	Team.MP	Team Minutes Played	Total team minutes played for that season
30	Year.3PAr	Year 3Pt Attempt Rate	Measure of what % of a player's shots come from long-distance for the whole league that year
31	Team.TS.	Team True Shooting %	**Team true shooting percentage is an APBRmetrics statistic that measures a team's efficiency at shooting the ball.**
32	Tm.TS.W.O.Plyr	Team True Shooting % w/o Player	**Team true shooting percentage without the player**

33	TrueSalary	True Salary	NBA Player's Salary
34	Estimated.Position	Estimated Position	Their estimated position they would play based on their stats
35	Rounded.Position	Rounded Position	The position they play most of the time
36	Height	Height	Height of player
37	Weight	Weight	Weight of player
38	Yrs.Experience	Years Experience	Years experience in NBA
39	Championship Team	Championship Team	If the team player on won the championship
40	Runner Up	Runner Up	If the team player on was runner up
41	MVP	MVP	Most Valuable Player

Data Cleansing Summary

Some of players did not have salary or did not play enough to have relevant statistics so we removed those from our data set. Some of our data had to be converted to numeric values because they were strings.

In addition, some teams moved and changed their name, but the team remained the same. So we changed the names to the current team name so we could sort thru the data for each team. For instance, the Nets moved from New Jersey to Brooklyn so the team name changed.

Furthermore since we wanted to compare the players statistics to the MVP and Championship team we had to add that data as well.

Descriptive Statistics and Structure

After narrowing down our dataset we ended up with 4169 observations (rows) with 41 variables (columns).

Means

Age	26.863	Offensive Box +/-	-0.376
Games Played	57.208	Defensive Box +/-	0.052
Minutes Played	1450.856	Box +/-	-0.323
Player Efficiency Rating	14.313	Value over Replacement Player	0.865
True Shooting %	0.534	Offensive Win Shares Per 48min	0.048
3Pt Attempt Rate	0.229	Defensive Win Shares Per 48min	0.051
FT Attempt Rate	0.3	% Shots of Team	16.423
Offensive Rebound %	5.634	Team Minutes Played	19475.15
Defensive Rebound %	14.634	Year 3Pt Attempt Rate	0.228
True Rebound %	10.139	Team True Shooting %	0.538
Assist %	13.567	Team True Shooting % w/o Player	0.538
Block %	1.65	True Salary	5483873
Turnover %	13.48	Estimated Position	2.959
Usage %	18.924	Rounded Position	2.946
Offensive Win Shares	1.75	Height	78.883
Defensive Win Shares	1.506	Weight	218.746
Win Shares	3.256	Years Experience	5.118
Win Shares Per 48min	0.099		

Medians

Age	26	Offensive Box +/-	-0.5
Games Played	63	Defensive Box +/-	0

Minutes Played	1411	Box +/-	-0.6
Player Efficiency Rating	14	Value over Replacement Player	0.4
True Shooting %	0.534	Offensive Win Shares Per 48min	0.047
3Pt Attempt Rate	0.216	Defensive Win Shares Per 48min	0.049
FT Attempt Rate	0.276	% Shots of Team	16.2
Offensive Rebound %	4.3	Team Minutes Played	19827
Defensive Rebound %	13.6	Year 3Pt Attempt Rate	0.222
True Rebound %	9.2	Team True Shooting %	0.536
Assist %	10.5	Team True Shooting % w/o Player	0.537
Block %	1.1	True Salary	3.00E+06
Turnover %	13	Estimated Position	3
Usage %	18.7	Rounded Position	3
Offensive Win Shares	1.1	Height	79
Defensive Win Shares	1.2	Weight	220
Win Shares	2.4	Years Experience	4
Win Shares Per 48min	0.096		

Structure Output

```
'data.frame':	4169 obs. of  41 variables:
 $ Year                          : int  2005 2005 2005 2005 2005 2005 2005 2005 2005 2005 ...
 $ Team                          : Factor w/ 30 levels "ATL","BOS","BRK",..: 6 20 30 24 11 23 24 18 26 21 ...
 $ Player Name                   : Factor w/ 2828 levels "0","A.C. Green",..: 1670 2491 999 1345 2663 67
2440 1558 1908 2182 ...
 $ Age                           : int  20 27 23 23 25 29 26 28 26 29 ...
 $ Games Played                  : int  80 82 80 82 78 75 81 82 80 78 ...
 $ Minutes Played                : int  3388 3281 3274 3240 3182 3174 3146 3121 3084 3064 ...
 $ Player Efficiency Rating      : num  25.7 21.9 21.3 15.1 22.9 23.2 21.7 28.2 19.2 20.9 ...
 $ True Shooting %               : num  0.554 0.575 0.565 0.556 0.526 0.532 0.556 0.567 0.543 0.555 ...
 $ 3Pt Attempt Rate              : num  0.183 0.248 0.369 0.314 0.262 0.186 0.265 0.018 0.288 0.372 ...
 $ FT Attempt Rate               : num  0.378 0.42 0.42 0.153 0.336 0.432 0.213 0.404 0.327 0.286 ...
 $ Offensive Rebound %           : num  3.8 1.8 2.8 4.2 2.7 1.8 8.4 9.5 2.8 3.2 ...
 $ Defensive Rebound %           : num  17 7.1 10.5 9.5 14.8 8.9 22 30.2 9.2 10.7 ...
 $ Total Rebound %               : num  10.2 4.4 6.5 7 8.9 5.3 15.5 20.3 6 6.9 ...
 $ Assist %                      : num  32.9 36 22.9 13.2 28.6 37.6 7.5 27.1 28.1 18.1 ...
```

```
 $ Block %                        : num  1.1 0.1 0.5 0.5 1.3 0.2 2.5 2.6 0.7 0.1 ...
 $ Turnover %                     : num  11.8 13.1 11.8 10.5 9.5 13.7 8.1 12.2 12.3 9.2 ...
 $ Usage %                        : num  29.7 24.8 27.3 19 31.2 35 21.3 27.1 23.8 28 ...
 $ Offensive Win Shares           : num  9.7 10.4 9.2 5.8 6.5 5.3 7.3 10.1 6.6 9.7 ...
 $ Defensive Win Shares           : num  4.6 1.3 2.3 1.7 5.4 3.7 5.2 6 1.9 1 ...
 $ Win Shares                     : num  14.3 11.7 11.5 7.6 12 9 12.5 16.1 8.5 10.7 ...
 $ Win Shares Per 48min           : num  0.203 0.171 0.169 0.112 0.18 0.136 0.191 0.248 0.133 0.168 ...
 $ Offensive Box +/-              : num  6.7 5.2 5.2 2.4 4.9 4.9 2.6 5.2 3.4 5.4 ...
 $ Defensive Box +/-              : num  1 -2.2 -1.5 -0.1 1.6 -0.6 2 3.7 -1.1 -1.8 ...
 $ Box +/-                        : num  7.8 3 3.7 2.3 6.5 4.2 4.6 8.9 2.2 3.6 ...
 $ Value over Replacement Player  : num  8.3 4.1 4.7 3.6 6.8 5 5.3 8.6 3.3 4.3 ...
 $ Offensive Win Shares Per 48min : num  0.138 0.152 0.135 0.086 0.098 0.08 0.112 0.156 0.103 0.152
...
 $ Defensive Win Shares Per 48min : num  0.065 0.019 0.034 0.026 0.082 0.056 0.079 0.092 0.03 0.016
...
 $ % Shots of Team                : num  26.2 21.6 24.1 17 28.2 30.2 19.6 23.8 20.9 25.4 ...
 $ Team Minutes Played            : int  19855 19880 19780 19780 19855 19855 19780 19755 19855 19755
...
 $ Year 3Pt Attempt Rate          : num  0.196 0.196 0.196 0.196 0.196 0.196 0.196 0.196 0.196 0.196 ...
 $ Team True Shooting %           : num  0.518 0.532 0.523 0.571 0.535 0.528 0.571 0.534 0.541 0.546 ...
 $ Team True Shooting % w/o Player: num  0.505 0.52 0.51 0.573 0.538 0.526 0.574 0.525 0.541 0.543
...
 $ True Salary                    : num  27300000 16900000 18700000 16900000 24900000 19600000
21500000 28700000 15600000 18500000 ...
 $ Estimated Position             : num  2.8 1 1.2 2.9 2.4 1 3.5 4.2 1 1.9 ...
 $ Rounded Position               : num  3 1 1 3 2 1 3 4 1 2 ...
 $ Height                         : num  80 74 75 79 80 72 79 83 73 77 ...
 $ Weight                         : num  240 180 191 240 210 166 220 220 190 205 ...
 $ Years Experience               : num  1 8 3 3 7 8 5 9 6 8 ...
 $ Championship Team              : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Runner Up                      : num  0 0 0 0 0 0 0 0 0 0 ...
 $ MVP                            : num  0 0 0 0 0 0 0 0 0 0 …
```

Interesting findings

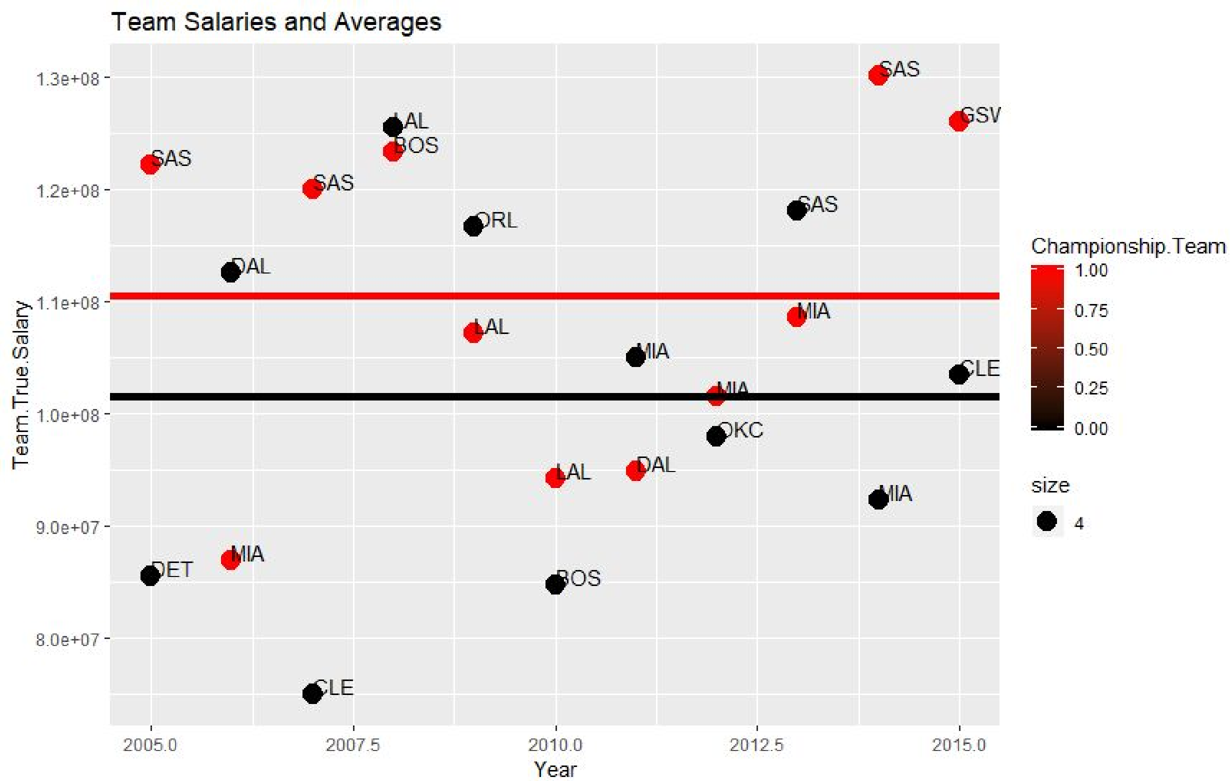
By glancing through the data we found that all in all, the best player in the last 10 years was none other than Lebron James, who has won the most championships and MVPs in the last 10 years.

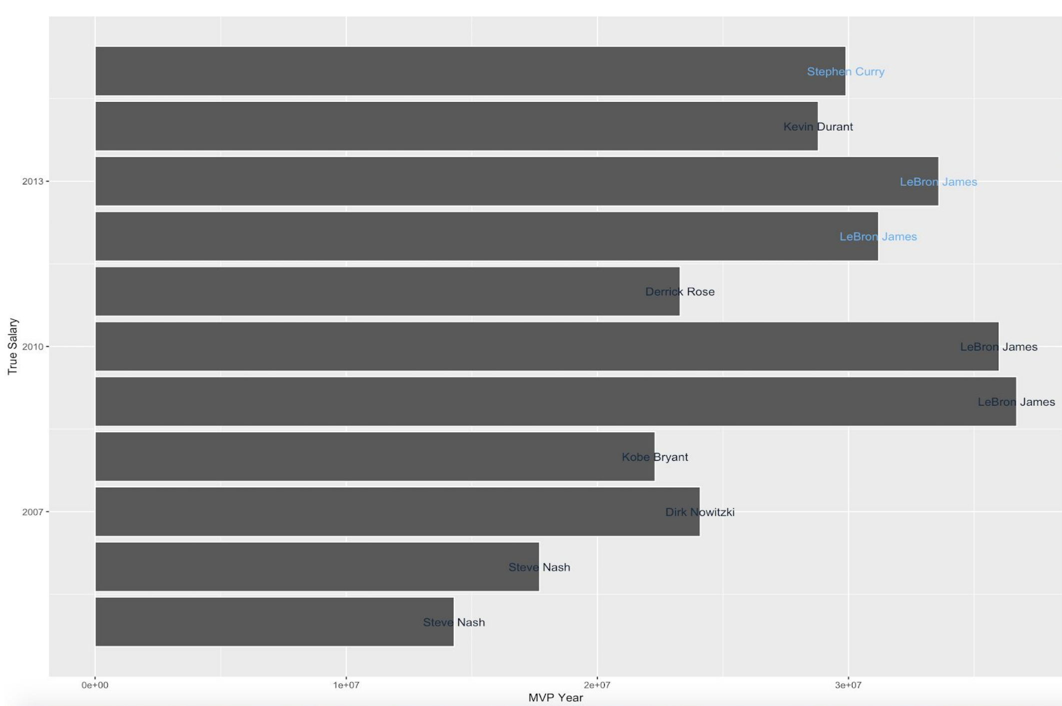
Based on the information, we would conclude that the 2010 Finals win by the Lakers was an upset based on our tests of that match-up based on our data.

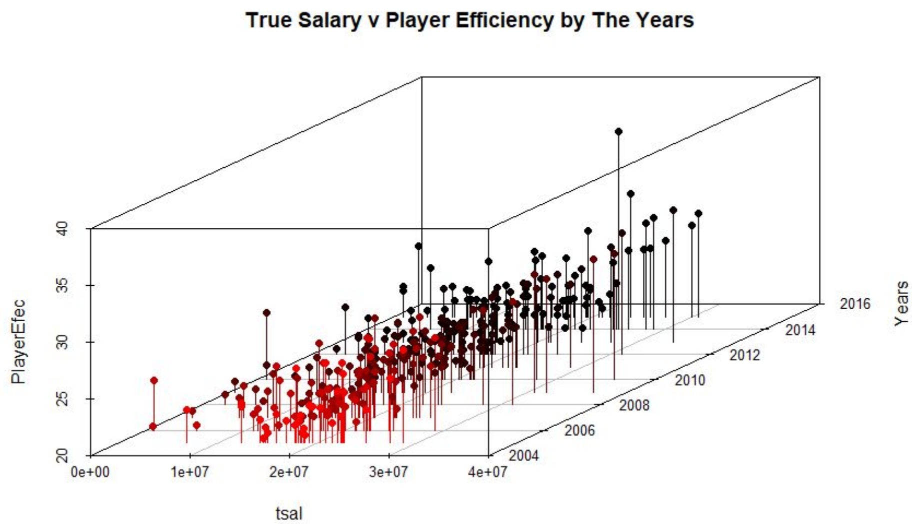
Our modeling efforts for MVP showed that weight was more statistically significant than height in its impact.

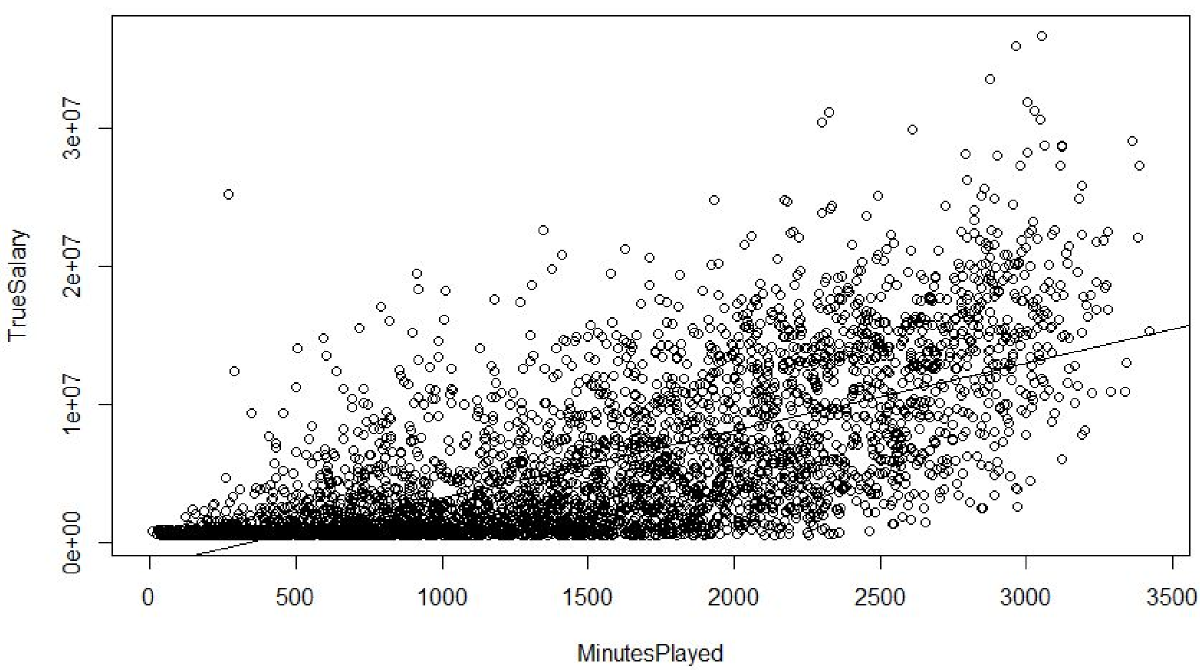
One of our visualizations indicates that Miami has underpaid their players compared to the number of appearances and wins. In any year that they won or were 2nd, their team spending was less than the average for champions.

Initial visualizations

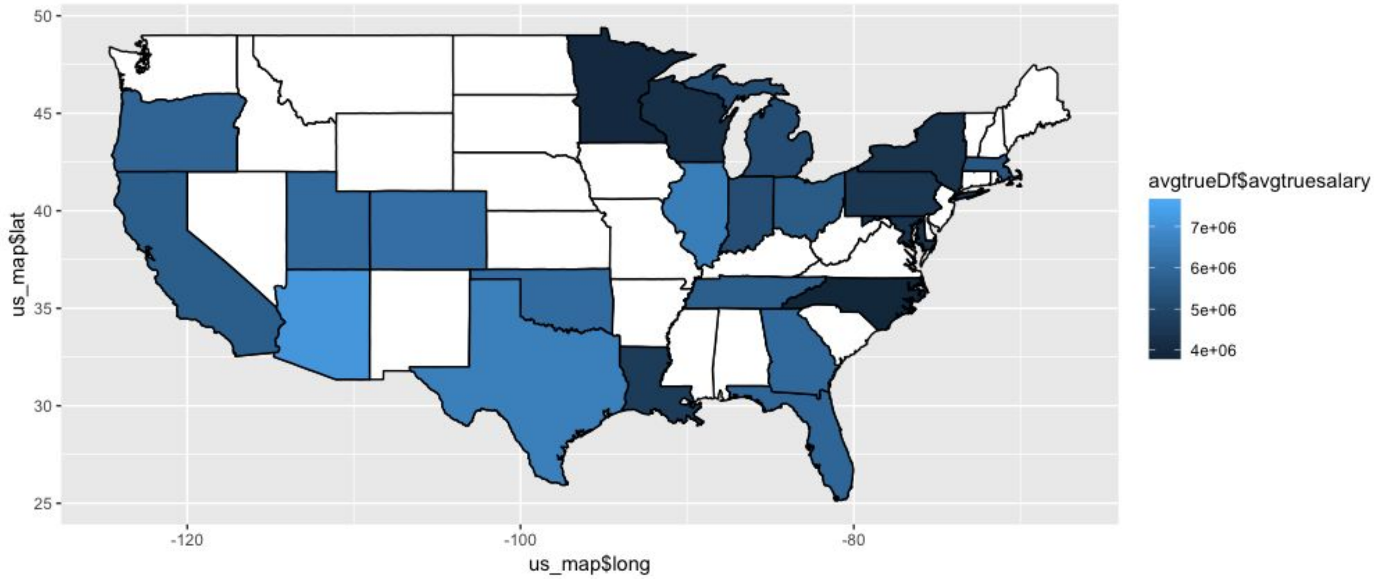

Team Salaries and Averages
Team.True.Salary
Year
1.3e+08
1.2e+08
1.1e+08
1.0e+08
9.0e+07
8.0e+07
2005.0
2007.5
2010.0
2012.5
2015.0
SAS
GSV
LAL
BOS
SAS
SAS
SAS
ORL
DAL
MIA
LAL
MIA
CLE
MIA
OKC
LAL
DAL
MIA
DET
MIA
BOS
CLE
Championship.Team
1.00
0.75
0.50
0.25
0.00
size
4


Stephen Curry
Kevin Durant
LeBron James
LeBron James
Derrick Rose
LeBron James
LeBron James
Kobe Bryant
Dirk Nowitzki
Steve Nash
Steve Nash
True Salary
2013
2010
2007
0e+00
1e+07
2e+07
3e+07
MVP Year


True Salary v Player Efficiency by The Years
PlayerEfec
20
25
30
35
40
0e+00
1e+07
2e+07
3e+07
4e+07
tsal
2004
2006
2008
2010
2012
2014
2016
Years


TrueSalary
0e+00
1e+07
2e+07
3e+07
0
500
1000
1500
2000
2500
3000
3500
MinutesPlayed

Avg True Salary per State



Summary of Techniques and Results

- Which player statistics seem to have the most impact on the selection that was made for regular season mvp

  Running a simple linear regression showed that the highest correlating factors for the MVP award were Minutes Played, Player Efficiency Rating, and interestingly True Salary. True Shooting % was less statistically significant than True Salary.

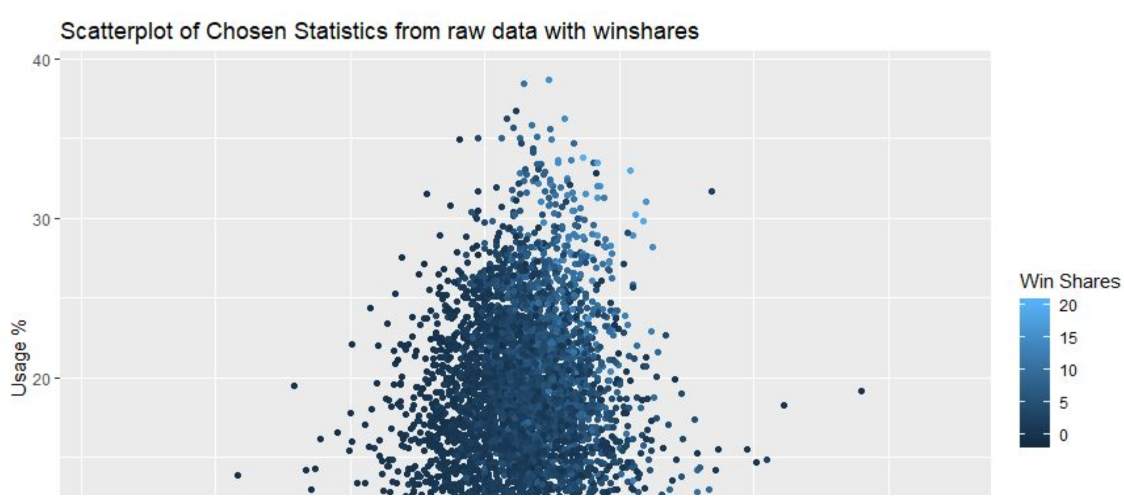
- Which team averages seem to have the most impact on the team that wins the Championship?

Running another linear regression showed that the highest correlating factors for NBA Championships were Team True Shooting and Team Minutes Played.

- Do any random or non-random samples from this data set suggest something contrary?

In summary, random sampling does not suggest conflicting correlating factors. For non random sampling, we looked at the teams in the finals. Some of the matchups did show contradicting circumstances, the explanations for which are below.

We choose samples to answer some of our questions because it allowed us to generate random teams to test the reliability of our correlation models. In order to do so, we did have to add an



“Average Player” row for each year so teams with less than the number of players in the sample would get the correct players. From there, we compared the samples generated to the generic players. The graphs below for True Shooting % and Usage % Indicating win shares is quite similar for the sample groupings and the players in general. The bar chart shows that having a single player in a sample team was most likely for a randomly generated distribution and that for players

    - Out of 1000 random teams, 9 Teams had the year’s MVP but were not in the top 5% in Win Shares. This is out of 31 Teams with the years MVP on them.
    - Out of 1000 random teams, 3 Teams had the seasons mvp and were not finalists

- Is there a sample of years where the statistics indicate, for either category, that the wrong player was chosen as MVP or there was an upset based on this data in the outcome of that year’s finals?
    - A simple selection of our data shows only 4 years where the regular season MVP made it to the finals
        - This indicates that being an MVP is not even a strong indicator of making it to the finals at least in this time frame
    - The MVP test was fairly simple. Comparing the teams in the finals took slightly more effort. Essentially, We chose True Shooting% and Usage% again along with Player Efficiency Rating. From their, we ran t.tests in R using the basic t.test(sample1, sample2) function. For each instance, we used two different sample sets, the first set contained all the players on a team for that year. The second set only compared the top five for each team with respect to Player Efficiency Rating. The concept here is that the five most efficient players would be the starters who receive the most playing time.
    - Using the results of the t.test functions, we looked at two sets of results. A set by comparison of means and a result by statistic significance($p<0.1$). Conclusions based on comparison of means were only made if one set (top5 or whole team) swept a year, or if the 2nd place teams had better averages in 4/6 categories.
    - Based on a comparison of means:
        - we would say that the winners of 2012 and 2014 were more of a group effort comparatively,
        - 2008 could be considered an upset
    - Based on a statistical significance of 0.10:
        - The 2007 Champions(Spurs) were better shooters at a statistically significant level. This is supported by their 4-0 victory in the series
        - The 2005 Champions(Spurs) were significantly better at using team plays
        - The 2010 Runner-Ups(Celtics) were in fact significantly better shooters. The fact that they still lost is explainable by the fact that the series went all 7 games and that overall their players were less efficient and their top

5 ran less team plays. Interestingly enough, the Lakers did pay their team more money in this time.

  - We did note that we did not gather data on the playing time for each player in the final. This project merely takes regular season data and compares the indicators from that data set to the outcome of each year's MVP selection and final match.

- What similarities exist between the players or teams in those samples?

  One factor that we could use in analysis in the future would be perhaps strength of conference. Because the data for all the teams is accumulated from a yearly perspective. The NBA is divided into the Western and Eastern conference and you play more games against your own conference. If let say for instance the East is top-heavy (very few elite teams), then the stats accumulated will be theoretically be better because of inferior competition. On the other hand if many teams in a conference were evenly matched this could skew some stats as well.

  However, our data shows that the 2008 Championship was mostly considered an upset because statistically the Celtics were a poorer team. This possibly can be attributed to the fact that the Celtics put a new superstar roster that year and they needed time to gel together and work better as a team. Needless to say this was the only time within our selected dataset where the team with the MVP that made the finals did not win the championship. Because our statistics are only for the whole season it does not take into account improvement in the latter half of a season. For instance some teams struggle in the beginning only to become dominant teams towards the end of season. This is especially the case with teams that assemble new rosters or acquire new superstar players.

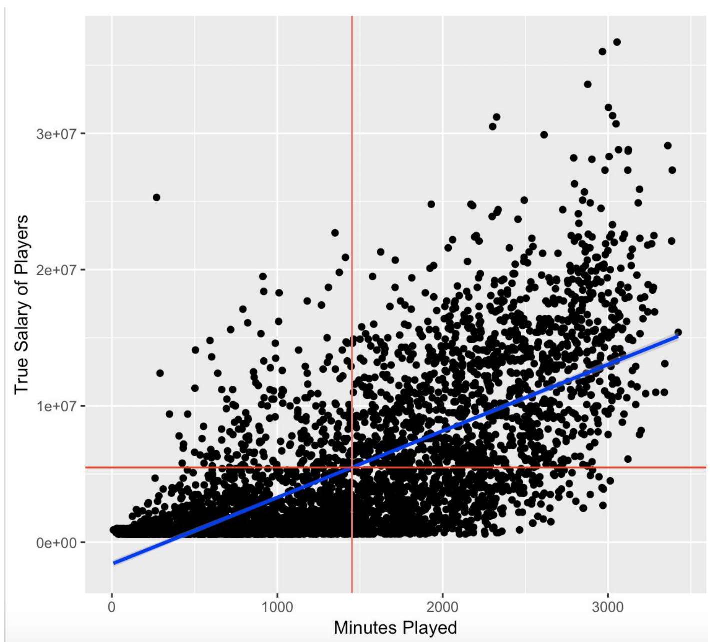
- Can we draw any conclusions to what statistics about a player or team, might secure them an MVP award or NBA Championship over a more generally higher qualified player or team?

  The best overlying fields to help determine MVP was Player Efficiency rating and win shares. These two fields were consistently close in determining the the MVP of the league. However the player needed to play enough games in order to be able to win these awards. For instance the highest player efficiency rating in our dataset did not win MVP in that year because he only played 8 games. Win shares was also a good statistics that helped determine whether or not a player deserved MVP. However, from personal NBA knowledge the record of the team is also a large contributing factor to the MVP race. We did not have a field for this, but if we wanted to do further analysis this would be something that we would add.

Only 27.27% of time was the MVP on the Championship team. This stat only goes up 36.3% if you factor in being the runner up as well. This makes sense because basketball is a team game and having the MVP is a correlating factor in winning a championship, but it is definitely not the most important.

One field that could factor into whether a player or team might secure a MVP/NBA Championship over a player with higher PER or Win Share was Team True Shooting % w/o Player. In most cases, when the winner who won over the higher qualified player like a player with higher win share or PER had a higher Team True Shooting % w/o Player. For instance in 2015, James Harden had a higher win share and Anthony Davis had a higher PER, but the MVP winner Steph Curry had a higher Team True Shooting % w/o Player. This goes to show that teammates and teamwork are just as important as individual talent when going for the MVP award.

- Did each player earn their salary.



On top, the scatter plot graphs each players played minutes against their contracted salary. The data reveals the following information:

- Plot Arrangement
- The linear best fit line is represented by the blue diagonal
- The minutes played and true salary means are represented by the intersecting red lines

- Analysis
  - Points located within the left quadrant of the intercept are players representing inefficient spending by organizations
  - Only points in the right quadrant above the best fit line represent appropriate

team salary contracts

```
Residual standard error: 4215000 on 4167 degrees of freedom
Multiple R-squared:  0.476,	Adjusted R-squared:  0.4759
F-statistic:  3785 on 1 and 4167 DF,  p-value: < 2.2e-16
```

With the aim to predict player's salaries based on their minutes played there's a average to low relationship. Thus, exemplifying the fact of organizations operating too frequently within the left quadrant. Players have extensive/expensive contracts for minimal team representation and playing contributions.

The low P-Value does support the statistical notation that there is a relationship between these variables. Yet, the data supports a hypothesis of continual randomized allocation of salary funds while measuring player's game activity because of an average R-Squared value.

- How much of an impact did Salary expenditures have on season outcome.

	Championship Team	Team	Year	Team True Salary
1	1	MIA	2006	87000000
2	1	LAL	2010	94300000
3	1	DAL	2011	94900000
4	1	MIA	2012	101600000
5	1	LAL	2009	107200000
6	1	MIA	2013	108600000
7	1	SAS	2007	120000000
8	1	SAS	2005	122200000
9	1	BOS	2008	123400000
10	1	GSW	2015	126100000
11	1	SAS	2014	130200000

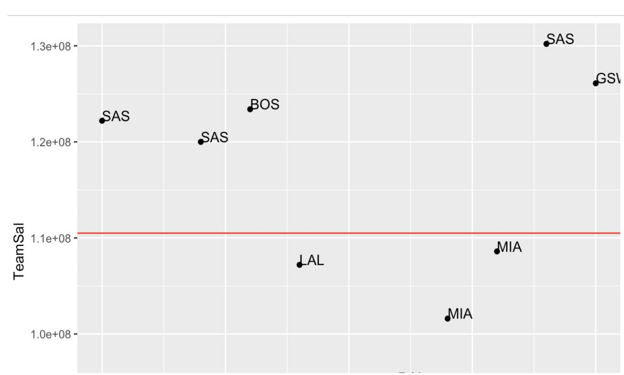
This dataframe portrays the expenditures the NBA title winning teams took. The data is arranged by ascending True Salary, which is grouped to illustrate their year team salary.

A T-Test between the True Salaries of 1st place and 2nd place does reveal a 80% confidence that a higher team salary is significant in winning the finals

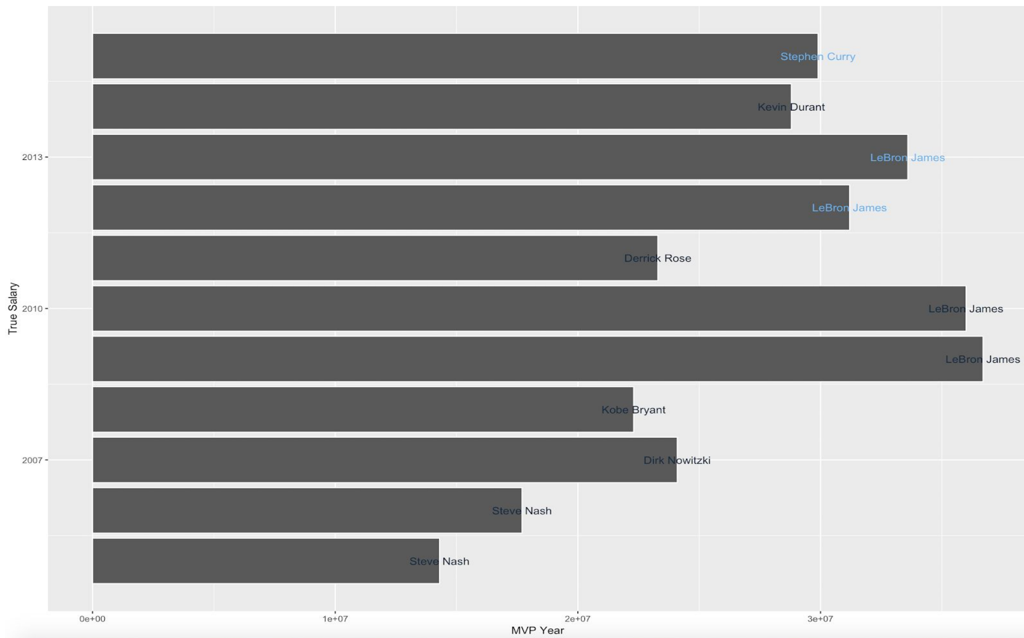
The scatterplot shows the recent NBA title champions yearly spending against each other.

- ❖ Plot Arrangement
  - ➢ The mean Team Salary is represented by the horizontal line

- ❖ Analysis:
  - ➢ The San Antonio Spurs spent the most yearly salary to win their titles
  - ➢ By win volume, the Miami Heat spent the least yearly salary to win their titles
    - ➢ Boston, at the time, had overspent to win their championship because they have yet to reclaim a title while positioning themselves



well above the mean



Presented here are the MVP recipients mapped against their salaries.

- Light blue labeled players won the NBA title the year of their MVP accolade
- Analysis
  - Lebron James was the highest paid MVP. Through the years of being the top paid, he was 1 for 3 in championship titles
  - MVPs are considered to be the players most likely to have a “max contract”. However, only 3 out of 11 teams won a championship

Overall Interpretation of Results

- Overall we see that in general, the team with the better players will win.
- How much you spend paying players does matter or it at leasts correlate to winning
- It is possible to win with strong team play over a more accomplished group of players
- Regular season MVP is not a guarantee of winning or even making it to the finals
- Most players aside from a few front people are interchangeable in a teams roster according to the data

Actionable Insights

Coaches

- Place more emphasis on team play
- Ensure that players are paid appropriate to their contribution

Oddsmakers

- Don’t set as much store by which team has the MVP later on in the playoff season
- Pay attention to team qualities that have potential to overcome strong players in a close series such as team usage

References

1. Initial csv from:
2. MVP and championship information from “http://www.wikipedia.com” search for “NBA MVP” and “NBA Finals”. We self-constructed the file “smalltable.csv”
3. Partial definition retrieval from https://www.basketball-reference.com/about/glossary.html

Appendix - R Code

```
#Final Project IST 687 Summer 2018
#Scott Snow, Jeffrey Kao, Kendra Osburn, Benjamin Schneider

#ensure required libraries

EnsurePackage <- function(x) {
  x <- as.character(x)

  if (!require(x, character.only=TRUE)) {
    install.packages(pkgs=x, repos="http://cran.r-project.org")
    require(x,character.only=TRUE)

  }
}
# get packages
EnsurePackage("dplyr")
EnsurePackage("sqldf")
EnsurePackage("ggplot2")
EnsurePackage("kernlab")
EnsurePackage("gdata")
EnsurePackage("ggmap")
EnsurePackage("scatterplot3d")
library(dplyr)
library(sqldf)
library(ggplot2)
library(kernlab)
library(gdata)
library(ggmap)
library(scatterplot3d)
#-------------------------------------------------------------------------------
# get csv
urlToRead <-
"https://trello-attachments.s3.amazonaws.com/5b6dc416cb77f61d2d3919d7/5b6dc416e8e0a46275da9
2ef/24742472ff3d38988c48f004878be4d5/NBASeasonData1978-2016.csv"
# Read CSV
csv <- read.csv(url(urlToRead), header=TRUE, sep=",")
```

```
# Keep only rows from 2005-2015
nba <- csv[11083:16859, ]
# Convert to dataframe
nbadf <- as.data.frame(nba)
# Select columns needed
nbadfselected <- nbadf %>% select("Year",
                  "Tm",
                  "Player",
                  "Age",
                  "G",
                  "MP",
                  "PER",
                  "TS.",
                  "X3PAr",
                  "FTr",
                  "ORB.",
                  "DRB.",
                  "TRB.",
                  "AST.",
                  "BLK.",
                  "TOV.",
                  "USG.",
                  "OWS",
                  "DWS",
                  "WS",
                  "WS.48",
                  "OBPM",
                  "DBPM",
                  "BPM",
                  "VORP",
                  "OWS.48",
                  "DWS.48",
                  "Shot.",
                  "Team.MP",
                  "Year.3PAr",
                  "Team.TS.",
                  "Tm.TS.W.O.Plyr",
                  "TrueSalary",
                  "Estimated.Position",
                  "Rounded.Position",
                  "Height",
                  "Weight",
                  "Yrs.Experience")

# Rename columns
colnames(nbadfselected) <- c("Year", "Team", "Player Name",
                    "Age", "Games Played", "Minutes Played",
                    "Player Efficiency Rating", "True Shooting %",
```

```
                         "3Pt Attempt Rate", "FT Attempt Rate",
                         "Offensive Rebound %", "Defensive Rebound %",
                         "True Rebound %", "Assist %", "Block %",
                         "Turnover %", "Usage %", "Offensive Win Shares",
                         "Defensive Win Shares", "Win Shares",
                         "Win Shares Per 48min", "Offensive Box +/-",
                         "Defensive Box +/-", "Box +/-",
                         "Value over Replacement Player", "Offensive Win Shares Per 48min",
                         "Defensive Win Shares Per 48min", "% Shots of Team",
                         "Team Minutes Played", "Year 3Pt Attempt Rate",
                         "Team True Shooting %", "Team True Shooting % w/o Player",
                         "True Salary","Estimated Position",
                         "Rounded Position", "Height", "Weight", "Years Experience")

# remove blank True salaries
nbadfselected <- nbadfselected %>% filter(`True Salary`!="")
# should get 4169 obs. of 38 variables.

# replaces team names with their current team name
# i.e. a team changed their name or moved locations or both
tempteamnames <- as.character(nbadfselected$Team)
for (i in seq(1:length(tempteamnames))) {
  if(tempteamnames[i] == "SEA") {
    tempteamnames[i] <- "OKC"
  }
  if(tempteamnames[i] == "NJN") {
    tempteamnames[i] <- "BRK"
  }
  if(tempteamnames[i] == "CHA" || tempteamnames[i] == "CHH"){
    tempteamnames[i] <- "CHO"
  }
  if(tempteamnames[i] == "NOK" || tempteamnames[i] == "NOH"){
    tempteamnames[i] <- "NOP"
  }
}
nbadfselected$Team <- as.factor(tempteamnames)

#quick csv containing nba champs, runner ups and the years MVP
urlToRead2 <-
"https://trello-attachments.s3.amazonaws.com/5b6dc416cb77f61d2d3919d7/5b6dc416e8e0a46275da9
2ef/31892bb2bbc23a8c996972b0285f4434/smalltable.csv"
smallcsv <- read.csv(url(urlToRead2), header=TRUE, sep=",")

#creates new columns
nbadfselected$'Championship Team' <- NULL
nbadfselected$'Runner Up' <- NULL
nbadfselected$'MVP' <- NULL
```

```
# contains 1 if that player played in the finals or was the mvp respectively
for(j in seq(1:length(nbadfselected$Year))){
  nbadfselected$'Championship Team'[j] <- 0
  nbadfselected$'Runner Up'[j] <- 0
  nbadfselected$'MVP'[j] <- 0
  for (i in seq(1:length(smallcsv$Year))) {
    if(nbadfselected$Year[j] == smallcsv$Year[i]) {
      if(nbadfselected$'Team'[j] == smallcsv$NBA.Champion[i]){
        nbadfselected$'Championship Team'[j] <- 1
      }
      if(nbadfselected$'Team'[j] == smallcsv$NBA.Runner.Up[i]){
        nbadfselected$'Runner Up'[j] <- 1
      }
      if(nbadfselected$`Player Name`[j] == smallcsv$MVP[i])
      nbadfselected$'MVP'[j] <- 1
    }
  }
}

#convert True Salary to Numeric
temp <- as.character(nbadfselected$`True Salary`)
temp <- gsub("\\$", "", temp)
temp <- gsub(",", "", temp)
temp <- as.numeric(temp)
nbadfselected$`True Salary` <- temp

#convert both position columns, height, weight and years experience to numeric
nbadfselected$`Estimated Position` <- as.numeric(as.character(nbadfselected$`Estimated Position`))
nbadfselected$`Rounded Position` <- as.numeric(as.character(nbadfselected$`Rounded Position`))
nbadfselected$Height <- as.numeric(as.character(nbadfselected$Height))
nbadfselected$Weight <- as.numeric(as.character(nbadfselected$Weight))
nbadfselected$`Years Experience` <- as.numeric(as.character(nbadfselected$`Years Experience`))

#retrieval function that will be used later
getSTAT <- function(player, year, stat) {
  index <- which(match(nbadfselected$`Player Name`, player) == match(nbadfselected$`Year`, year))
  return(mean(nbadfselected[index, which(match(colnames(nbadfselected), stat) == 1)]))
}
nbadfselected[2148,]
#ensures the team and player are characters not factors
nbadfselected$`Player Name` <- as.character(nbadfselected$`Player Name`)
nbadfselected$Team <- as.character(nbadfselected$Team)

#creates an "average" replacement player to test a teams performance without a star player for each
year
years <- unique(nbadfselected$Year)
avgplayersdf <- data.frame()
```

```
for(j in years) {
  avgplayer <- c(j,"AVG","Average Player")
  for(i in 4:dim(nbadfselected)[2]) {
    avg <- as.numeric(sum(nbadfselected[nbadfselected$Year ==
j,i])/dim(nbadfselected[nbadfselected$Year == j,])[1])
    if(i > 38) {
      avg <- 0
    } else if (i > 35) {
      avg <- round(avg)
    } else if(i == 35) {
      avg <- 0
    }
    avgplayer <- c(avgplayer, avg)
  }
  avgplayersdf <- rbind.data.frame(avgplayersdf, as.numeric(avgplayer))
  #print(avgplayersdf)
}
colnames(avgplayersdf) <- colnames(nbadfselected)
avgplayersdf$Team <- "AVG"
avgplayersdf$`Player Name` <- "Average Player"
nbadfselected <- rbind.data.frame(nbadfselected, avgplayersdf)

#descriptive statistics
round(sapply(nbadfselected[,4:38], mean), digits=3)
round(sapply(nbadfselected[,4:38], median), digits=3)

#exporting cleaned data to csv
setwd("~/Desktop")
write.csv(nbadfselected,'nbadata.csv')
#----------------------------------------------------------------------
#str(nbadfselected)
#calculates the average teamsize(30 teams in the nba)
teamsize <- floor(mean(as.numeric(unlist(sqldf("SELECT Year, COUNT(Team)/30 FROM nbadfselected
GROUP BY Year ")[2]))))
totalteams <- 30

getfinalsample <- function(samplesize, stat1, stat2) {
# gets the number of
  fantasyyears <- replicate(samplesize, sample(years, 1), simplify=TRUE)
  fantasyteams <- data.frame(nextCol=vector(length = 11))
  for(i in 1:samplesize) {
    thisyear <- fantasyyears[i]
    thisteam <- NULL
    positionbins <- unlist(fn$sqldf("SELECT COUNT([Rounded Position]) AS Bin FROM nbadfselected
WHERE Year = $thisyear GROUP BY [Rounded Position]"))
    for(j in 1:5) {
      positionbins[j] <- round(positionbins[j]/totalteams)
      yearsplayers <- nbadfselected[nbadfselected$Year == fantasyyears[i],]
```

```
    thisteam <- c(thisteam, as.character(sample(yearsplayers[yearsplayers$`Rounded Position` ==
j,]$`Player Name`, positionbins[j])))
   }
   if(dim(fantasyteams)[1] < length(thisteam)) {
    thisteam <- thisteam[1:dim(fantasyteams)[1]]
   }
   while (dim(fantasyteams)[1] > length(thisteam)) {
    thisteam <- c(thisteam, "Average Player")
   }
   fantasyteams$nextCol <- thisteam
   colnames(fantasyteams)[i] <- as.character(fantasyyears[i])
  }

  fantasystats <- data.frame(row.names = sprintf("Team %d", seq(1:samplesize)))
  for(i in 1:ncol(fantasyteams)) {
   sumchampsorrun <- 0
   AVG1 <- 0
   AVG2 <- 0
  Totalwinshares <- 0
  mvpflag <- FALSE
  for(j in 1:nrow(fantasyteams)) {
   if(getSTAT(fantasyteams[j,i], colnames(fantasyteams)[i], "MVP") == 1) {
    mvpflag <- TRUE
   }
   sumchampsorrun <- sumchampsorrun + ceiling(getSTAT(fantasyteams[j,i], colnames(fantasyteams)[i],
"Championship Team")) + ceiling(getSTAT(fantasyteams[j,i], colnames(fantasyteams)[i], "Runner Up"))
   AVG1 <- AVG1 + getSTAT(fantasyteams[j,i], colnames(fantasyteams)[i], stat1)
   AVG2 <- AVG2 + getSTAT(fantasyteams[j,i], colnames(fantasyteams)[i], stat2)
   Totalwinshares <- Totalwinshares + getSTAT(fantasyteams[j,i], colnames(fantasyteams)[i], "Win
Shares")
  }
  fantasystats <- rbind.data.frame(fantasystats, c(as.integer(colnames(fantasyteams)[i]), mvpflag,
sumchampsorrun, AVG1/nrow(fantasyteams), AVG2/nrow(fantasyteams), Totalwinshares))
  }
  colnames(fantasystats) <- c("Year", "Has MVP", "Finalist Total", sprintf("%s Average", stat1),
sprintf("%s Average", stat2), "Total Win Shares")
  return(fantasystats)
}

samplestats <- getfinalsample(1000, "True Shooting %", "Usage %")
colnames(nbadfselected)
sqldf("SELECT COUNT(Year) FROM samplestats WHERE [Has MVP] == 1")
percenttop <- quantile(samplestats$`Total Win Shares`, c(0.0, .90, 1))[2]
fn$sqldf("SELECT COUNT(Year) FROM samplestats WHERE [Has MVP] == 1 AND [Total Win Shares] >
$percenttop")

maxfinalists <- as.integer(sqldf("SELECT MAX([Finalist Total]) FROM samplestats"))
```

```
top5count <- c()
totcount <- c()
for (i in 0:maxfinalists) {
  top5count <- c(top5count, as.numeric(fn$sqldf("SELECT COUNT(Year) FROM samplestats WHERE
[Finalist Total] == $i AND [Total Win Shares] > $percenttop")))
  totcount <- c(totcount, as.numeric(fn$sqldf("SELECT COUNT(Year) FROM samplestats WHERE
[Finalist Total] == $i")))
}
totfins <- sort(unique(samplestats$`Finalist Total`))
data1plot <- data.frame(totfins, top5count, totcount)
plot1 <- ggplot(data1plot, aes(x=totfins)) + geom_col(aes(y=totcount, fill="Total Finalists"))
plot1 <- plot1 + geom_col(aes(y=top5count, fill="Top 10% of Win Shares")) + ggtitle("Summary of
Team's Total Finalists")
plot1 <- plot1 + geom_text(data=data1plot, aes(y=top5count, label = top5count), vjust=-1)
plot1 <- plot1 + geom_text(data=data1plot, aes(y=totcount, label = totcount), vjust=1)
plot1

data2plot <- data.frame(totfins, samplestats$`Total Win Shares`, samplestats$`True Shooting %
Average`, samplestats$`Usage % Average`)
colnames(data2plot) <- c("totfins", "winShares", "avgTS", "avgUSG")
plot2 <- ggplot(data2plot, aes(x=avgTS, y=avgUSG)) + geom_point(aes(size=totfins, color=winShares))
plot2 <- plot2 + ggtitle("Scatterplot of Chosen Statistics, With win shares")
plot2

plot3 <- ggplot(nbadfselected, aes(x=`True Shooting %`, y=`Usage %`)) + geom_point(aes(color=`Win
Shares`))
plot3 <- plot3 + ggtitle("Scatterplot of Chosen Statistics from raw data with winshares")
plot3

#-----------------------------------------------------------------------------------------------------

testdata1 <- nbadfselected[nbadfselected$MVP == 1, -4:-38]
sqldf("SELECT COUNT(Year) FROM testdata WHERE ([Championship Team] + [Runner Up]) == MVP")

#----------------------------------------------------------------------------------------------------

runnerups <- nbadfselected[nbadfselected$`Runner Up` == 1,]
champs <- nbadfselected[nbadfselected$`Championship Team` == 1,]

runstats <- function(year) {
  print("These tests includes all significant players of each team.")
  fiveR <- runnerups[runnerups$Year == year,]
  fiveC <- champs[champs$Year == year,]

  sample1.1 <- fiveR$`True Shooting %`
  sample1.2 <- fiveC$`True Shooting %`

  test1 <- t.test(sample1.1, sample1.2)
```

```
  result1 <- test1[[3]] < 0.1
  print(sprintf("This test is for True Shooting %% in year %d", year))
  print(sprintf("The p-value of the test is %f, Reject the null hypothesis: %d", test1[[3]], result1))
  print(sprintf("The means are %f and %f for Runner Up and Champ respectively. The champs were
better than the unner ups: %d", test1[[5]][1], test1[[5]][2], test1[[5]][1] < test1[[5]][2]))
  cat("\n")
  sample2.1 <- fiveR$`Usage %`
  sample2.2 <- fiveC$`Usage %`

  test2 <- t.test(sample2.1, sample2.2)
  result2 <- test2[[3]] < 0.1
  print(sprintf("This test is for Usage %% in year %d", year))
  print(sprintf("The p-value of the test is %f, Reject the null hypothesis: %d", test2[[3]], result2))
  print(sprintf("The means are %f and %f for Runner Up and Champ respectively. The champs were
better than the unner ups: %d", test2[[5]][1], test2[[5]][2], test2[[5]][1] < test2[[5]][2]))
  cat("\n")
  sample3.1 <- fiveR$`Player Efficiency Rating`
  sample3.2 <- fiveC$`Player Efficiency Rating`

  test3 <- t.test(sample3.1, sample3.2)
  result3 <- test3[[3]] < 0.1
  print(sprintf("This test is for Player Efficiency Rating in %d", year))
  print(sprintf("The p-value of the test is %f, Reject the null hypothesis: %d", test3[[3]], result3))
  print(sprintf("The means are %f and %f for Runner Up and Champ respectively. The champs were
better than the unner ups: %d", test3[[5]][1], test3[[5]][2], test3[[5]][1] < test3[[5]][2]))
  cat("\n")

  print("These tests only include the top 5 players of each team.")
  fiveRtop5 <- sqldf("SELECT * FROM fiveR ORDER BY -[Player Efficiency Rating] LIMIT 5")
  fiveCtop5 <- sqldf("SELECT * FROM fiveC ORDER BY -[Player Efficiency Rating] LIMIT 5")

  sample1.1 <- fiveRtop5$`True Shooting %`
  sample1.2 <- fiveCtop5$`True Shooting %`

  test1 <- t.test(sample1.1, sample1.2)
  result1 <- test1[[3]] < 0.1
  print(sprintf("This test is for True Shooting %% in year %d", year))
  print(sprintf("The p-value of the test is %f, Reject the null hypothesis: %d", test1[[3]], result1))
  print(sprintf("The means are %f and %f for Runner Up and Champ respectively. The champs were
better than the unner ups: %d", test1[[5]][1], test1[[5]][2], test1[[5]][1] < test1[[5]][2]))
  cat("\n")
  sample2.1 <- fiveRtop5$`Usage %`
  sample2.2 <- fiveCtop5$`Usage %`

  test2 <- t.test(sample2.1, sample2.2)
  result2 <- test2[[3]] < 0.1
  print(sprintf("This test is for Usage %% in year %d", year))
  print(sprintf("The p-value of the test is %f, Reject the null hypothesis: %d", test2[[3]], result2))
```

```
  print(sprintf("The means are %f and %f for Runner Up and Champ respectively. The champs were
better than the unner ups: %d", test2[[5]][1], test2[[5]][2], test2[[5]][1] < test2[[5]][2]))
  cat("\n")
  sample3.1 <- fiveRtop5$`Player Efficiency Rating`
  sample3.2 <- fiveCtop5$`Player Efficiency Rating`

  test3 <- t.test(sample3.1, sample3.2)
  result3 <- test3[[3]] < 0.1
  print(sprintf("This test is for Player Efficiency Rating in %d", year))
  print(sprintf("The p-value of the test is %f, Reject the null hypothesis: %d", test3[[3]], result3))
  print(sprintf("The means are %f and %f for Runner Up and Champ respectively. The champs were
better than the unner ups: %d", test3[[5]][1], test3[[5]][2], test3[[5]][1] < test3[[5]][2]))
  cat("\n")
}

for(i in years) {
  runstats(i)
}

#T-Test Analysis
#In the year 2010, The runner up were statistically better shooters
#In 2007 The top 5 players for the champions were statistically better shooters
#In 2005 The Usage % in 2005 for the top 5 matchup was statistically greater for the champions

#----------------------------------------------

#Average Comparison
#The PER for 2006 all players showed the runner ups having a higher average

#The TS% for 2008 all players showed the runner ups having a higher average
#The PER for 2008 all players showed the runner ups having a higher average
#The TS% for 2008 top 5 showed the runner ups having a higher average
#The PER for 2008 top 5 showed the runner ups having a higher average

#The TS% for 2009 top 5 showed the runner ups having a higher average
#The PER for 2009 top 5 showed the runner ups having a higher average

#The TS% for 2010 all players showed the runner ups having a higher average
#The USG% for 2010 all players showed the runner ups having a higher average
#The TS% for 2010 Top 5 showed the runner ups having a higher average

#The TS% for 2011 all players showed the runner ups having a higher average
#The USG% for 2011 Top 5 showed the runner ups having a higher average
#The PER for 2011 Top 5 showed the runner ups having a higher average

#2012 Top 5 Showed Runner Ups having higher statistics for all 3 measures

#The USG% for 2013 all players showed the runner ups having a higher average
```

```
#The TS% for 2014 all players showed the runner ups having a higher average
#2014 Top 5 Showed Runner Ups having higher statistics for all 3 measures

#The TS% for 2015 Top 5 showed the runner ups having a higher average

#-----------------------------------------------------------------------------------

samp1 <- sqldf("SELECT Year, [Team], SUM([True Salary]) AS [Team True Salary] FROM nbadfselected
WHERE [Runner Up] = 1 GROUP BY Year, Team ORDER BY Year")
samp2 <- sqldf("SELECT Year, [Team], SUM([True Salary]) AS [Team True Salary] FROM nbadfselected
WHERE [Championship Team] = 1 GROUP BY Year, Team ORDER BY Year")

mean(samp1$`Team True Salary`)
mean(samp2$`Team True Salary`)

t.test(samp1$`Team True Salary`, samp2$`Team True Salary`)
#-----------------------------------------------------------------------------------
# lm model

df <- read.csv(file="nbadata.csv", header=TRUE, sep=",")
head(df)
str(df)

influencesMVPtest <- lm(MVP ~ Player.Efficiency.Rating
                    + FT.Attempt.Rate
                    + True.Shooting..
                    + Defensive.Win.Shares
                    + Offensive.Win.Shares
                    + Usage..
                    + Turnover..
                    + Box....
                    + Minutes.Played
                    + Games.Played
                    + Offensive.Box....
                    + Defensive.Box....
                    + X..Shots.of.Team
                    + Team.True.Shooting..
                    + Team.True.Shooting...w.o.Player
                    + True.Salary
                    + Value.over.Replacement.Player
                    + Estimated.Position
                    + Weight
                    + Win.Shares, data = df)
summary(influencesMVPtest)

influencesMVPEdited <- lm(MVP ~ Minutes.Played
                     + Player.Efficiency.Rating
```

```
                    + FT.Attempt.Rate
                    + True.Salary
                    + True.Shooting..
                    + Value.over.Replacement.Player, data = df)
summary(influencesMVPEdited)

influencesChampionship <- lm(Championship.Team ~ Team.True.Shooting..
                      + Team.Minutes.Played,
                      data = df)
summary(influencesChampionship)
#---------------------------------------------------------------------------------------------
#svm model

projectData <- read.csv(file="nbadata.csv", header=TRUE, sep=",")
str(projectData)
dim(projectData)
table(projectData$MVP)

randIndex <-sample(1:dim(projectData)[1])
cutPoint2_3 <- floor(2* dim(projectData)[1]/3)

trainData <- projectData[randIndex[1:cutPoint2_3],]
testData <- projectData[randIndex[(cutPoint2_3+1):dim(projectData)[1]],]

svmOutput <- ksvm(MVP~.,
             data=trainData,kernel="rbfdot",kpar="automatic",C=50,cross=3,prob.model=TRUE)

svmPred <- predict(svmOutput, testData, type="votes")
compTable <- data.frame(testData[,42], svmPred[1,])
table(compTable)

svmPred
#---------------------------------------------------------------------------------------------
#Viz

TrueSalary <- nbadfselected$`True Salary`
MinutesPlayed <- nbadfselected$`Minutes Played`
plot(MinutesPlayed,TrueSalary)
Model_1<-data.frame(TrueSalary,MinutesPlayed)
mod<- lm(formula= TrueSalary ~ MinutesPlayed, data=Model_1) #Predicts salary based on minutes
played
summary(mod)
#47% can be explained therefore, there is a current issue in over paying non-effiecent players.
abline(mod)

y<-TrueSalary
x<-MinutesPlayed
mean(x)
```

```
mean(y)
gplot<- ggplot(Model_1, aes(x=x,y=y)) +geom_point()
gplot
gplot + stat_smooth(method = 'lm', col
='blue')+geom_vline(aes(xintercept=mean(x),color="red"))+geom_hline(yintercept = mean(y), color="red")
#Everything Inside the left of the red intersection is ineffecient spending

PlayerEfec<-nbadfselected$`Player Efficiency Rating`
TrueShooting<-nbadfselected$`True Shooting %`
ThreePtAttRt<-nbadfselected$`3Pt Attempt Rate`
GamesPlayed<-nbadfselected$`Games Played`
FTAttRt<-nbadfselected$`FT Attempt Rate`
YearsExp<-nbadfselected$`Years Experience`

TurnOvers<-nbadfselected$`Turnover %`
mean(TurnOvers)
gplot2<-ggplot(Model_2,aes(x=TurnOvers,y=TrueSalary))+geom_point()
gplot2
gplot2+geom_hline(yintercept = mean(y),
color="red")+geom_vline(aes(xintercept=mean(TurnOvers),color="red"))
#Shows everyone inside the left upper quadrant is a defensive liability and overpaid for such.

Model_2<-data.frame(TrueSalary,TurnOvers)
mod2<- lm(formula=TrueSalary~TurnOvers, data=nbadfselected)
summary(mod2)

nbaquick <- nbadfselected[nbadfselected$`Player Efficiency Rating` > 20,]
Years<-nbaquick$Year
PlayerEfec<-nbaquick$`Player Efficiency Rating`
tsal <- nbaquick$`True Salary`
scatterplot3d(tsal,Years,PlayerEfec, pch = 16, highlight.3d=TRUE,
            type="h", main="True Salary v Player Efficiency by The Years")

ChampTeamdf <-data.frame(sqldf("SELECT [Championship Team], Team, Year, SUM([True Salary]) AS
[Team True Salary] FROM nbadfselected WHERE [Championship Team] = 1 OR [Runner Up] = 1 GROUP
BY Year, [Championship Team] ORDER BY [Year]"))
TeamSal<-ChampTeamdf$Team.True.Salary
ChampYear<-ChampTeamdf$Year
ChampTeam<-ChampTeamdf$Team
meanch <- sqldf("SELECT AVG([Team.True.Salary]) FROM ChampTeamdf WHERE
[Championship.Team] = 1")
meanru <- sqldf("SELECT AVG([Team.True.Salary]) FROM ChampTeamdf WHERE [Championship.Team]
= 0")

ChampSalG <-
ggplot(ChampTeamdf,aes(x=Year,y=Team.True.Salary,label=Team))+geom_point(aes(color=Championsh
ip.Team, size=4))
```

```
ChampSalG <- ChampSalG + geom_hline(yintercept = meanch[[1]], size=2,
color="red")+geom_text(aes(label=Team),hjust=0,vjust=0)
ChampSalG <- ChampSalG + geom_hline(yintercept = meanru[[1]], size=2, color="black") +
scale_color_gradient(low="#000000", high="#FF0000")
ChampSalG + ggtitle("Team Salaries and Averages")
##Shows how well each champion team payed in salary that year to win the championship..on average
the Spurs paid the most for
#their championships

MVPdf<-sqldf("SELECT [Player Name],[Championship Team], Team, Year, Age,MVP, [Player Efficiency
Rating],[Games Played],[Team True Shooting %],[True Salary] FROM nbadfselected WHERE [MVP] = 1
GROUP BY Year ORDER BY [Year]")
#MVPSalG<-ggplot(MVPdf, aes(MVPdf$Year,MVPdf$`True Salary`,label=MVPdf$`Player Name`))
MVPSalG<-ggplot(MVPdf, aes(MVPdf$Year,MVPdf$`True Salary`,label=MVPdf$`Player
Name`,color=MVPdf$`Championship Team`)) +geom_bar(stat="identity",color="white")
MVPSalG+geom_text(aes(label=MVPdf$`Player Name`))+coord_flip()+ theme(legend.position="none")
#This graphs the MVPs per year by their salary and indicates if they won the championship or not that
year



#map code for avg true salary
avgtrueDf <- as.data.frame(tapply(as.numeric(nbadfselected$'True Salary'),nbadfselected$'Team',mean))

colnames(avgtrueDf) <- "avgtruesalary"

avgtrueDf$'Location' <- c("georgia", "massachusetts","new york","illinois","north
carolina","ohio","texas","colorado","michigan",
                         "california","texas","indiana", "california", "california",
"tennessee","florida","wisconsin","minnesota",
                         "louisiana","new
york","oklahoma","florida","pennsylvania","arizona","oregon","california","texas","canada",
                         "utah","maryland")



us_map <- map_data("state")
map.simple       <- ggplot()
map.simple       <- map.simple +
geom_map(data=us_map,aes(x=us_map$long,y=us_map$lat,map_id=region),map=us_map,fill="white",
color="black")
map.simple

map.salary <- map.simple +
geom_map(data=avgtrueDf,map=us_map,aes(fill=avgtrueDf$avgtruesalary,map_id=avgtrueDf$Location),
color="black",na.rm=TRUE)
map.salary
```

#--