DECEMBER 16, 2016

DOES HUSTLE REALLY WIN GAMES?

AND ARE THE BEST PLAYERS REALLY THE HARDEST WORKERS?

TEJAS BALA
NORTHEASTERN UNIVERSITY

Table of Contents

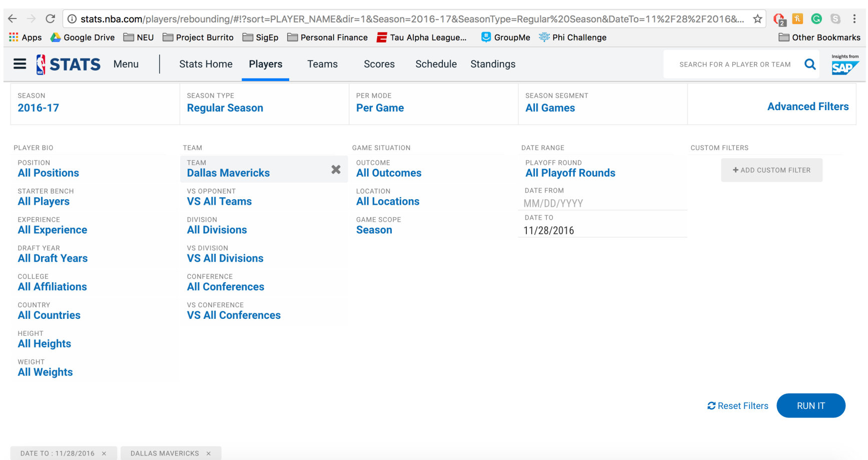
Abstract

This report will explore professional basketball statistics to try to find how much truth is in the day old mantra ringing through gyms across the country, “hustle wins games.” I have recorded a journal throughout this process so you will be able to read about my trials and triumphs with each step along with the results and justification for that step.

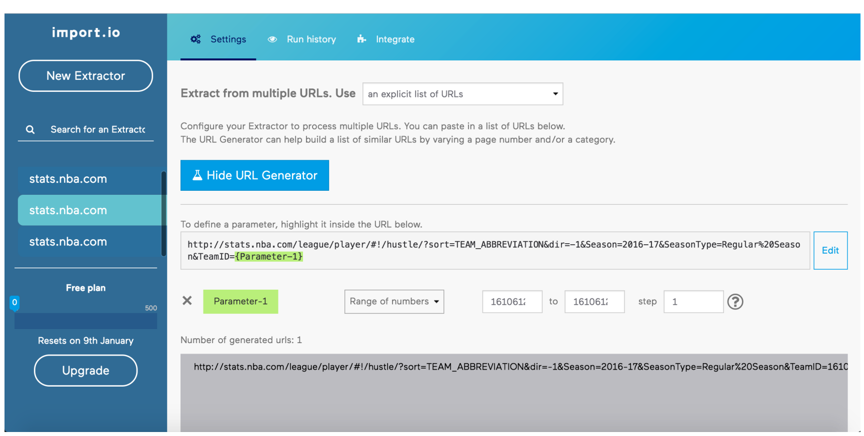
Data Retrieval

The data was scraped from the National Basketball Association's statistics portion of the site. I used import.io to scrape data from three different tables: Traditional Team Stats, Player Hustle Stats, and Player Tracking Rebounding Stats. Since the question was about hustle on the court the Hustle Stats table was an obvious and was the NBA's best collection of data for an otherwise immeasurable characteristic. I added the Rebound Table because I felt that the statistic for Contested Rebounds also demonstrated "grit". All of the data was originally scraped on November 28th but when I realized I forgot to include the rebounding statistics I decided to scrape all of the data again to ensure consistency. The data was scraped for the final time before analysis on December 12th.

When scraping the player data, the data table displayed records 50 players at a time. So when I originally tried to use import.io to scrape the data I only got the first 50 players. I realized I had to filter by each team and input each URL into the scraper.



I saw that import.io had a URL generator for situations like this so I played with the filters and observed the URL until I found the parameter for the team identification number and the range.



Unfortunately, the team identification numbers weren't sequential and resulted in invalid URLs. I had to manually filter each team and paste the URL into the scraper.

When I had to re-scrape the data a few weeks later I originally scraped data up until the date of the last scrape and while examining the data I realized that roster changes would affect me adding the Contested Rebounds column to the player data. I decided to scrape until that day which would also provide me with about 10 more games of data. So I just downloaded the saved set of URLs from the scraper and used the find-replace functionality in a text editor to change the date parameter.

Data Cleaning

Data cleaning as a whole was the most time intensive part.

Team Data

When I downloaded the team data I put it in the data frame teamData. I first had to filter the columns to just get the relevant data: Name, Wins, Losses, Win Percentage and Point Differential (+/-).

```r
#team data first
teamData <- read.csv("team_stats.csv", header = TRUE, sep = ",")
#get relevant columns: name, win, loss, win pct, min
teamData <- teamData[,c(2,5,6,7,8,42)]
```

The data table had the full team name but when I realized that the player data had only the abbreviations I had to create a list of the corresponding team abbreviations (which I got from Wikipedia) and added it as a column to teamData.

```r
#create team abbreviations
teamAbrv <- list("ATL", "BOS", "BKN", "CHA", "CHI", "CLE", "DAL", "DEN", "DET",
                 "GSW", "HOU", "IND", "LAC", "LAL", "MEM", "MIA", "MIL", "MIN", "NOP", "NYK",
                 "OKC", "ORL", "PHI", "PHX", "POR", "SAC", "SAS", "TOR", "UTA", "WAS")
#add team abbreviations to df
teamData$Abrv <- teamAbrv
```

Player Data

I read in the player data CSV file into the playerData data frame. I again filtered the columns to get the relevant data. I excluded Contested Shots as it was a summation of two other columns that I wanted to weight individually when I came up with the GRIT scores.

```r
#player hustle data

playerData <- read.csv("player_hustle.csv")
playerData <- playerData[,c(2,3,4,5,6,7,8,9,10,11)]
playerData[,2] <- as.character(playerData[,2])
colnames(playerData) <- c("Name", "Team", "Age", "MPG", "Screen Assists", "Deflections",
                          "Looseballs Recovered", "Charges Drawn", "Contested 2FG", "Contested 3FG")
```

Rebounding Data

When I originally downloaded the rebounding data I thought I would keep it in its own data frame and create a JOIN statement in my relational database. I decided to just copy the column into playerData because I figured it was easier.

```r
#rebound data
#created its own DF for data integrity purposes,
#to make sure all the players line up before integrating columns with playerData
reboundData <- read.csv("rebound_stats.csv")
reboundData <- reboundData[,c(2,4,5,7)]
colnames(reboundData) <- c("Name", "GP", "MPG", "Contested Rebounds")

playerData$`Contested Rebounds` <- reboundData$`Contested Rebounds`
playerData[is.na(playerData)] <- 0
```

Calculating GRIT

When I first scraped the data I was just playing with coming up with an all-encompassing metric for hustle. I created arbitrary weights for each statistic based on how often I felt it could happen, how impactful it is in the game and trying not to overweight the work of a forward or center versus a guards. I came up with this formula for GRIT:

```
GRIT = (Screen Assists) + 2(Deflections) + 3(Looseballs Recovered) +
3(Charges Drawn) + (Contested 2FG) + 2(Contested 3FG)
```

I also decided to make a normalized GRIT statistic which projected each players' GRIT score as if they played a full 48 minutes. In retrospect this number should have been far closer to 30 which is around the average minutes that starters will play in a game.

Originally my GRIT score calculation was incorrect and I could tell because naturally a starter and a big-man would have a higher GRIT score than a reserve guard and this was not true.

Calculating Team Statistics

I had the option of scraping team hustle stats but I figured if I ever wanted to see how player transactions would affect a team's GRIT or win percentage it's not that hard to just calculate it. For each hustle stat and the GRIT scores I summed up all of the players' stats for each team.

The final data frames ended up like this:

playerData

	Name	Team	Age	MPG	Screen Assists	Deflections	Looseballs Recovered	Charges Drawn	Contested 2FG	Contested 3FG	Contested Rebounds	GRIT	Normalized GRIT
1	DeAndre' Bembry	ATL	22	5	0.0	0.3	0.0	0.00	0.1	0.3	0.0	1.30	12.48000
2	Dennis Schroder	ATL	23	30	0.2	1.6	1.1	0.13	3.3	2.5	0.8	16.19	25.90400
3	Dwight Howard	ATL	31	29	3.9	2.3	0.2	0.00	6.9	0.9	5.4	23.20	38.40000

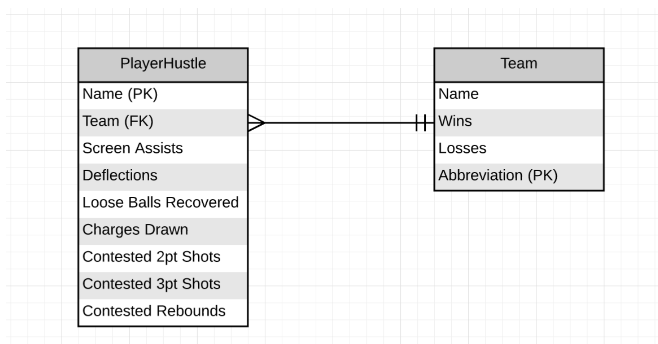
teamData

	Team	Wins	Losses	Win %	MPG	+/-	Team GRIT	Screen Assists	Deflections	Looseballs Recovered	Charges Drawn	Contested 2FG	Contested 3FG	Contested Rebounds
1	ATL	12	12	0.500	48.0	-1.5	186.41	12.6	18.6	5.9	1.07	49.9	24.1	17.6
2	BOS	13	11	0.542	48.0	1.5	197.89	13.7	17.8	7.4	1.33	51.1	27.6	16.1
3	BKN	6	17	0.261	48.4	-8.7	218.44	8.7	21.4	7.8	1.28	58.0	31.7	18.3

Databases and Information Retrieval

I am pretty familiar with using MySQL so I created a database and added the tables and data that were in my data frames.

I had planned to have to following schema:



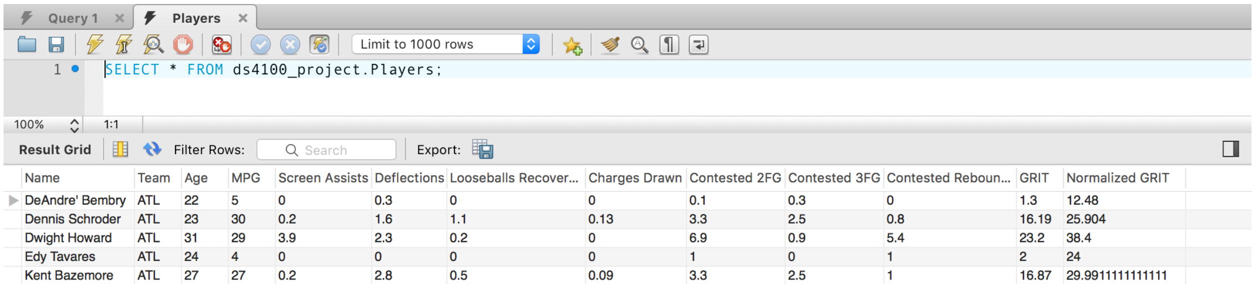
I never created the foreign key relation between player and team but this would be useful for player transaction analysis. Additionally, I added all of the calculated hustle statistics in the team table. For future work I'd like to create indexes for more efficient queries on the database.

I used two methods of information retrieval for my exploratory analysis:

1) Retrieval through R commands. I pulled Team Win Percentage, Team GRIT, and Player Normalized GRIT with SQL select statements.

```
#get calls
winPct <- dbGetQuery(mydb, "SELECT `WIN %` FROM Teams")
teamGRIT <- dbGetQuery(mydb, "SELECT `Team GRIT` FROM Teams")
normPlayerGRIT <- dbGetQuery(mydb, "SELECT `Normalized GRIT` FROM Players")
```

2) Exporting tables from MySQL to .xml Excel files. This helped later along with the predictive models and explanatory analysis.

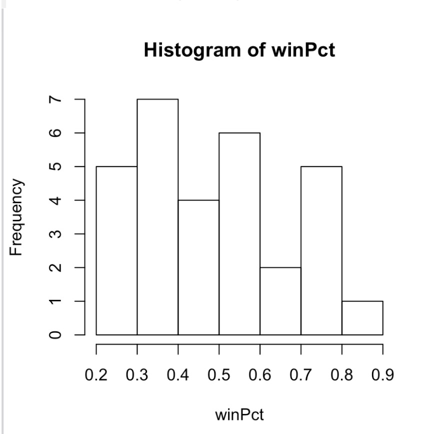


Name	Team	Age	MPG	Screen Assists	Deflections	Looseballs Recover...	Charges Drawn	Contested 2FG	Contested 3FG	Contested Reboun...	GRIT	Normalized GRIT
DeAndre' Bembry	ATL	22	5	0	0.3	0	0	0.1	0.3	0	1.3	12.48
Dennis Schroder	ATL	23	30	0.2	1.6	1.1	0.13	3.3	2.5	0.8	16.19	25.904
Dwight Howard	ATL	31	29	3.9	2.3	0.2	0	6.9	0.9	5.4	23.2	38.4
Edy Tavares	ATL	24	4	0	0	0	0	1	0	1	2	24
Kent Bazemore	ATL	27	27	0.2	2.8	0.5	0.09	3.3	2.5	1	16.87	29.9911111111111

Exploratory analysis

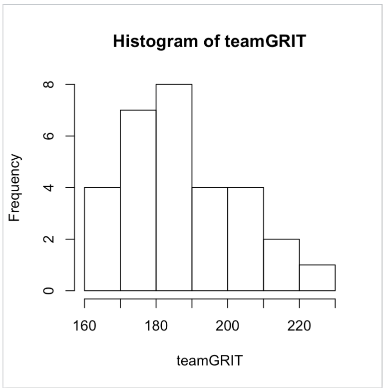
In R I created histograms for Team Win Percentage and Player Normalized GRIT.

```
winPct <- unlist(winPct)
winPct <- as.numeric(winPct)
winHist <- hist(winPct)
winPctMean <- mean(winPct)
```



The average win percentage was .500 and the distribution of winPct was fairly skewed to the left. This was to be expected, most teams are between .300 and .600.

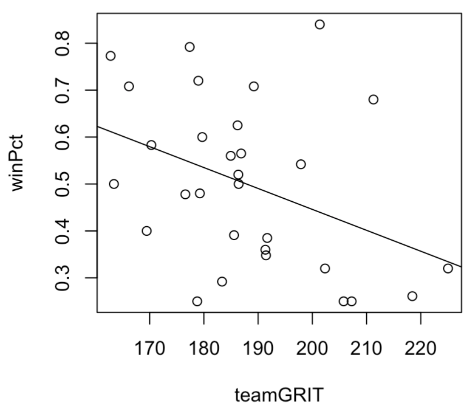
```
teamGRIT <- unlist(teamGRIT)
teamGRIT <- as.numeric(teamGRIT)
teamGritHist <- hist(teamGRIT)
teamGritMean <- mean(teamGRIT)
```



The average teamGRIT score was around 190 and the distribution is a little more normal but still skewed to the left.

The last visualization I made in R was a scatterplot of winPct and teamGRIT.

```
gritWin <- plot(teamGRIT,winPct)
regression <- lm(winPct ~ teamGRIT)
abline(regression)
summary(regression)
```



The trend line was pretty loose fitting and the data had a correlation coefficient of .12. I was pretty surprised to see that if anything GRIT had a negative correlation with win percentage which made me think that maybe a higher GRIT score equates to sloppier basketball or wasted effort on fairly inconsequential parts of the game.

Predictive Model

At this point I figured that my best chance for fitting a model was to use the individual team hustle stats in a multiple regression model for predicting win percentage.

This was my first model with all variables:

SUMMARY OUTPUT	7 var				
Regression Statistics					
Multiple R	0.49288386				
R Square	0.2429345				
Adjusted R S	0.00205002				
Standard Err	0.17760125				
Observations	30				
ANOVA					
	df	*SS*	*MS*	*F*	*Significance F*
Regression	7	0.22267448	0.03181064	1.0085104	0.45203254
Residual	22	0.69392848	0.0315422		
Total	29	0.91660297			
	Coefficients	*Standard Erro*	*t Stat*	*P-value*	
Intercept	1.23493996	0.55517766	2.224405	0.03669516	
Screen Assist	0.00563925	0.01617142	0.34871695	0.73061703	
Deflections	-0.0006397	0.01586049	-0.0403302	0.96819367	1st out
Looseballs R	0.03140888	0.05668533	0.55409196	0.58510472	
Charges Drav	-0.1388121	0.09134012	-1.5197279	0.14282165	
Contested 2F	-0.0054992	0.00770266	-0.7139313	0.48277593	
Contested 3F	-0.0127376	0.01332221	-0.9561135	0.34940415	
Contested Re	-0.01548	0.02314383	-0.6688591	0.51054363	

The model was extremely weak and none of the variables had the appropriate P-value to be significant.

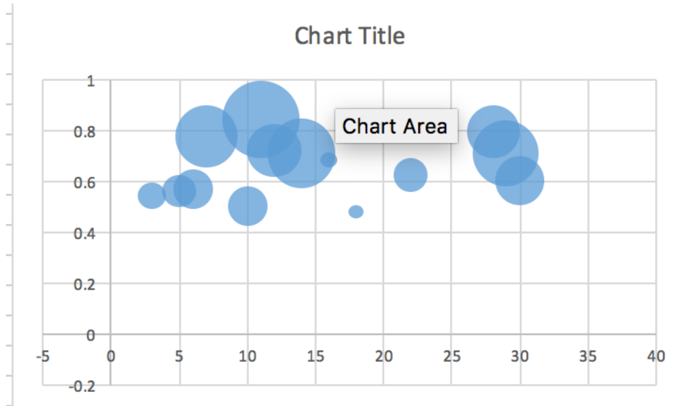
After 6 iterations of removing the least significant variable I ended with:

SUMMARY OUTPUT					
Regression Statistics					
Multiple R	0.33676979				
R Square	0.11341389				
Adjusted R S	0.08175011				
Standard Err	0.17036174				
Observations	30				
ANOVA					
	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	0.10395551	0.10395551	3.58181685	0.06879591
Residual	28	0.81264745	0.02902312		
Total	29	0.91660297			
	Coefficients	*Standard Erro*	*t Stat*	*P-value*	
Intercept	0.60187944	0.06215586	9.68339076	1.9487E-10	
Charges Drav	-0.1398345	0.07388607	-1.8925688	0.06879591	

So effectively I had no model for predicting win percentage based on hustle statistics.

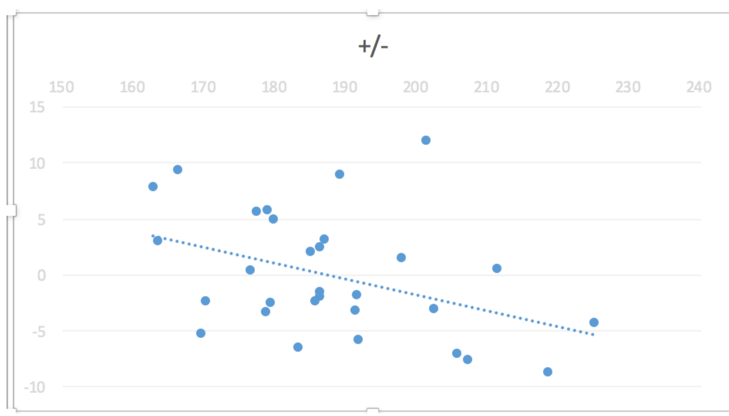
Explanatory analysis

When I started this step I really did not know what I wanted to present, I figured since the model didn't work that I didn't have a story to tell. I tried to play with some of the original ideas I had and create bubble charts with x: Team GRIT, y: Win Pct, size: +/-. It resulted in:

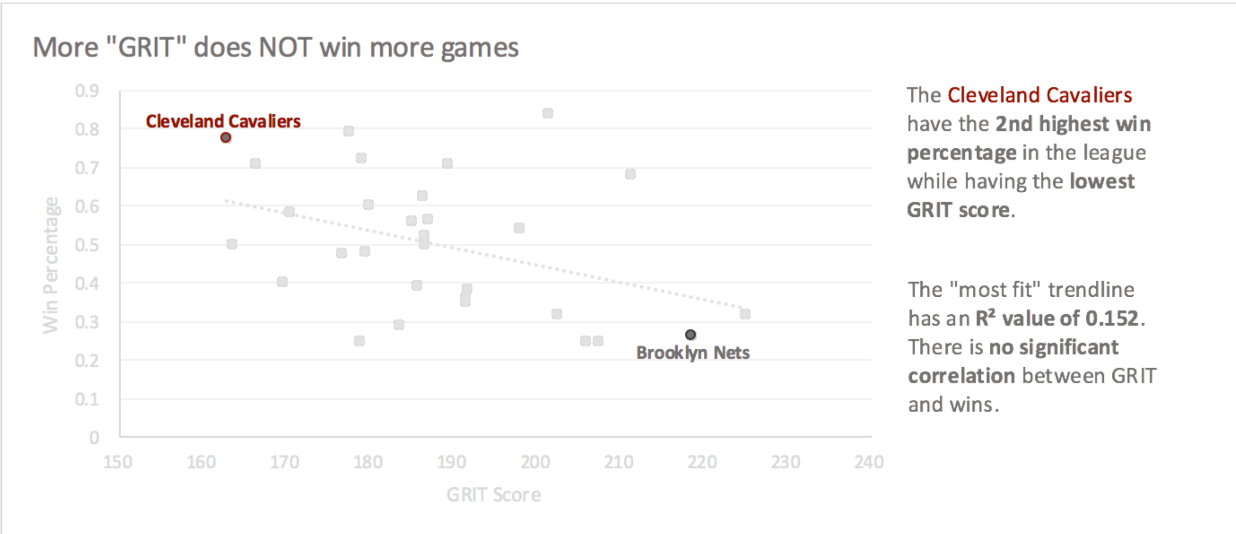


I was not sold enough on this visualization to try to figure out how to make the x-axis measure Team GRIT as I intended rather than the row number.

I went back to plotting basic scatter plots. The first is a chart of Team GRIT and Point Differential:

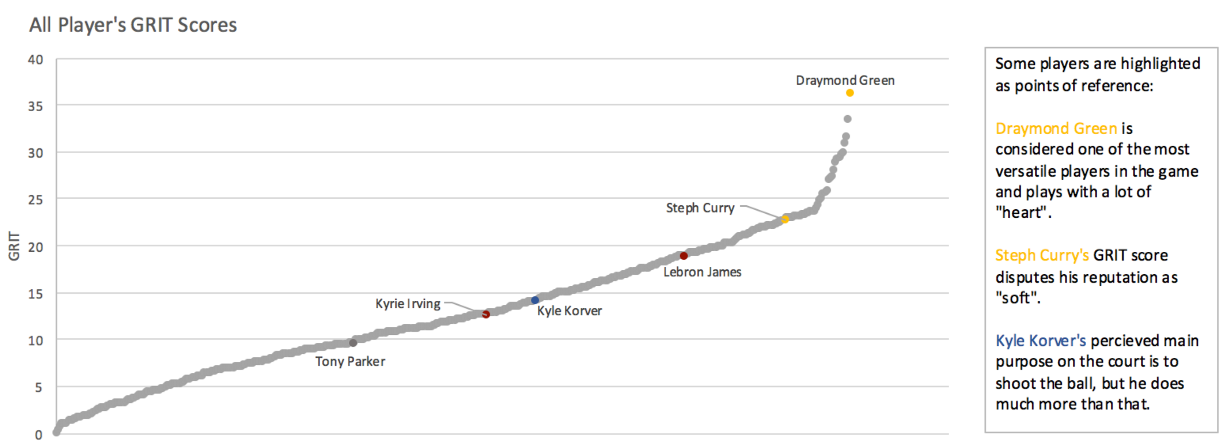


This was basically reflected the same information as the next chart, Win Percentage and Team GRIT.

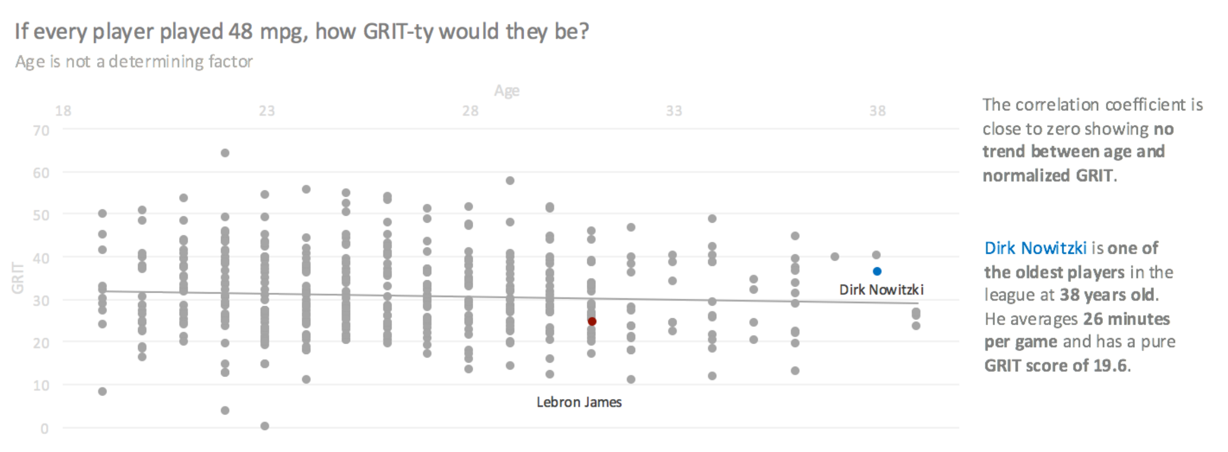


This visualization clearly states the conclusion and uses pre-attentive attributes to show the justification for disputing the hypothesis.

I then decided to see if more GRIT made a player more valuable. I started off by plotting all player's GRIT scores and highlighting points of reference.

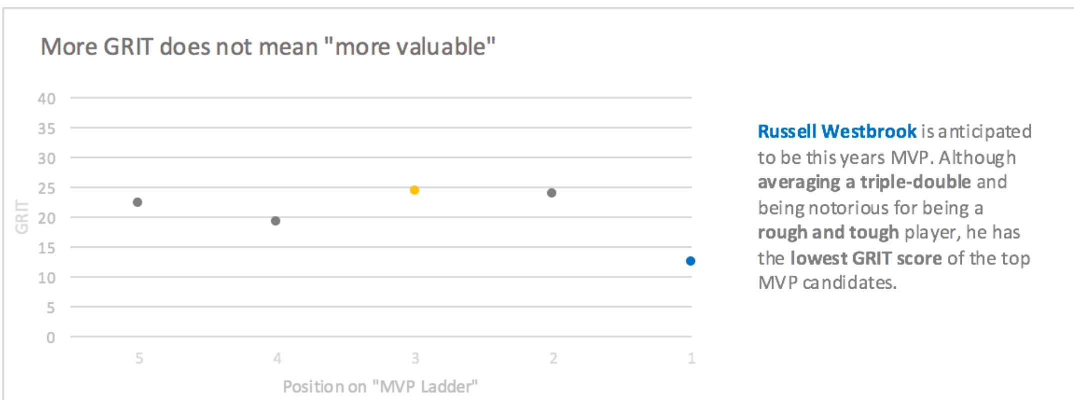


I moved on to see if age had an effect on GRIT. It obviously affects the average minutes per game that a player plays but I was more interested with how much "hustle" they played with while on the court. For this reason, I used Normalized GRIT.



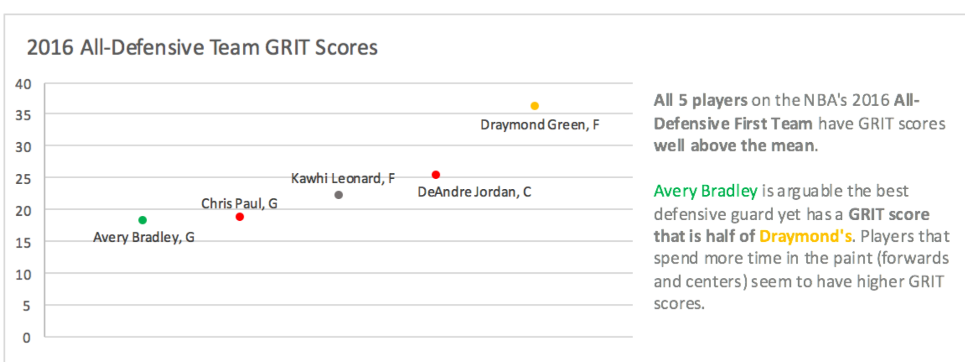
I concluded from this that age was not a determining factor.

I then used outside data to see what else I could try to find GRIT trends with.



This chart concluded that the most GRIT does not mean “Most Valuable”.

Hustle and grit are often buzz words used by coaches to inspire defenders. So I wanted to see if the top defensive players from last season had high GRIT scores.



Data Quality

Overall I think the quality of data was great. With the NBA's new sports data analytics initiative, they keep much more insightful data and they keep it in a very organized and clean manner. For some players that averages 0-2 minutes per game they had "N/A" for their hustle stats with which I just replaced with zeros.

Final Thoughts

I think if I looked more into a player's defensive effectiveness along with hustle stats I may have come up with more captivating findings. But I think the original hypothesis wasn't grounded in much truth so although I am surprised that there wasn't a stronger correlation between hustle stats and wins I can understand why there isn't.