# The Optimal NBA Roster-Building Strategy

Patrick Simpson

May 6, 2020

**Abstract**

Executives in the National Basketball Association customarily emulate the roster-building strategy of successful franchises. While a superstar is the foundation of a team, marginal advantages propel it to championship contention. One way an organization can gain an edge is through shrewd roster management. In this project, hierarchical clustering is used to group players by ability. After clusters are created, we investigate which players are similar. With this intel, we can make sound decisions regarding roster construction.

## 1 Background

*Cluster analysis* methods aim to divide data into meaningful groupings known as clusters. Objects in one cluster should share similarities that make them distinct from objects in other clusters. While humans instinctively group objects, a quantitative methodology was not introduced until the early 20th century. It originated in the fields of anthropology and psychology. In 1943, Raymond B. Cattell of Harvard University used one of the first known cluster analysis methods to group humans by common personality traits. Shortly afterward, scientists began applying cluster analysis in biology and epidemiology; it was particularly useful in taxonomy and medicine. With the technological advances made over the last thirty years, cluster analysis is now a fundamental method in data science, utilized in the fields of finance, marketing, and biostatistics.

## 2 Methodology

There are many methods used to perform cluster analysis. Centroid clustering, density clustering, distribution clustering, connectivity clustering, and hierarchical clustering are well-known methods in the field. Each method has distinct advantages. I applied hierarchical clustering in my research.

There are two types of hierarchical clustering: divisive and agglomerative. Divisive is a top-down approach. Suppose a data set has $N$ observations. Beginning with one cluster of size $N$, it will recursively split into smaller groups until there are $N$ clusters of size 1.

**Algorithm** Divisive Clustering Algorithm

1: Starting with a dataset of $N$ observations, all begin in one cluster.
2: The most dissimilar observation is removed and becomes its own cluster.
3: Repeat step (2) until each observation is in its own cluster.

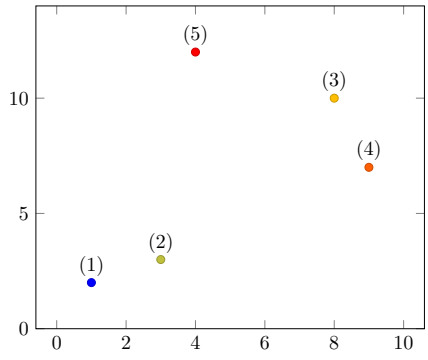
Agglomerative clustering is a bottom-up approach. Suppose a dataset has $N$ observations. Beginning with $N$ clusters of size 1, they will recursively converge to one group of size $N$.

**Algorithm** Agglomerative Clustering Algorithm

1: Starting with a dataset of $N$ observations, each begins in its own cluster. Form an $N$x$N$ proximity matrix.
2: Identify the two most similar clusters.
3: Merge those clusters and recompute the proximity matrix (Note: it will have one fewer row and column than the previous matrix).
4: Repeat steps (2) and (3) recursively until the proximity matrix is 1x1.

Agglomerative hierarchical clustering is the focus of this paper. I will use a sample data set to explain the algorithm:

| x | y |
|---|---|
| 1 | 2 |
| 3 | 3 |
| 8 | 10 |
| 9 | 7 |
| 4 | 12 |




There are five observations labeled 1-5. Each is in a single cluster. To form a $5x5$ proximity matrix, one must select a measure of distance. The two most commonly used measures of distance are Euclidean and Manhattan. For vectors $x$ and $y$, each with $n$ elements, the distance formulas are defined as follows:

$$d_{euc}(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$d_{man}(x,y) = \sum_{i=1}^{n}|x_i - y_i|$$

Euclidean distance is known for its relative simplicity. However, Manhattan distance may be preferable when working with vectors containing many dimensions. There are also correlation-based distance measures such as Pearson's correlation distance and Ward's method. The former determines the linear

relationship between two vectors. The latter minimizes the variance of observations within the same clusters.

In this example, I apply Euclidean distance.

The distance between each point:
***1*** & ***2***: $d_{euc}(1,2) = \sqrt{(1-3)^2 + (2-3)^2} = \mathbf{2.24}$
***1*** & ***3***: $d_{euc}(1,3) = \sqrt{(1-8)^2 + (2-10)^2} = \mathbf{10.63}$
***1*** & ***4***: $d_{euc}(1,4) = \sqrt{(1-9)^2 + (2-7)^2} = \mathbf{9.43}$
***1*** & ***5***: $d_{euc}(1,5) = \sqrt{(1-4)^2 + (2-12)^2} = \mathbf{10.44}$
***2*** & ***3***: $d_{euc}(2,3) = \sqrt{(3-8)^2 + (3-10)^2} = \mathbf{8.60}$
***2*** & ***4***: $d_{euc}(2,4) = \sqrt{(3-9)^2 + (3-7)^2} = \mathbf{7.21}$
***2*** & ***5***: $d_{euc}(2,5) = \sqrt{(3-4)^2 + (3-12)^2} = \mathbf{9.06}$
***3*** & ***4***: $d_{euc}(3,4) = \sqrt{(8-9)^2 + (10-7)^2} = \mathbf{3.16}$
***3*** & ***5***: $d_{euc}(3,5) = \sqrt{(8-4)^2 + (10-12)^2} = \mathbf{4.47}$
***4*** & ***5***: $d_{euc}(4,5) = \sqrt{(9-4)^2 + (7-12)^2} = \mathbf{7.07}$

Now a proximity matrix can be computed with the measures of distance:

| | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| **1** | **0** | 2.24 | 10.63 | 9.43 | 10.44 |
| **2** | 2.24 | **0** | 8.60 | 7.21 | 9.06 |
| **3** | 10.63 | 8.60 | **0** | 3.16 | 4.47 |
| **4** | 9.43 | 7.21 | 3.16 | **0** | 7.07 |
| **5** | 10.44 | 9.06 | 4.47 | 7.07 | **0** |

Since the distance between an observation and itself will always be 0, diagonal entries of any proximity matrix will always be 0. All entries will be non-negative, with larger values indicating clusters farther apart.

In step 2 of the algorithm, I locate the smallest non-zero entry of the proximity matrix. In this example, it is 2.24. Now, points *1* and *2* will merge to form a unique cluster: *12*. When recomputing the proximity matrix, there will be four clusters, *12*, *3*, *4*, and *5*. For the distances between single observation clusters, the matrix entries will hold constant. However, a dissimilarity measure must be specified to calculate the distance between a multi-observational cluster with another cluster.

| | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| **1** | **0** | 2.24 | 10.63 | 9.43 | 10.44 |
| **2** | 2.24 | **0** | 8.60 | 7.21 | 9.06 |
| **3** | 10.63 | 8.60 | **0** | 3.16 | 4.47 |
| **4** | 9.43 | 7.21 | 3.16 | **0** | 7.07 |
| **5** | 10.44 | 9.06 | 4.47 | 7.07 | **0** |

**Definition 1.** *Let G and H represent two such clusters. The dissimilarity d(G,H) between G and H is computed from the set of pairwise observation dissimilarities $d_{ii'}$ where one member of the pair i is in G and the other $i^{'}$ is in H.*

The three most common measures of dissimilarity are single linkage, complete linkage, and group average linkage. Single linkage agglomerative clustering takes the intergroup dissimilarity to be that of the closest (least dissimilar) pair:

$$d_{SL}(G, H) = \min_{i \in G, i' \in H} d_{ii'}$$

Complete linkage agglomerative clustering takes the intergroup dissimilarity to be that of the furthest (most dissimilar) pair:

$$d_{CL}(G, H) = \max_{i \in G, i' \in H} d_{ii'}$$

Group average agglomerative clustering takes the intergroup dissimilarity to be the average dissimilarity between the groups:

$$d_{GA}(G, H) = \frac{1}{N_G N_H} \sum_{i \in G} \sum_{i' \in H} d_{ii'}$$

Where $N_G$ and $N_H$ represent the number of observations in groups G and H. For this example, the chosen measure of dissimilarity will be complete linkage. The distance between each cluster is as follows:

***12*** & ***3***: $d_{CL}(12,3) = \max\Big(d_{euc}(1,3), d_{euc}(2,3)\Big) = \max\Big(\sqrt{(1-8)^2 + (2-10)^2}, \sqrt{(3-8)^2 + (3-10)^2}\Big) = \max(10.63, 8.60) = \mathbf{10.63}$

***12*** & ***4***: $d_{CL}(12,4) = \max\Big(d_{euc}(1,4), d_{euc}(2,4)\Big) = \max\Big(\sqrt{(1-9)^2 + (2-7)^2}, \sqrt{(3-9)^2 + (3-7)^2}\Big) = \max(9.43, 7.21) = \mathbf{9.43}$

***12*** & ***5***: $d_{CL}(12,5) = \max\Big(d_{euc}(1,5), d_{euc}(2,5)\Big) = \max\Big(\sqrt{(1-4)^2 + (2-12)^2}, \sqrt{(3-4)^2 + (3-12)^2}\Big) = \max(10.44, 9.06) = \mathbf{10.44}$

***3*** & ***4***: $d_{euc}(3,4) = \sqrt{(8-9)^2 + (10-7)^2} = \mathbf{3.16}$

***3*** & ***5***: $d_{euc}(3,5) = \sqrt{(8-4)^2 + (10-12)^2} = \mathbf{4.47}$

***4*** & ***5***: $d_{euc}(4,5) = \sqrt{(9-4)^2 + (7-12)^2} = \mathbf{7.07}$

The new proximity matrix is as follows:

| | **12** | **3** | **4** | **5** |
|---|---|---|---|---|
| **12** | **0** | 10.63 | 9.43 | 10.44 |
| **3** | 10.63 | **0** | 3.16 | 4.47 |
| **4** | 9.43 | 3.16 | **0** | 7.07 |
| **5** | 10.44 | 4.47 | 7.07 | **0** |

Now, steps 2 and 3 of the algorithm are repeated until the proximity matrix is $1x1$. The smallest non-zero entry is 3.16, denoting the distance between clusters *3* and *4*. These clusters will merge to form *34*. Calculating the new distances:

***12 & 34***: $d_{CL}(12,34) = \max\Big(d_{euc}(1,3),\, d_{euc}(2,3),\, d_{euc}(1,4),\, d_{euc}(2,4)\Big) =$
$\max\Big(\sqrt{(1-8)^2 + (2-10)^2}, \sqrt{(3-8)^2 + (3-10)^2}, \sqrt{(1-9)^2 + (2-7)^2},$
$\sqrt{(3-9)^2 + (3-7)^2}\Big) = \max\Big(10.63, 8.60, 9.43, 7.21\Big) = \mathbf{10.63}$

***12 & 5***: $d_{CL}(12,5) = \max\Big(d_{euc}(1,5),\, d_{euc}(2,5)\Big) = \max\Big(\sqrt{(1-4)^2 + (2-12)^2},$
$\sqrt{(3-4)^2 + (3-12)^2}\Big) = \max(10.44, 9.06) = \mathbf{10.44}$

***34 & 5***: $d_{CL}(34,5) = \max\Big(d_{euc}(3,5),\, d_{euc}(4,5)\Big) = \max\Big(\sqrt{(8-4)^2 + (10-12)^2},$
$\sqrt{(9-4)^2 + (7-12)^2}\Big) = \max(4.47, 7.07) = \mathbf{7.07}$

The new proximity matrix:

| | **12** | **34** | **5** |
|---|---|---|---|
| **12** | **0** | 10.63 | 10.44 |
| **34** | 10.63 | **0** | 7.07 |
| **5** | 10.44 | 7.07 | **0** |

Next, *34* and *5* will merge to form *345*. Calculating the new distances:

***12 & 345***: $d_{CL}(12,345) = \max\Big(d_{euc}(1,3),\, d_{euc}(1,4),\, d_{euc}(1,5),\, d_{euc}(2,3),$
$d_{euc}(2,4),\, d_{euc}(2,5)\Big) = \max\Big(\sqrt{(1-8)^2 + (2-10)^2}, \sqrt{(1-9)^2 + (2-7)^2},$
$\sqrt{(1-4)^2 + (2-12)^2}, \sqrt{(3-8)^2 + (3-10)^2}, \sqrt{(3-9)^2 + (3-7)^2},$
$\sqrt{(3-4)^2 + (3-12)^2}\Big) = \max(10.63, 9.43, 10.44, 8.60, 7.21, 9.06) = \mathbf{10.63}$

The new proximity matrix:

| | 12 | 345 |
|---|---|---|
| 12 | 0 | 10.63 |
| 345 | 10.63 | 0 |

Now there are only two clusters to merge, *12* and *345*. The final proximity matrix:

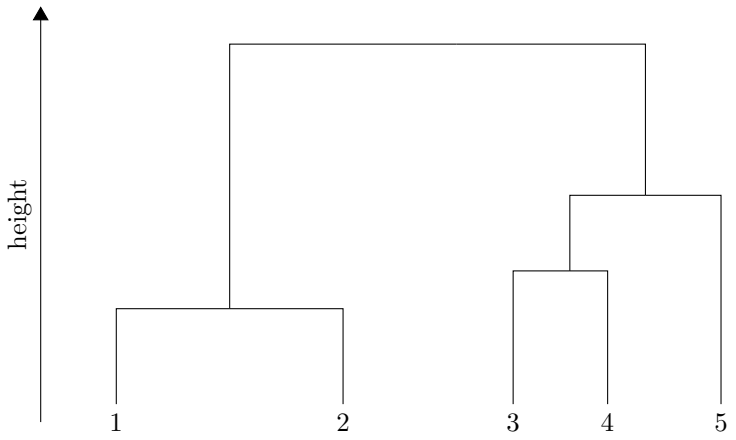
| | 12345 |
|---|---|
| 12345 | 0 |

## Examining Measures of Dissimilarity

If the data has strong clustering properties, all three dissimilarity measures yield similar results. Each method also has disadvantages. In single linkage, the threshold for grouping two objects is low. Thus, the maximum distance between any two observations in a cluster, the diameter, tends to be very large in groups produced by single linkage. Known as chaining, it can lead to clusters containing observations that lack relative closeness. In complete linkage, the threshold for grouping two objects is high. Thus, observations within the same cluster tend to be compact with small diameters. Consequently, objects within a cluster may be closer to observations in another group violating the "closeness" property. Group average linkage is theoretically a compromise of single and complete linkage. Each cluster should be relatively compact and far apart. However, unlike single and complete, group average linkage is not invariant to transformations on the distance measures.

## What about Divisive Clustering?

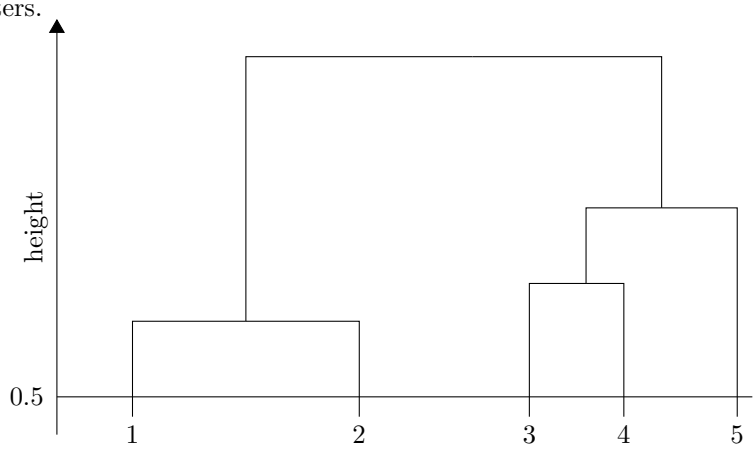
Agglomerative clustering is much more prevalent in practice than divisive. With divisive clustering, a user must first determine how many clusters he or she desires, much like centroid based clustering. This inconvenience does not exist for agglomerative clustering. Once the final iteration is complete, one can choose how many clusters he or she wants. Also, agglomerative methods can be accompanied by a dendrogram, a visualization of the hierarchical structure. Only some divisive methods can. Below is a dendrogram representing the sample data.



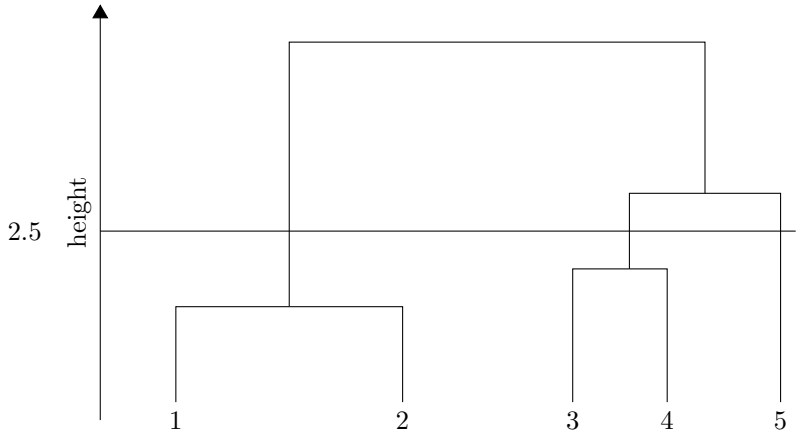

Each observation is located along the bottom of the dendrogram. These are known as terminal nodes. If there are $N$ observations, there will be $N$ terminal nodes. Observations merge at non-terminal (parent) nodes. For any hierarchical clustering dendrogram with $N$ terminal nodes, there are $N-1$ non-terminal nodes. For every parent node, two daughter nodes belong to it. In this example, there are 5 terminal and 4 non-terminal nodes.

All agglomerative clustering methods possess a monotonicity property. Each dissimilarity measure between clusters increases with the level of the merger. For example, the first two observations merged will be more similar than the $(n-1)$ and $n$ observations merged. Therefore, the distance between parent nodes and daughter nodes are monotone increasing.

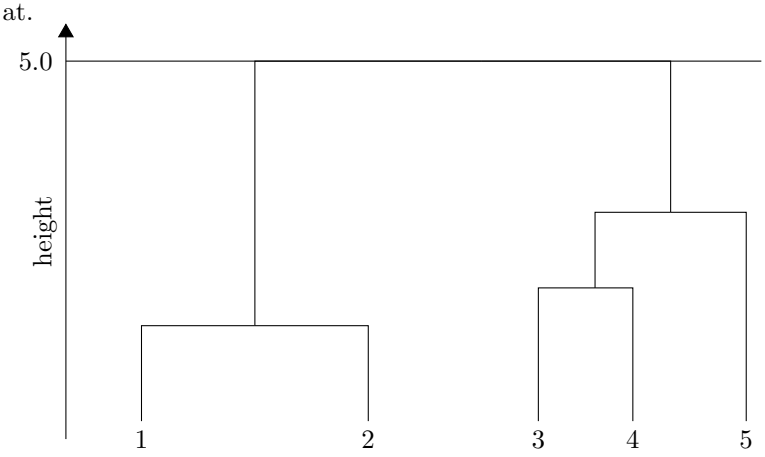
The dendrogram can be "cut" at any height to obtain clusters. Suppose the dendrogram is "cut" at a height of 0.50. Since this is not greater than the height of any non-terminal node, all five observations would remain in individual clusters.




If the dendrogram is "cut" at a height of 2.5, the objects form clusters accordingly: *12*, *34*, and *5*.




If the dendrogram is "cut" at a height of 5, one cluster would be obtained: *12345*. Note that there are infinitely many heights that a dendrogram can be cut at.




## Hierarchical Clustering in R

If a dataset contains a large number of observations or if each vector consists of many elements, technology is necessary to perform hierarchical clustering analysis. In this project, agglomerative clustering analysis was performed in the open-source statistical package, R. In the stats package in R, four functions were frequently used: *dist*, *hclust*, *cutree*, and *dendrogram*.

### dist Function

The dist function returns the initial proximity matrix for a data set. This function requires at least two user inputs: a data frame with the observations as rows and a distance measure. Returning to the sample data, the data frame would be represented by this:

| | x | y |
|---|---|---|
| 1 | 1 | 2 |
| 2 | 3 | 3 |
| 3 | 8 | 10 |
| 4 | 9 | 7 |
| 5 | 4 | 12 |

Note that the first column contains the row names representing each vector. For more complex data sets, it is wise to "normalize" the vector components before creating the proximity matrix with the dist function (i.e. subtract the mean and divide by the standard deviation). Suppose the vectors used in hierarchical clustering analysis contained two elements with different units: A and B. If unit B is much larger than unit A, the distance measure will be heavily weighted in favor of B. Without normalizing each component, the analysis may not return meaningful results. The *scale* function in R normalizes the columns of a data frame.

For the distance measure, the *dist* function accepts *euclidean*, *maximum*, *manhattan*, *canberra*, *binary*, and *minkowski*. For the sample data, euclidean distance would be used. Suppose the data frame is saved to the R environment as "DF." To obtain the proximity matrix one enters dist(scale(DF), method = "euclidean").

**hclust function**

The hclust function performs the cluster analysis. Two user inputs are required: a proximity matrix and a method of dissimilarity. The proximity matrix is obtained using the dist function and the method of dissimilarity can be *single*, *complete*, *average*, *ward.D*, *ward.D2*, *mcquitty*, *median*, or *centroid*. Suppose the proximity matrix is saved to the R environment as "proximityMatrix" and complete linkage is used. The input would be hclust(proximityMatrix, method = "complete"). To obtain a dedrogram, one can pass the hclust function through *dendrogram*: dendrogram(hclust(proximityMatrix, method = "complete")).

**cutree function**

After obtaining a dendrogram, it can be "cut" at different heights depending on how many clusters one desires. In R, the cutree function will return $n$ clusters if given the following parameters: the output from the hclust function and the number of desired clusters (specified by $n$). Suppose the hclust output in the sample dataset is saved as "hclustOutput" and 3 clusters are wanted. Using the cutree function, the input is cutree(hclustOutput, k = 3).

# 3 Data Collection

Shooting data was obtained by web scraping nba.com/stats with python. A detailed record of every shot attempt dating back to the 1996-1997 season was acquired. There were 4,463,265 observations in this data frame. The variables are listed below:

- **slugSeason:** The season the shot took place in
- **idTeam:** The id of the team the shooter played for
- **idPlayer:** The id of the player
- **namePlayer:** The name of the player
- **nameTeam:** The name of the team
- **typeEvent:** Whether the shot was made or missed
- **typeAction:** Description of the shot attempted
- **typeShot:** Whether the shot was a 3PT or 2PT field goal
- **dateGame:** Date the game took place
- **slugTeamHome:** Home team abbreviation
- **slugTeamAway:** Away team abbreviation
- **idGame:** ID number of the game
- **idEvent:** Every action that takes place in a game is recorded as an "event." The lower the number is, the earlier it took place.
- **numberPeriod:** The quarter the shot was attemped in
- **minutesRemaining:** The number of minutes remaining in the quarter when the shot was attempted
- **secondsRemaining:** The number of seconds remaining in the quarter when the shot was attempted
- **zoneBasic:** The court is divided into 6 sections (Above the break 3, Right Corner 3, Left Corner 3, Mid-Range, In the Paint (Non-Restricted Area, Restricted Area). This is where the shot was attempted.
- **slugZone:** Abbreviation of the name of zone
- **zoneRange:** The distance of the shot was divided into 3 zones. This is the name of the zone
- **locationX:** The x-coordinate of the shot (used for graphing)

- **locationY:** The y-coordinate of the shot (used for graphing)
- **distanceShot:** The distance of the shot in feet

I obtained additional data using the r package "nbastatr," created by Alex Bresler. Using its various functions, I acquired the number of dribbles a player took before attempting each shot, defender proximity on each shot, team rosters for each season, traditional and advanced box scores from every game, play-by-play information for every game, and player bios. Individual "Raptor" defensive ratings for players were collected from 538.com. I also gathered data from synergy sports technology that partitions each shot into one of eleven categories:

- **Transition:** (Fast-Break possessions)
- **Isolation:** (Player creates own shot without an assist from teammate)
- **Pick and Roll Ball Handler:** (Described in detail in **4.4**)
- **Pick and Roll Roll Man:** (Described in detail in **4.4**)
- **Postup:** (Player creates shot with back to basket without an assist from teammate)
- **Spotup:** (Catch and Shoot)
- **Handoff:** (Player recieves ball via a handoff before attempting a shot)
- **Cut:** (Player cuts to basket before receiving a pass leading to a shot)
- **Off Screen:** (Off-ball player comes off a screen before attempting a shot)
- **Off Rebound:** (Put back via offensive rebound)
- **Miscellaneous:** (Anything else)

Passing data was scraped from pbpstats.com. The following categories were used in my analysis:

- **Assist Zone:** Where a player shoots after a pass from a given player (At the rim, Corner 3, etc.)
- **Bad Pass Turnovers:** Turnovers that the passer is responsible for
- **Live Ball Turnovers:** Opposing team immediately has possession of the ball after a turnover
- **Dead Ball Turnovers:** Stoppage in play after a turnover (ball goes out of bounds, etc.)
- **Potential Assists:** Player passes to a teammate who immediately shoots
- **Passes:** Number of passes a player made

While some of these statistics are subjective: type of shot attempted (made shots are more likely to have a vivid description), whether the pass was "bad," most of them were conclusive. When preparing the data for cluster analysis, I aimed to be objective. While some player tracking data is unreliable to an extent, synchronized game data is much more valuable than numbers from traditional box scores.

# 4 Data Organization

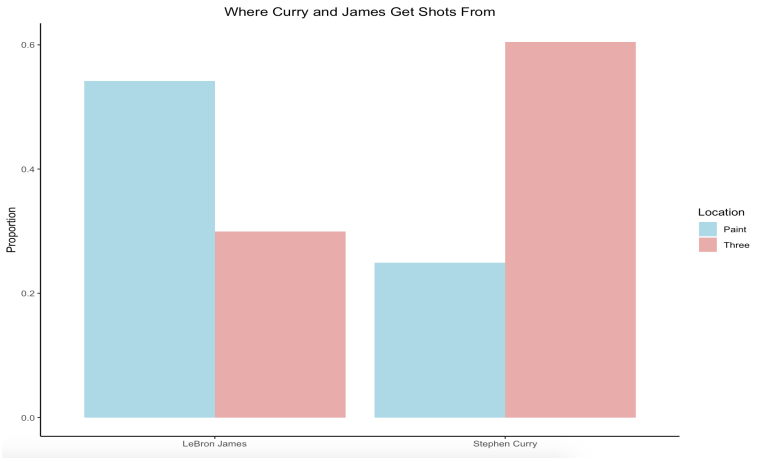
The player tracking era began in 2015. Thus, my analysis focused on the last four seasons (2015-2019). Before performing hierarchical clustering analysis for each season, I had to generate player vectors. Each vector was composed of 8 elements that I created. Each metric is a numeric value with larger values indicating a more prominent ability. The eight components are listed below:

- **3PT Shooting:** Ability to shoot 3PT shots
- **Mid-Range Scoring:** Ability to score from the mid-range
- **Inside Scoring:** Ability to score inside the paint
- **Playmaking:** Ability to create shots for teammates
- **Roll Gravity:** Stress the roll-man in the pick-and-roll puts on the defense
- **Isolation:** Ability to create shot for self
- **Rebounding:** Ability to rebound the basketball
- **Defense:** Individual Defensive Impact

## Why Eight Components?

My goal was to cluster players by skill-sets. However, these are often complex. For example, LeBron James and Stephen Curry are great scorers but get their points in different ways. In 2018-19, 29.95% of LeBron's attempts were three-point shots compared to 59.63% of Curry's. 54.16% of James' attempts came from the paint compared to 24.92% of Curry's.

Suppose two "catch-all" metrics were used, one for offensive ability and another for defensive ability. Due to their tremendous offensive talent and average defense, Curry and James would be considered similar players. Although each is an offensive fulcrum, their playstyle differs and additional components are necessary to capture this.




There were challenges when creating each component. Players on the same teams will inevitably have similar shooting profiles. Some players will have favorable numbers due to the environment they play in. To obtain the true ability of a player, I attempted to adjust for team philosophy, teammate ability, and volume.

## Background

There are three locations a shot may be attempted from: the paint, mid-range, and three-point range. Over the past five seasons, shot selection has drastically changed. Organizations are now fixated with maximizing offensive efficiency. To accomplish this, teams have focused on shots from three-point range and the paint. The math is simple: a player is likely to score more points when attempting a three or a shot in the paint:

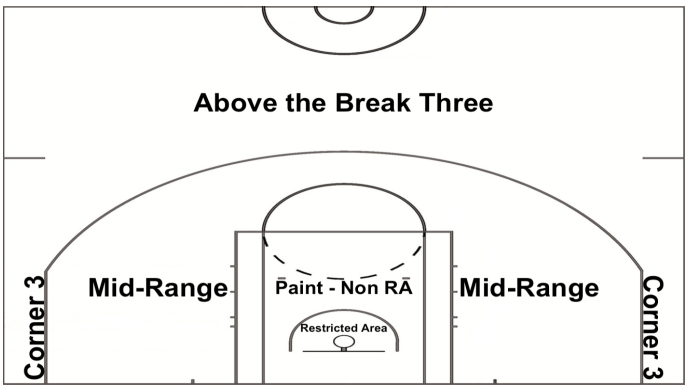
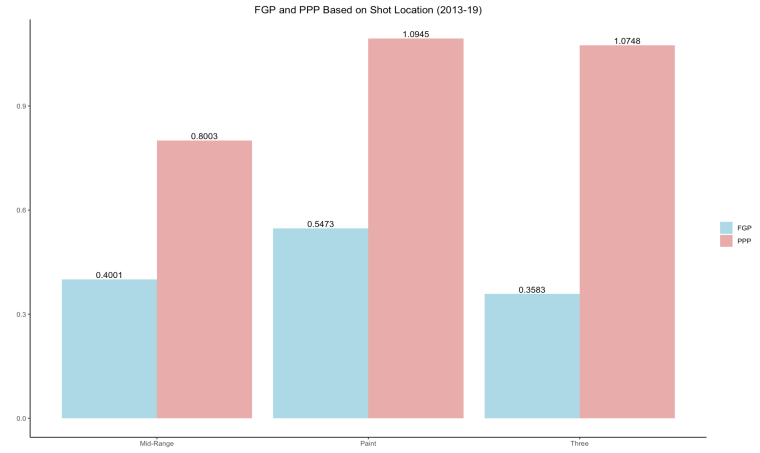


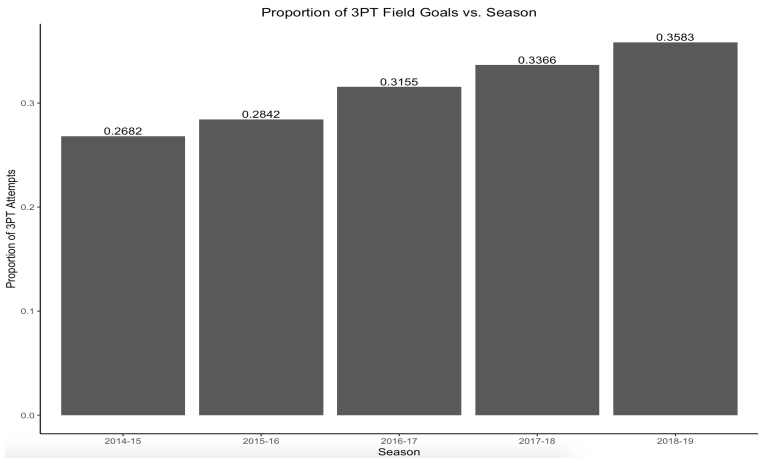

Figure 1: Locations on the Court




As displayed in the graph above, three-point shots have the lowest field goal percentage. However, since they are worth three points, the expected points per possession is nearly identical to a shot attempted close to the basket. Note that PPP = field goal percentage $*$ points awarded for a made basket (A three-point shot is worth 3-points while any other shot is worth 2 points).

## 4.1 3PT Shooting Component

Basketball is in the midst of the three-point revolution. Over the past five seasons, frequency has steadily increased. It is unlikely that teams return to pre-2014 levels, but many wonder if the league is approaching its upper limit.




Three-point shooting ability has never had been more valuable. While field-goal percentage provides a general sense of a player's shooting ability, it can be

misleading. Defender intensity, play type, and three-point location, all affect a player's success rate.

Using tracking data, I classified each three-point shot based on defender proximity. At the release, if a defender was within 0-4 feet, it was considered "contested." If a defender was within 4-6 feet, it was considered "open." Finally, if the nearest defender was 6+ feet away, it was deemed "wide-open." Additionally, I partitioned three-point attempts by location: Corner or Above-the-Break. Corner three-point attempts are slightly more valuable. They are the shortest distance of any shot that counts for 3 points. Also, over 90% of them are assisted.

I calculated the "shooting ability over-expectation" in five different areas: contested, open, wide-open, corner, and above the break. For each element, SAOE = proportion of attempts * (fgp - $E$[fgp]) where fgp denotes field goal percentage. $E$[fgp] was the league average percentage in each circumstance. For example, $SAOE_{open}$ = proportion of threes which were open * (fgp on open threes - $E$[fgp on open threes]). After calculating the SAOE in each category, the sum represented the unadjusted SAOE. That is, $SAOE_{unadjusted}$ = $\sum_{i \in A}(SAOE_i)$ where A = (Contested, Open, Wide Open, Corner, ATB).

I am assuming that players attempt shots they can make. When examining the $SAOE_{unadjusted}$ for each player, there were many outliers. Some players were successful shooters with very few attempts. Under my assumption, these individuals are not good shooters. To correct for this, I adjusted the SAOE.

First, I grouped the data set by season. Then I normalized the number of three-point attempts each player attempted (subtracted the mean for the season and divided by the standard deviation). Each normalized variable was between -6 and 6, so I passed it through the logistic function to obtain a value between 0 and 1. Players who attempted more threes would have a value closer to 1. Players with low volume would have a value close to 0. Suppose the value from the logistic function was $r$. $r^2$ was used as a multiplier to adjust the SAOE for volume. Thus, $SAOE_{adjusted}$ is calculated as follows:

$$SAOE_{adjusted} = \begin{cases} SAOE_{unadjusted} - SAOE_{unadjusted} * r^2, & SAOE_{unadjusted} \leq 0 \\ SAOE_{unadjusted} * r^2, & SAOE_{unadjusted} > 0 \end{cases}$$

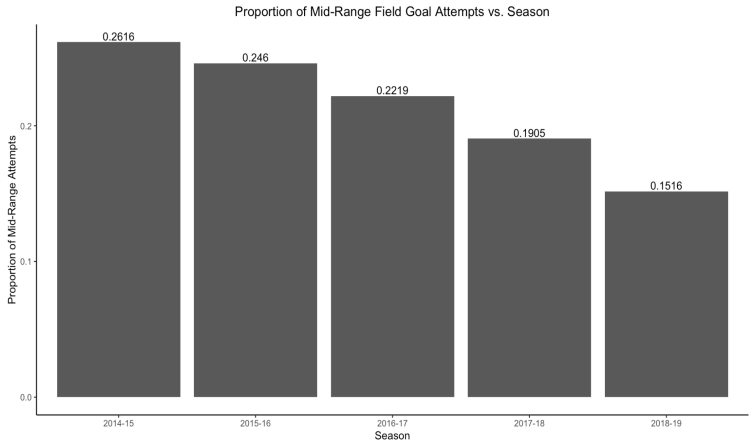
While players with low volume would see their component decrease, above-average shooters with high volume would increase in value. Also, below-average shooters before the adjustment would not be penalized for high volume.

| Name | Season | FG3A | Component |
|---|---|---|---|
| Kyle Korver | 2014-15 | 449 | 0.2637 |
| JJ Redick | 2015-16 | 413 | 0.2134 |
| Stephen Curry | 2015-16 | 884 | 0.2089 |
| Joe Harris | 2018-19 | 386 | 0.2059 |
| Stephen Curry | 2014-15 | 646 | 0.1931 |

Table 1: The top 5 3pt shooting seasons of the past 5 seasons

## 4.2 Mid-Range Scoring Component

In terms of points per possession, a mid-range shot is the least efficient shot in basketball. As three-point rates have increased, mid-range rates have decreased. However, having a player who can score from the mid-range area is extremely valuable late in games when a defense locks-in and eliminates high value looks for the offense. Frequently, the top scorers in the league are efficient from mid-range.




Similar to the three-point shooting component, I calculated the "Mid-Range Ability Over Expectation," *MRAOE*. This metric captures how efficient a player is in the mid-range area relative to league average. I did not adjust for defensive impact as most attempts are contested and volume has decreased in recent years.

The component is calculated as follows: MRAOE = proportion of shots which were MR attempts (MRfgp - E[MRfgp]). E[MRfgp] was equal to the league average for the given season. The metric was then adjusted for the volume of attempts. I normalized the attempts variable for each season. That value was then passed through a logistic function to obtain a value, $r$ which was between 0 to 1. Then $MRAOE_{adjusted}$ is calculated as follows:

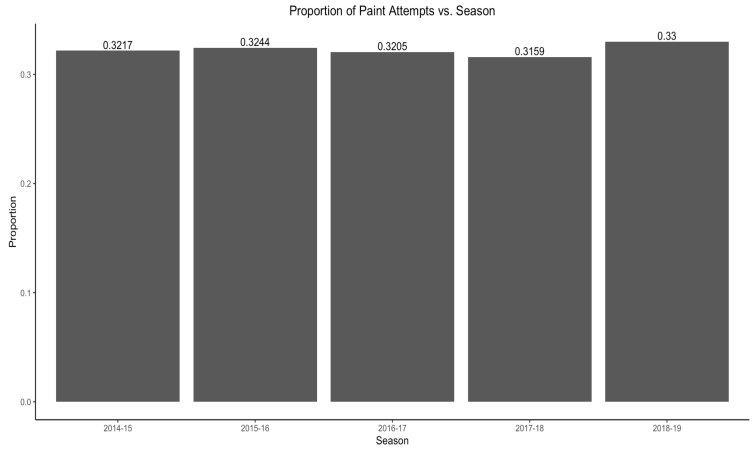
$$MRAOE_{adjusted} = \begin{cases} MRAOE_{unadjusted} - MRAOE_{unadjusted} * r, & MRAOE_{unadjusted} \leq 0 \\ MRAOE_{unadjusted} * r, & MRAOE_{unadjusted} > 0 \end{cases}$$

| Name | Season | FGMRA | Component |
|---|---|---|---|
| Kevin Durant | 2018-19 | 461 | 0.1466 |
| Stephen Curry | 2017-18 | 130 | 0.1232 |
| Al Horford | 2018-19 | 170 | 0.1109 |
| Mike Scott | 2017-18 | 158 | 0.1105 |
| Chris Paul | 2017-18 | 180 | 0.0999 |

Table 2: The top 5 Mid-Range Scoring Seasons over the past 5 seasons

### 4.3 Inside Scoring

Shots close to the basket are among the most valuable shots in basketball. Unlike three-point and mid-range attempts, the proportion of shots attempted within the paint has remained stable over the last five seasons.




The number of shots a player attempts and field goal percentage within the paint is dependent on one's "position" and role in the offense. Perimeter players are less likely to have a large proportion of their shots attempted close to the basket. Interior players are the opposite. I wanted the inside scoring component to reflect ability relative to similar players. Without adjusting for the position, "bigs" scored high, while guards scored low. Over the past five seasons, the NBA has been moving away from traditional "positions." The position assigned to each player was arbitrary.

Using lineup combinations from pbpstats.com, I determined which position a player played. There are five players on the court at a time. I ordered each

player 1 through 5 by height, with 1 denoting the shortest. Players who played $\geq$ 50% of their minutes at the "1" or "2" were considered guards. Players who played $\geq$ 50% of their minutes at the "4" or "5" were listed as "bigs." The rest were listed as "wings." While this provided a foundation, it was not perfect. I examined the position assigned to each player and made adjustments if warranted. In recent seasons, Draymond Green of the Golden State Warriors was considered a wing because he is 6'7". However, he functions as a big on Golden State.

When calculating finishing ability over expectation for each player, $FAOE$, the data set was first filtered by his positional designation. The $FAOE$ was composed of multiple elements. I calculated each players $FAOE_P$, $FAOE_{RA}$, and $FAOE_C$ where $P$ = Paint (Non restricted-area), $RA$ = Restricted Area, and $C$ = Off of Cut. Each FAOE component was based upon the same formula. For a given $X$, $FAOE_X$ = Proportion of shots from $X$ * ($FGP_X$ - E[$FGP_X$]). Note that E[$FGP_X$] is the expected field goal percentage on the shot type given season and position. A player's unadjusted $FAOE$, denoted by $FAOE_{unadjusted}$, is equal to $\sum_{i \in A}(FAOE_i)$ where A = (P, RA, C). Like the three-point shooting and mid-range scoring component, the finishing component was adjusted for volume. I normalized the total paint attempts variable before passing it through a logistic function to obtain a value, $r$. Then $FAOE_{adjusted}$ was computed as follows:

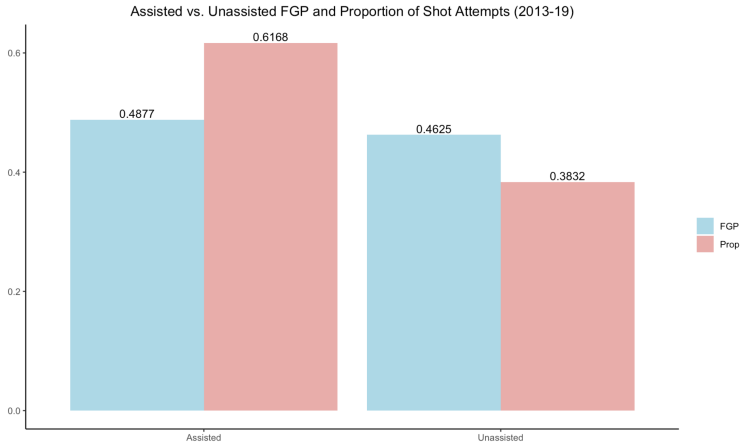
$$FAOE_{adjusted} = \begin{cases} FAOE_{unadjusted} - FAOE_{unadjusted} * r, & FAOE_{unadjusted} \leq 0 \\ FAOE_{unadjusted} * r, & FAOE_{unadjusted} > 0 \end{cases}$$

| Name | Season | Paint FGA | Component |
|---|---|---|---|
| DeAndre Jordan | 2016-17 | 573 | 2.5493 |
| DeAndre Jordan | 2015-16 | 504 | 2.5346 |
| LeBron James | 2016-17 | 737 | 2.3288 |
| Nikola Jokic | 2017-18 | 541 | 2.2951 |
| Giannis Antetokounmpo | 2018-19 | 930 | 2.2847 |

Table 3: The top 5 Finishing Seasons over the past 5 seasons

## 4.4 Playmaking

An elite playmaker directs the offense and generates open shots for teammates. When playing alongside one, teammates usually outperform their ability. Great playmakers are known for their decision-making and communication ability. The majority of field goal attempts come off of a pass. Thus, an exceptional playmaker is a necessary component of an elite offense.




To quantify the playmaking ability of a player, I used the following statistics from pbpstats.com: passes, potential assists, assists, and live-ball turnovers. A pass refers to the event where a player has the ball and gives the ball to a teammate. A player is awarded a potential assist after passing the ball to a teammate who immediately shoots. If the shot is successful, the passer also receives an assist. A live ball turnover occurs when the defense steals the ball from the player without a stoppage in play.

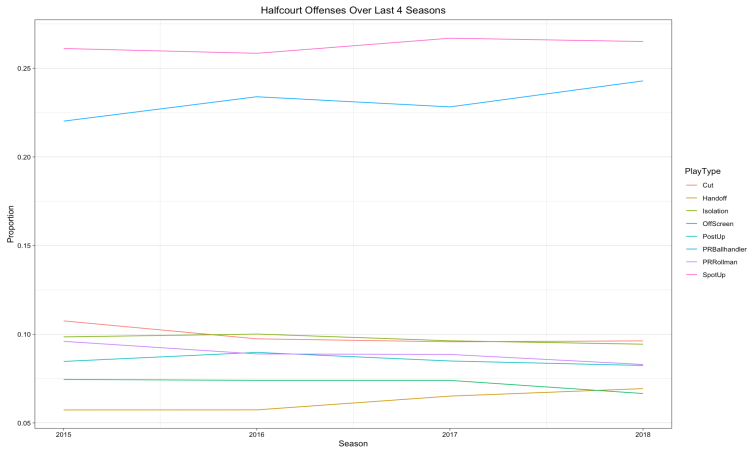
When creating the playmaking component, I adjusted for the position a player played. Otherwise, since guards typically initiate the offense, they would have an advantage. Given position and season, I made the following calculations. First, I calculated a player's potential assists to pass ratio and subtracted the average potential assists to pass ratio. This value will be denoted with $x$. Then, I calculated the player's live-ball turnovers per game and subtracted the average live-ball turnovers per game. This value will be denoted $y$. Summing $x$ and $y$ yielded the unadjusted passing component, denoted $P_{unadjusted}$. Next, I normalized the total assists variable. This value was passed through a logistic function to obtain a multiplier, $r$. Then $P_{adjusted}$ was computed as follows:

$$P_{adjusted} = \begin{cases} P_{unadjusted} - P_{unadjusted} * r, & P_{unadjusted} \leq 0 \\ P_{unadjusted} * r, & P_{unadjusted} > 0 \end{cases}$$

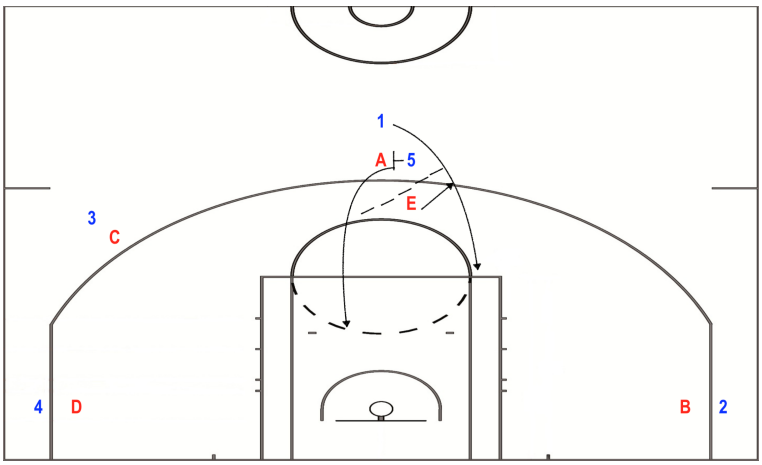
| Name | Season | Assists | Component |
|---|---|---|---|
| LeBron James | 2017-18 | 747 | 3.0151 |
| LeBron James | 2016-17 | 646 | 2.7921 |
| Ben Simmons | 2017-18 | 661 | 2.6616 |
| John Wall | 2016-17 | 831 | 2.3857 |
| Chris Paul | 2015-16 | 738 | 2.5652 |

Table 4: Top 5 Playmaking Components over past 5 Seasons

## 4.5 Roll Gravity




The pick-and-roll is a staple of any NBA offense. As floor spacing is emphasized, it has become one of the most popular half-court plays. While the ability of the ball-handler in the pick-and-roll is critical to success, an elite roller can put the defense in an extremely unfavorable situation. Below is a diagram of a high pick-and-roll. It is called "high" because it takes place at the top of the three-point arc.




In the diagram, the offense is blue and the defense is red. Player 1 is receiving a screen (pick) from Player 5. Player 1 will dribble around the screen probing Defender E's reaction. Suppose $E$ steps up and traps 1. Then 1 will pass the ball to 5 once he starts rolling toward the basket. Since $A$ will likely be behind 5, if 5 has the requisite ball-handler and finishing ability to make $A$

irrelevant, the rest of the defense will be in a dilemma. Defender $D$ or $B$ could come down to impede 5, but then a shooter in the corner will be left open. If neither defender bumps down, it will be a layup.

Now, suppose $E$ does not completely cut off $1's$ path to the rim and instead stays back in the paint. If $A$ is not quick enough getting around the screen and recovering to 1, 5 may be available for a layup or dunk. Defenders $D$ and $B$ will again be put in a bind. In both scenarios, the defense is compromised if 5 is a threat once he gets the ball. I attempted to quantify a player's "roll gravity" in the pick-and-roll.

To determine a player's roll gravity, I used pick-and-roll data from synergy sports technology dating back to the 2015-16 season. In particular, I focused on the points per possession generated when the roller received the ball. However, just examining the points per possession may be a bit misleading. Some rollers are the beneficiary of playing with an elite pick-and-roll ball-handler. This would likely cause the roller to outperform his ability. Also, rollers who play with weak shooters will be underrated. Passing to an open shooter who misses will decrease the points per possession.

I found synergy unreliable when differentiating a spot-up vs. pick-and-roll possession. If a roller passed to an open shooter who made a shot, sometimes it only counted as a spot-up possession. Thus, I assumed the playmaking aspect of roll gravity would be underrated if solely examining roll points per possession. For each season, I calculated the average roll points per possession. This was then subtracted from each player's roll points per possession to obtain the unadjusted roll gravity component. To modify the component to include a player's playmaking ability, I used the player's playmaking component value (**4.4**). Next, I adjusted for the shooting ability around each player since a pick-and-roll is more difficult with poor shooting. To do this, I calculated the three-point shooting percentage for each team. Then I normalized the variable. I passed the negative of each z-score through a logistic function to obtain a multiplier, $r$. The worst shooting team had the largest multiplier while the best had the smallest. Finally, the product of the playmaking component and $r$ was added to the unadjusted roll gravity component. This calculation represented the adjusted roll gravity component.

| **Name** | **Season** | **Component** |
|---|---|---|
| Clint Capela | 2017-18 | 3.3992 |
| DeAndre Jordan | 2016-17 | 2.8505 |
| Rudy Gobert | 2016-17 | 2.5914 |
| DeAndre Jordan | 2015-16 | 2.5234 |
| Al Horford | 2015-16 | 2.1843 |

Table 5: Top 5 Roll Gravity Components since 2015-16

## 4.6 Isolation

There are only 24 seconds for the offense to create a shot every possession. When the defense takes away the first two options, a team usually gets the ball to its best scorer to create a shot with limited time remaining. Elite shot creators can get a shot under any circumstance. They often require the toughest defensive match-up.

To quantify the shot-creating ability of a player, I used the isolation data from synergy and shot type data from nba.com/stats. The following calculations occurred after players were grouped by season and position (guard, wing, big). First, I calculated each player's isolation points per possession and subtracted the league average points per possession. Then I multiplied the value by the proportion of player possessions considered isolation. This product will be denoted with $x$. Next, I calculated each player's unassisted field goal percentage and subtracted the expected field goal percentage. Then I multiplied it by the proportion of player shot attempts considered unassisted. This product will be denoted with $y$. Let $iso_{unadjusted} = x + y$. To adjust for volume, I normalized the total isolation possessions variable. Then I placed the value through a logistic function. This would be the multiplier, denoted $r$. Finally, $iso_{adjusted}$ was computed as follows:

$$iso_{adjusted} = \begin{cases} iso_{unadjusted} - iso_{unadjusted} * r, & iso_{unadjusted} \leq 0 \\ iso_{unadjusted} * r, & iso_{unadjusted} > 0 \end{cases}$$

| Name | Season | Isolation PPP |
|---|---|---|
| James Harden | 2017-18 | 1.2181 |
| LeBron James | 2017-18 | 0.9597 |
| James Harden | 2018-19 | 1.1054 |
| LeBron James | 2016-17 | 0.9735 |
| Kyrie Irving | 2016-17 | 1.1236 |

Table 6: Top 5 Isolation Scoring Seasons since 2015-16 with Points per Possession

## 4.7 Rebounding

Rebounds have never been easier to accumulate due to several factors. The pace of play has consistently increased leading to more possessions per game. Offenses do not try to rebound their misses as often, instead opting to prevent transition opportunities for their opponent. Nevertheless, an elite rebounder maintains value if he improves the team rebounding percentage. Also, if a team has guards and wings that are good rebounders, it allows them to "play small," creating advantages on the offensive end.

To calculate the rebounding component, I examined offensive rebounds, defensive rebounds, and rebounds off of missed free throws. Since rebounds via

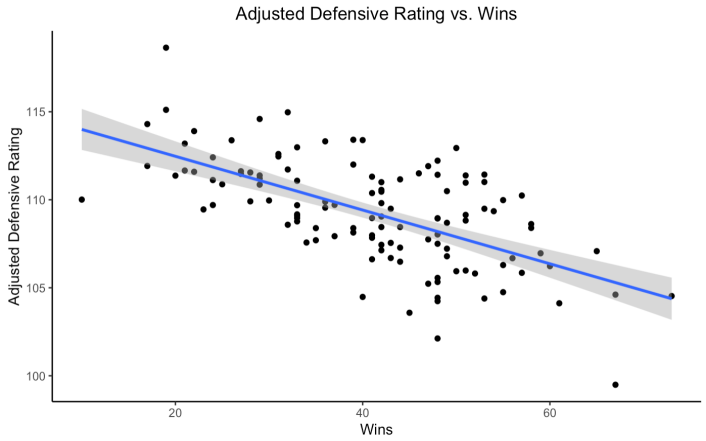
free throws are usually uncontested, I subtracted them from defensive rebounds for each player. Next, I normalized the new defensive rebounds variable and the offensive rebounds variable. Then I placed them both through a logistic function. Suppose the 2 values obtained were $x$ and $y$ respectively. The rebounding component, $R$, is equal to $\frac{1}{3}\ x + \frac{2}{3}\ y$. The normalized offensive rebounding element was multiplied by a larger proportion because they are more valuable than defensive rebounds. An offensive rebound is effectively an additional possession.

| Name | Season | DRPG | ORPG |
|---|---|---|---|
| Andre Drummond | 2016-17 | 9.5 | 4.3 |
| Andre Drummond | 2017-18 | 10.9 | 5.1 |
| DeAndre Jordan | 2017-18 | 10.9 | 4.3 |
| Rudy Gobert | 2016-17 | 8.9 | 3.9 |
| Tristan Thompson | 2015-16 | 9.0 | 5.7 |

Table 7: Top 5 Rebounding Seasons since 2015-16

## 4.8 Defense

Although individual defense is difficult to measure, teams with more effective defenses tend to win more games. The graph below illustrates the relationship between adjusted defensive rating and team success (wins).




Note that team adjusted defensive rating represents the number of points allowed per 100 possessions. There is some evidence of a linear relationship between the two variables.

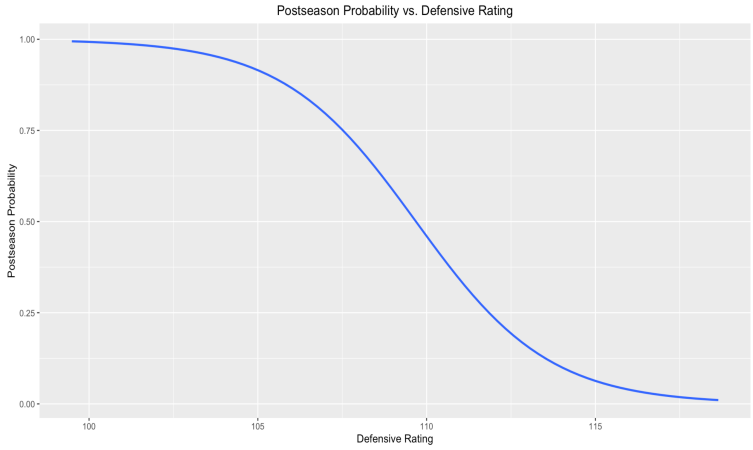
```
Call:
lm(formula = Adjusted.DRtg.A ~ W, data = teamstats)

Residuals:
    Min      1Q  Median      3Q     Max
-6.0746 -1.5248 -0.1999  1.7331  6.0180

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 115.52273    0.75525 152.959  < 2e-16 ***
W            -0.15267    0.01766  -8.645 3.12e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.353 on 118 degrees of freedom
Multiple R-squared:  0.3878,    Adjusted R-squared:  0.3826
F-statistic: 74.73 on 1 and 118 DF,  p-value: 3.118e-14
```

It is extremely difficult to reach the postseason with a poor defense:



```
Call:
glm(formula = Postseason ~ Adjusted.DRtg.A, data = teamstats)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.94926  -0.36601   0.03639   0.39095   0.78638

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    10.03441    1.43838   6.976 1.89e-10 ***
Adjusted.DRtg.A -0.08696    0.01316  -6.608 1.18e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.1847454)

    Null deviance: 29.867  on 119  degrees of freedom
Residual deviance: 21.800  on 118  degrees of freedom
AIC: 141.88

Number of Fisher Scoring iterations: 2
```

The impact that an individual player has on defense is difficult to measure. Since I was unable to obtain quality defensive data, I used 538's "Raptor" defensive rating to represent a player's defensive impact.

| Name | Season | Raptor Defense |
|---|---|---|
| Draymond Green | 2017-18 | 6.4 |
| Rudy Gobert | 2017-18 | 5.6 |
| Draymond Green | 2015-16 | 5.5 |
| Draymond Green | 2016-17 | 5.4 |
| Tim Duncan | 2016-17 | 5.2 |
| Kawhi Leonard | 2015-16 | 4.8 |

Table 8: Most valuable defensive players over the last 4 seasons according to Raptor

# 5 Results

For each season (2015-16, 2016-17, 2017-18, 2018-19), I created 15 clusters in R.

### 2018-19

There were 423 player who appeared in at least 15 games during the 2018-19 season. 5 of the 15 clusters had an average VORP $> 0$. Traditional averages for players within the respective clusters:

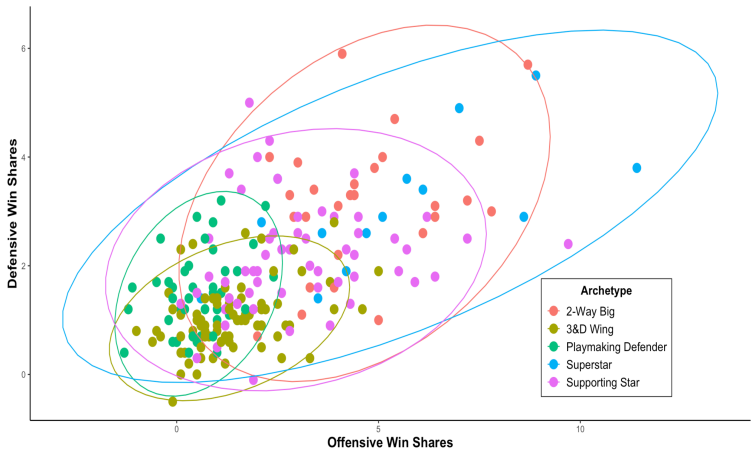
| Cluster | PPG | RPG | APG | SALARY |
|---|---|---|---|---|
| 3 | 10.5 | 2.9 | 2.6 | 7.4M |
| 4 | 15.2 | 9.5 | 2.4 | 15.8M |
| 8 | 23.6 | 7.1 | 5.0 | 24.5M |
| 12 | 14.2 | 4.6 | 3.8 | 16.6M |
| 14 | 8.0 | 3.9 | 2.1 | 7.3M |

Significant members of each cluster:

- **3:** Klay Thompson, Robert Covington, P.J. Tucker
- **4:** Joel Embiid, Anthony Davis, Rudy Gobert
- **8:** LeBron James, Kevin Durant, Kawhi Leonard
- **12:** Stephen Curry, Kyle Lowry, Kyrie Irving
- **14:** Draymond Green, Andre Iguodala, Lonzo Ball

Kevin Durant, Kawhi Leonard, Stephen Curry, Kyle Lowry, Draymond Green, Andre Iguodala, and Klay Thompson all participated in the NBA finals. Rudy Gobert won the defensive player of the year award. A short description classifying each cluster:

- **3:** Strong defensive wings who are among the best shooters in the league. These players are known as 3&D wings.
- **4:** Bigs who are elite defensively while also serving as an integral part of their respective offenses. Known as 2-Way Bigs.
- **8:** Players with few weaknesses. They can among the best players in the league. Known as superstars.
- **12:** Few weaknesses but not quite as impactful as superstars. Known as supporting stars.
- **14:** Role players who are good defensively while also serving as a secondary playmaker on offense. Known as Playmaking Defenders.




This graph displays defensive vs. offensive win shares for the members of the five clusters. Win shares attempt to assign credit to players for team success. Note that the diameter of the superstar cluster is large. These players are unique and affect the game in different ways. Note that Playmaking Defender and 3&D Wing clusters have relatively small diameters. These players fill a role on the team and their performance tends to be static. They are ideal to support higher variance star players.

## 2017-18

There were 419 players who appeared in at least 15 games during the 2017-18 season. 4 of the 15 clusters had an average VORP $> 0$. Traditional averages for players within the respective clusters:

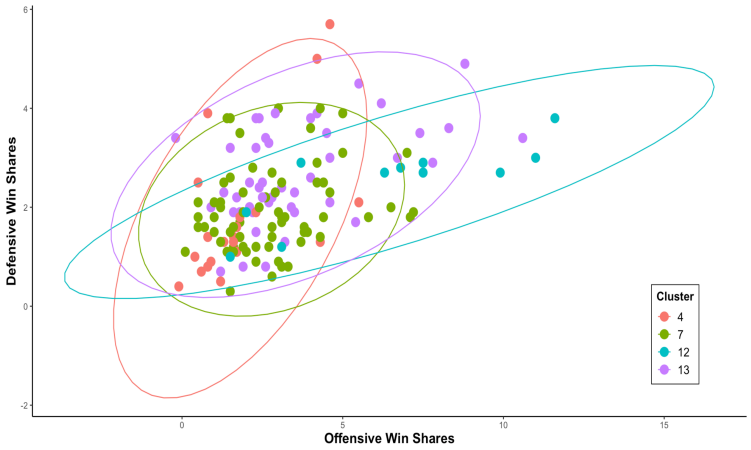
| Cluster | PPG | RPG | APG | SALARY |
|---|---|---|---|---|
| 4 | 7.6 | 5.3 | 1.6 | 6.7M |
| 7 | 13.3 | 4.0 | 3.1 | 10.5M |
| 12 | 23.3 | 5.2 | 5.9 | 21.3M |
| 13 | 13.1 | 7.7 | 2.3 | 10.1M |

Table 9: Clusters with Positive VORP: 2017-18

Significant members of each cluster:

- **4:** Ben Simmons, Bam Adebayo, Jusuf Nurkic
- **7:** Stephen Curry, Kyle Lowry, Kemba Walker
- **12:** LeBron James, Kevin Durant, James Harden
- **15:** Al Horford, Draymond Green, Pascal Siakam

Stephen Curry, LeBron James, Kevin Durant, and Draymond green participated in the NBA finals. James Harden won the MVP award. As a rookie, Bam Adebayo played less than 20 minutes per game. Being a member of a valuable cluster despite young age and low volume could have indicated that he would become an all-star less than two years later. Pascal Siakam made his first appearance in a high-value cluster despite being a reserve appearing in 20 minutes per game. The following season, he would become an all-star.




## 2016-17

There were 414 players who appeared in at least 15 games during the 2016-17 season. 4 of the 15 clusters had an average VORP $> 0$. Traditional averages for players in the respective clusters:

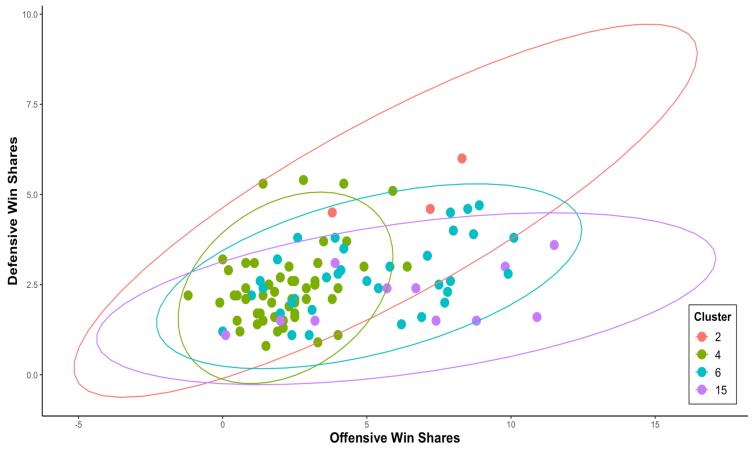
| Cluster | PPG | RPG | APG | SALARY |
|---|---|---|---|---|
| 2 | 12.9 | 11.8 | 1.2 | 12.5M |
| 4 | 10.8 | 6.6 | 1.4 | 7.6M |
| 6 | 19.1 | 6.1 | 4.6 | 14.3M |
| 15 | 22.8 | 5.7 | 5.5 | 17.4M |

Table 10: Clusters with Positive VORP 2016-17

Significant members of each cluster:

- **2:** Rudy Gobert, Steven Adams, DeAndre Jordan
- **4:** Draymond Green, Anthony Davis, Myles Turner
- **6:** Stephen Curry, Kevin Durant, Russell Westbrook
- **15:** LeBron James, Kyrie Irving, Isaiah Thomas

Stephen Curry, Kevin Durant, Draymond Green, LeBron James, and Kyrie Irving participated in the NBA finals. Green also won the defensive player of the year award. Russell Westbrook won the MVP award, while Isaiah Thomas finished third in the voting.




## 2015-16

There were 405 players who appeared in at least 15 games during the 2015-16 season. 5 of the 15 clusters had an average VORP $> 0$. Traditional averages for players in the repective clusters:

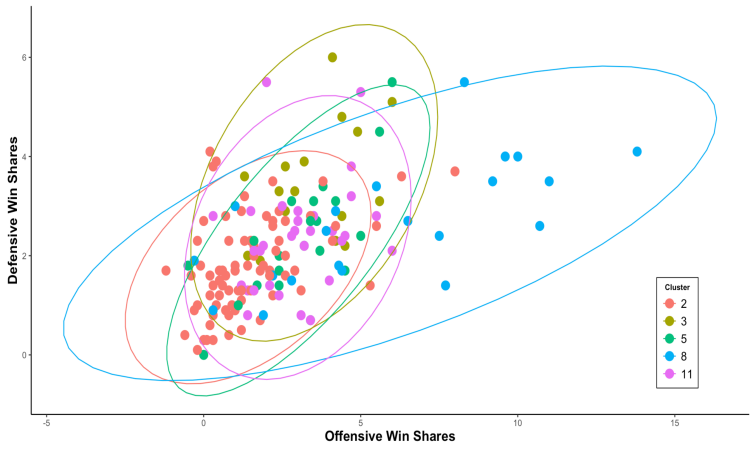
| Cluster | PPG | RPG | APG | SALARY |
|---|---|---|---|---|
| 2 | 8.9 | 3.3 | 2.2 | 4.7M |
| 3 | 13.7 | 8.3 | 2.7 | 9.3M |
| 5 | 9.9 | 7.3 | 1.3 | 9.9M |
| 8 | 20.6 | 6.1 | 4.9 | 13.2M |
| 11 | 12.3 | 7.1 | 1.4 | 7.1M |

Table 11: Top 5 Clusters by VORP 2015-16

Significant members of each cluster:

- **2:** Danny Green, Wesley Matthews, Klay Thompson
- **3:** Draymond Green, Anthony Davis, Nikola Jokic
- **5:** DeAndre Jordan, Clint Capela, Dwight Howard
- **8:** LeBron James, Stephen Curry, Giannis Antetokounmpo
- **11:** Kevin Love, Serge Ibaka, Aaron Gordon

Stephen Curry, Klay Thompson, and Draymond Green were the core of the 73 win Golden State Warriors. This set an NBA record for wins in a season. Curry won the first unanimous MVP award. However, Golden State lost in the finals to the Cleveland Cavaliers led by LeBron James. Love was also a member of the Cavaliers. Nikola Jokic was a rookie playing 21 minutes per game. Since then, he has developed into a 2-time all-star. This was the year before Giannis Antetokounmpo began his current streak of four straight all-star appearances. He won the MVP award in the 2018-19 season.




# 6 Conclusion

Classifying every cluster is a matter of semantics. That is why I only included a description of each group for the 2018-19 season. Other than a cluster containing superstars for each season, there are many ways to describe groupings. Some major takeaways:

- For a team to be a championship contender, it must have at least one player who is performing at a superstar level. Every team that participated in the NBA finals over the last four years had at least one player included in the superstar archetype.
- Besides 2015-16, there was not one cluster containing strictly interior-oriented players. Perhaps versatile bigs with playmaking ability are more valuable now than pure rim-runners.
- The cluster analysis appeared to recognize future all-stars early in their careers.
- Nearly every player across all clusters had a positive offensive and defensive rating. It is important to have "2-Way" players when building a team.

- Low usage players with one offensive skill (shooting or playmaking) and defensive ability are the most valuable "role-players." These are the types of players teams should be targeted when filling a roster.

# 7 Future Research

There are infinitely many versions of the proximity matrix for this analysis. If I use different components or transform any of them, the results could be significantly different. There is also an abundance of data available on the web. I only used a small portion in my analysis.

In the future I would like to link seasons together. The hierarchical clustering analysis was performed separately on all four seasons. This led to having groups that were essentially meaningless year-to-year. For example, if stars are in cluster *12* one year then *7* the next, cluster descriptions require constant redefinition. I am also interested in modeling the career trajectory of a rookie. Similar archetypes appear on championship teams over the years. When drafting, teams must identify the potential of a player before acquiring him. How do we determine what type of player someone will be three years from now? This may only be possible if I obtain more data or adjust the current vector components. It is unlikely the current sample size of four seasons is large enough to have predictive power.

# Appendix

All code and terminology used in this project is found here: https://github.com/psimpson625/NBAHC

# References


[1] "Calculating Win Shares: College Basketball at Sports." Reference.com, www.sports-reference.com/cbb/about/ws.html.

[2] Fichman, Mark, and John Robert O'Brien. "Optimal Shot Selection Strategies for the NBA." *Journal of Quantitative Analysis in Sports*, vol. 15, no. 3, 2019, pp. 203–211., doi:10.1515/jqas-2017-0113.

[3] "Hierarchical Clustering." *Cluster Analysis*, by Brian Everitt, Wiley, 2011.

[4] "NBA Stats." *NBA Stats*, stats.nba.com/.

[5] "pbpstats_client." *pbpstats_client*, www.pbpstats.com/.

[6] "The Elements of Statistical Learning: Data Mining, Inference, and Prediction." *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, by Trevor Hastie et al., Springer, 2017, pp. 501–528.

[7] Yu-Han Chang, Rajiv Maheswaran, Jeff Su, Sheldon Kwok, Tal Levy, Adam Wexler, Kevin Squire. *Quantifying Shot Quality in the NBA*, 2014