# The Impact of NBA Player's Performance on Their Salary

*Final Submission*


**Group 1:**

**Jishen Wang**

**David Wiley**

**Scott McCauley**

Texas Tech University


ISQS-6350-D01 Multivariate Analysis

Alireza Sheikh-Zadeh, Ph.D.

*December 5th, 2020*

# Table of Contents

# Introduction and Problem Context

## Introduction (Author: Scott)

Ever since professional baseball has gravitated towards data analytics for better decision making, the National Basketball Association (NBA) soon followed. The use of cameras now record every movement of both the ball and all 10 players 25 times per-second (Merrimack College, 2020). Scoring is another area affected by NBA analytics, where each individual player's free throw percentage and field goal locations are all closely analyzed in order to improve shooting form. Finally, NBA analytics plays a large role in assessing player matchups. By analyzing the strengths and weaknesses of their players and opponents, teams can strategize to place their players in favorable matchups. Our group will delve into this new and exciting field of analytics by performing various multivariate procedures on a dataset of NBA player performance metrics.

The dataset was downloaded from the Kaggle[1] website. The data consists of various player statistics extracted from the 2018-19 NBA season. The dataset includes twenty-seven total variables, however we will only use 15 of them for the sake of this analysis. We have broken down these variables into three main categories:

- Player description - Name, Height, Weight
- Player value - Salary
- Player performance - Points, Rebounds, Assists, Steals, Blocks, FT, FT%, FG, FG%, FG3, FG3%.

For a thorough description of each variable along with their variable types, see Appendix A.

## Project Approach (Author: David)

Prior to performing any project analysis, we will clean and process the NBA dataset. Specifically, we extract the variables of interest, impute missing values, and scale the data for analysis. These changes are detailed in the Data Preparation section.

After cleaning the data, we will explore the dataset using an initial exploratory analysis that consists of creating a correlation matrix and several histograms and scatter plots to show variable distributions and relationships between variables. Selected plots that provide insight for our multivariate analysis are shown in the Exploratory Analysis section.

Next, we will perform three multivariate procedures (dimension reduction, clustering, and Confirmatory Factor Analysis (CFA)) on the dataset to better understand which performance variables are highly correlated. Specifically, dimension reduction using Exploratory Factor Analysis (EFA) was performed to reduce our performance variables into a smaller set of frequent playstyles that still explain a majority of the variability within the dataset. Clustering was performed using three separate algorithms to group similar players together, and CFA was performed to test our EFA hypotheses.

Additionally, we will categorize the players by salary into three classes: high, medium, and low. Our team will then conduct multivariate techniques on each of these salary groups in order to better understand how player salary is impacted by player performance.

## Motivation (Author: Jishen)

Our team's motivation towards pursuing this project is centered around a strong interest in sports and a curiosity towards how analytics has influenced basketball and how it could continue to transform the sport. We believe that our project can potentially create value for NBA players, coaches, staff, and owners in many ways, which are specified below.

Performing multivariate analysis on these statistical measurements provides insight into the overall value of NBA players for the coaches, staff, and owners. In the NBA there is a time period called Free Agency where teams can sign any eligible players that are not under contract during that time. Analyzing a player's performance using exploratory factor analysis can be useful towards identifying playstyles (factors)

that account for the most variation within each salary category. Teams can use the results to assist in assigning value estimates to players and then deciding how much they are willing to invest in each player.

Another application to NBA analytics is projecting growth for younger NBA players. Using clustering techniques, NBA teams will be able to group players together and identify players with similar playstyles and variable measurements. Teams can then identify younger players of interest and correspond them to older more experienced players in their cluster. Finally, teams can identify the growth and performance of the older players to estimate the potential growth and improvements for the younger players.

# Data Preparation and Visualization

## Data Preparation (Author: David)

As mentioned in the Project Approach section, our first step in this project will be to clean our dataset and transform the data into different formats for multivariate analysis. The original dataset is saved as a dataframe under the 'nba' variable. We first notice that some values in the 'Salary' column of the dataframe do not have a numerical value, and instead have a "-" symbol. This likely indicates that these individuals were cut from their team and thus do not have a known salary value. Therefore, we impute all "-" values to zero and change the salary variable to numeric format.

```
nba$Salary[nba$Salary=='-'] = 0
nba$Salary = as.numeric(nba$Salary)
```

After performing this, we next extract only the 14 variables of interest for our analysis, as mentioned in the introduction. The process for this is shown below.

```
#Extract variables of interest
nba = nba[,c(2,3,6,7,8,9,10,11,12,13,14,15,16,17)]
```

Instead of extracting the 15th variable, which is player names, we had used these as observation labels when importing the data. Next, we will scale the overall

dataset to standardize all measurement units and calculate the distance matrix to input into hierarchical clustering.

```
scaled = scale(nba)
dists = dist(scaled)
```

Finally, we will perform outlier detection by taking the Mahalanobis distance of all players to discover the existence of any outliers in the data. After calculating the Mahalanobis distances, we sorted the data and observed the players with the largest and smallest values to see if there were any that stood out from the rest.

```
m_dist <- mahalanobis(nba_orig, data.center, data.cov, tol=1e-20)
nba_orig$M_Dist <- round(m_dist, 1)
head(rev(nba_orig[order(nba_orig$M_Dist, decreasing = T),])[1])
head(rev(nba_orig[order(nba_orig$M_Dist, decreasing = F),])[1])
```

| Player | Distance |
|---|---|
| Terrence Jones | 319.0 |
| Scott Machado | 149.0 |
| James Harden | 111.4 |
| Russell Westbrook | 93.3 |

| Player | Distance |
|---|---|
| Timothe Luwawu-Cabarrot | 3.0 |
| James Johnson | 3.5 |
| Bogdan Bogdanovic | 3.8 |
| Tyler Johnson | 3.8 |

**Table 1. Four Players With Highest and Lowest Distances**

From observing the output, we can see that Terrence Jones stands out to be an outlier, as the distance calculated for him is twice the distance of the next largest player. Therefore, we remove Terrence Jones from the dataset as his data point can greatly skew the results. We also consider removing Scott Machado, but leave him in the analysis for now.

## Exploratory Analysis (Author: Jishen)

As an initial exploratory analysis on our dataset and to evaluate its potential for multivariate analysis, our team constructed a correlation matrix on seven of the most relevant variables in the dataset. The correlation matrix is shown in Figure 1.

```
knitr::kable(round(cor(nba),2))
```

| | Height | Salary | Points | Blocks | Steals | Assists | Rebounds |
|---|---|---|---|---|---|---|---|
| Height | 1.00 | 0.05 | -0.01 | 0.50 | -0.14 | -0.36 | 0.49 |
| Salary | 0.05 | 1.00 | 0.63 | 0.30 | 0.51 | 0.51 | 0.50 |
| Points | -0.01 | 0.63 | 1.00 | 0.39 | 0.63 | 0.66 | 0.64 |
| Blocks | 0.50 | 0.30 | 0.39 | 1.00 | 0.31 | 0.09 | 0.70 |
| Steals | -0.14 | 0.51 | 0.63 | 0.31 | 1.00 | 0.66 | 0.47 |
| Assists | -0.36 | 0.51 | 0.66 | 0.09 | 0.66 | 1.00 | 0.32 |
| Rebounds | 0.49 | 0.50 | 0.64 | 0.70 | 0.47 | 0.32 | 1.00 |

**Figure 1: Correlation Matrix from the NBA Performance Dataset**

The matrix above has been rounded to two decimal places and highlights correlations over 0.5. We observe that blocks and rebounds are positively correlated (0.7) which supports our intuition since they both favor individuals of taller stature such as the Center and Power Forward positions. While the correlations provide initial observations, we will conduct various analysis techniques throughout this project to test these hypotheses.

The next graphic that our team chose to plot is a histogram of all NBA player salaries for the season.

```
hist(nba$Salary/1000000, breaks = 25, xlab = 'Salary in Millions', main =
'Histogram of Player Salaries', xlim = range(0,40), col = 'lightblue')
```

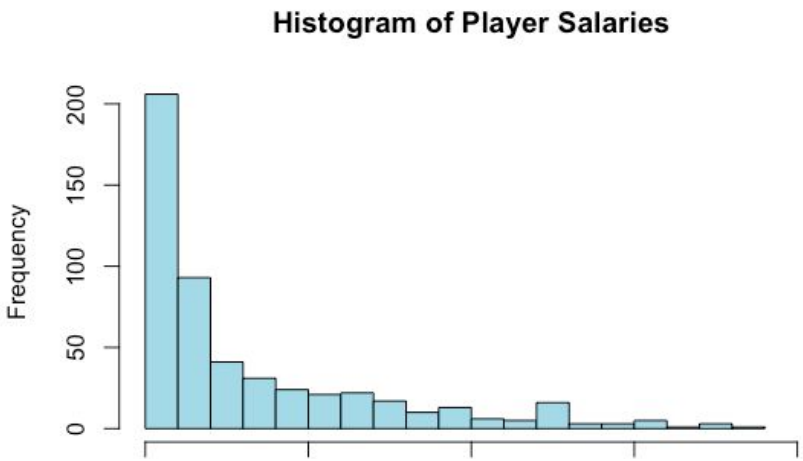


**Figure 2: Histogram of NBA Salaries**

After plotting this histogram, we can observe that the salaries are distinctly right skewed. While the majority of NBA players make less than 5 million dollars, we observe that the upper bound of salary is much higher. Some players are even valued at almost 40 million dollars for this season. One of our interests for this analysis will be to identify what performance factors influence salary and how players can distinctify themselves in order to obtain higher salaries. Our hypothesis is that the path to higher salaries is segmented and that there is no singular performance metric that determines salary, but rather multiple clusters of performance combinations which can each indicate higher salaries. In order to assess this, we will graphically observe relationships between variables using the R code and resulting scatter plots shown below.

```
nba$salgroup = cut(nba$Salary, b = c(0, 2000000,9000000,Inf), labels =
c('Low','Medium', 'High'))
#Scatter plot of assists vs steals
ggplot(nba, aes(x = Assists, y = Steals, color = salgroup))+geom_point()+
  ggtitle('Assists vs Steals Colored by Salary Class')
#Scatter plot of points vs rebounds
ggplot(nba, aes(x = Points, y = Rebounds, color = salgroup))+geom_point()+
  ggtitle('Points vs Rebounds Colored by Salary Class')
```

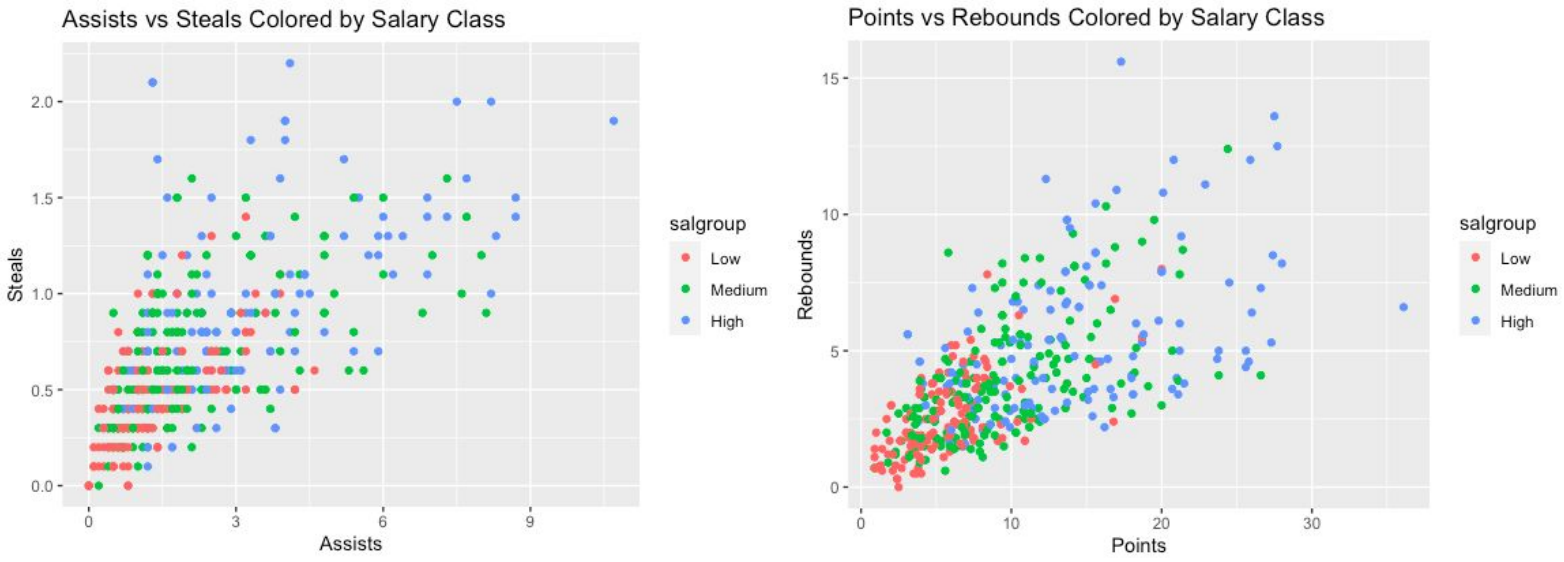



**Figure 3: Color Coded Scatter Plots by Salary**

The above plots display scatter plots of two sets of variables, color coded by salary group. The salaries have been grouped as ‘Low’, ‘Medium’ or ‘High’ using the below specifications:

- ‘Low’ salary group (shown in red): obtained salaries less than $2,000,000 this current season
- ‘Medium’ salary group (shown in green): obtained salaries from $2,000,000 - $8,000,000 this season
- ‘High’ salary group (shown in blue): obtained salaries greater than $8,000,000 for the current season

From the above plots we can observe that there is a general progression in salaries, from red to green to blue, in both scatter plots. This indicates that individuals can obtain higher salaries if they obtain more steals and assists. However, players can also obtain higher salaries if they obtain more rebounds and steals. This supports our hypothesis that there is more than one method (combination of performance metrics) to obtain higher salaries.

## Dimension Reduction (Author: Jishen)

Our group will begin the multivariate analysis by exploring the effect of dimension reduction techniques on the dataset. Our goals in dimension reduction are to reduce the number of variables in our dataset and find new variables that represent common play styles across NBA players. The insights gained from dimension reduction can be applied to real NBA analytics problems such as player value projection in Free Agency and the NBA Draft, as detailed in the “Motivation” section of this report.

While both PCA and EFA techniques were explored in our analysis, only the EFA dimension reduction model and results will be presented in this report. Our group highlights the results from EFA due to the flexibility of factor loadings. In EFA, factor loadings are not unique, which allows us to utilize Varimax rotation to select specific factor loadings that provide the easiest interpretation across our analysis. All EFA models were built using the ‘factanal’ function from the R programming language.

## EFA On All Players

We first perform EFA on all players regardless of salary level. After scaling our dataset, we perform EFA on the scaled variables and specify four factors. In order to ensure that four factors were sufficient, we calculated the root-mean-squared error (RMSE) between the approximate and actual correlation matrix.

```
corHat = loading %*% t(loading) + diag(efa$uniquenesses)
corTrue = cor(scaled)
rmse = sqrt(mean((corHat-corTrue)^2))
```

We obtained a RMSE of 0.04 which states that four factors are sufficient. The resulting factor loadings are displayed below, only loadings above 0.5 are shown.

```
efa = factanal(scaled, factors = 4)
print(efa$loadings, cut = .5)
```

```
Loadings:
         Factor1 Factor2 Factor3 Factor4
Height    0.885
Weight    0.834
Points            0.685           0.515
Blocks    0.609
Steals                    0.802
Assists                   0.632
Rebounds  0.646
FT%
FTA               0.764
FG3%
FG3A                            0.876
FG%
FGA               0.620   0.519   0.563
```

**Figure 4: EFA Factor Loadings**

We observe that our EFA outputs four factors that each represent general sets of physical and basketball performance metrics prevalent across NBA players. Players that have a high score for a particular factor are generally defined by the associated playstyle that the factor represents. These factors will be interpreted in more detail below:

- **Factor 1, the rebounder –** Factor 1 is characterized by large values for height and weight, indicating players of generally larger stature. Furthermore, Factor 1 also has large values for blocks and rebounds. Therefore, players with a large Factor 1 score tend to be larger and obtain many blocks and rebounds. These players likely play in the power forward and center positions.
- **Factor 2, the scorer –** Factor 2 is characterized by the largest value in points across all factors, meaning that they generally score the most points. Additionally, Factor 2 also has large values for free throws attempted (FTA) and field goals attempted (FGA), indicating that this factor represents players that attempt a lot of shots as well.
- **Factor 3, the passer –** Factor 3 has the largest values for steals and assists across all factors. Individuals with large Factor 3 values generally pass the ball often and steal opponent's passes often as well. These players likely play in the point or shooting guard position.
- **Factor 4, the 3-point shooter –** Factor 4 represents a unique factor characterized by its large 3-point field goal attempt (FG3A), which is much larger than for any other factor. This indicates that players with a large Factor 4 value attempt many 3-point shots.

The above factors can be characterized as common play styles among all NBA players.

## EFA Separated By Salary Class

Next, we will identify common play styles in the high and low salary classes and compare these playstyles. In doing this, our team hopes to gain insight on performance distinctions between salary groups in order to assess what certain NBA players do differently to obtain higher salaries. The high salary class contains all players with salaries greater than $9 million, while the low salary class contains players with salaries less than or equal to $2 million. We compute EFA for both salary classes and display their factor loadings side-by-side below.

```
high = nba[nba$salgroup=='High',]
low = nba[nba$salgroup=='Low',]
efa1 = factanal(high, factors = 3)
print(efa1$loadings, cut = .4)
```

```
efa2 = factanal(low, factors = 3)
print(efa2$loadings, cut = .4)
```

**High Salary Class EFA:**

```
Loadings:
         Factor1 Factor2 Factor3
Height             0.905
Weight             0.865
Points   0.906
Blocks             0.618
Steals   0.607
Assists  0.706
Rebounds 0.445   0.722
FT%                       0.532
FTA      0.839
FG3%                      0.652
FG3A     0.456            0.716
FG%                0.585  -0.442
FGA      0.893
```

**Low Salary Class EFA:**

```
Loadings:
         Factor1 Factor2 Factor3
Height             0.891
Weight             0.742
Points   0.905
Blocks             0.636
Steals   0.591
Assists  0.649
Rebounds 0.456   0.625
FT%
FTA      0.666            0.445
FG3%
FG3A     0.806
FG%                       0.692
FGA      0.976
```

**Figure 5: EFA Factor Loadings for High and Low Salary Groups**

The above EFA models were both restricted to 3 factors and only values above 0.4 or below -0.4 were printed. While the factor loadings are generally similar across both groups, we can notice some slight differences.

We observe that the 3rd Factor differs in meaning between the two salary groups. The high salary-class Factor 3 loadings resemble a 3-point shooter with large values for 3-point field goal percentage (FG3%) and 3-point field goal attempts (FG3A), while the low salary-class group does not contain this pattern. Although there are other factors with large values for FG3A in both salary groups, no other factor contains a value above 0.4 for FG3%. This indicates that the individuals in the high-salary Factor 3 group are special since not only do they take a lot of three-point shots but they make a high percentage of them too. Therefore, a playstyle that appears to be more prevalent among higher salary NBA players is the 3-point shooter style which consists of individuals that take many 3-point shots and make them with high accuracy. These

insights were discovered using an exploratory point of view with EFA and will be further tested using Confirmatory Factor Analysis (CFA) in a later section.

## Clustering (Author: David)

Clustering allows us to classify players into groups based on the statistics and performance measures in the data. We will experiment with three separate clustering algorithms then compare and contrast the algorithms and cluster results.

### Hierarchical Clustering

The first clustering technique used is hierarchical clustering analysis, which takes the scaled distance matrix of the data and clusters the data points iteratively according to their distance. To find which linkage criteria worked best, we used three different methods within R: “single”, “average”, and “complete”, the results of each were then cut into three groups. The results of “single” linkage displayed clusters highly skewed towards Group 1 as shown in Table 3, we also observed a similar skew in the “average” linkage results. The results for “complete” linkage are shown in Table 4. Although there is still an imbalance between clusters, the results are far better than in “single” and “average” linkages.

```
hc1 <- hclust(dists, "single")
ct_single <- cutree(hc1, 3)
table(ct_single)
```

| Group | Players |
|---|---|
| 1 | 432 |
| 2 | 1 |
| 3 | 1 |

**Table 3. Tree Cut of “Single” Linkage**

```
hc2 <- hclust(dists, "complete")
ct_complete <- cutree(hc2, 3)
```

```
table(ct_complete)
```

| Group | Players |
|---|---|
| 1 | 322 |
| 2 | 96 |
| 3 | 15 |

**Table 4. Tree Cut of "Complete" Linkage**

## K-means Clustering

The second clustering technique we used was the k-means technique. We passed our scaled data and again used 3 groups, which was determined through the Scree plot shown in Appendix B. This technique resulted in more balanced player groupings as shown in Table 5.

```
km <- kmeans(scaled, centers=3, nstart=10)
table(km$cluster)
```

| Group | Players |
|---|---|
| 1 | 233 |
| 2 | 98 |
| 3 | 103 |

**Table 5. Groupings from K-Means**

To visualize the data, we incorporated the k-means results and the principal component analysis by plotting the first 3 components grouped by k-means clusters. The k-means groupings were used to identify the colors within the space for each group. The results of each plot are shown in Figure 6 below.

```
pca <- princomp(nba_orig, cor=T)
scores = data.frame(pca$scores[,1:3])
```

```
ggplot(scores,aes(x=Comp.1, y=Comp.2, color =
as.factor(km$cluster)))+geom_point()+labs(color='Clusters')+ggtitle('PC1 vs
PC2')
ggplot(scores,aes(x=Comp.2, y=Comp.3, color =
as.factor(km$cluster)))+geom_point()+labs(color='Clusters')+ggtitle('PC1 vs
PC3')
```

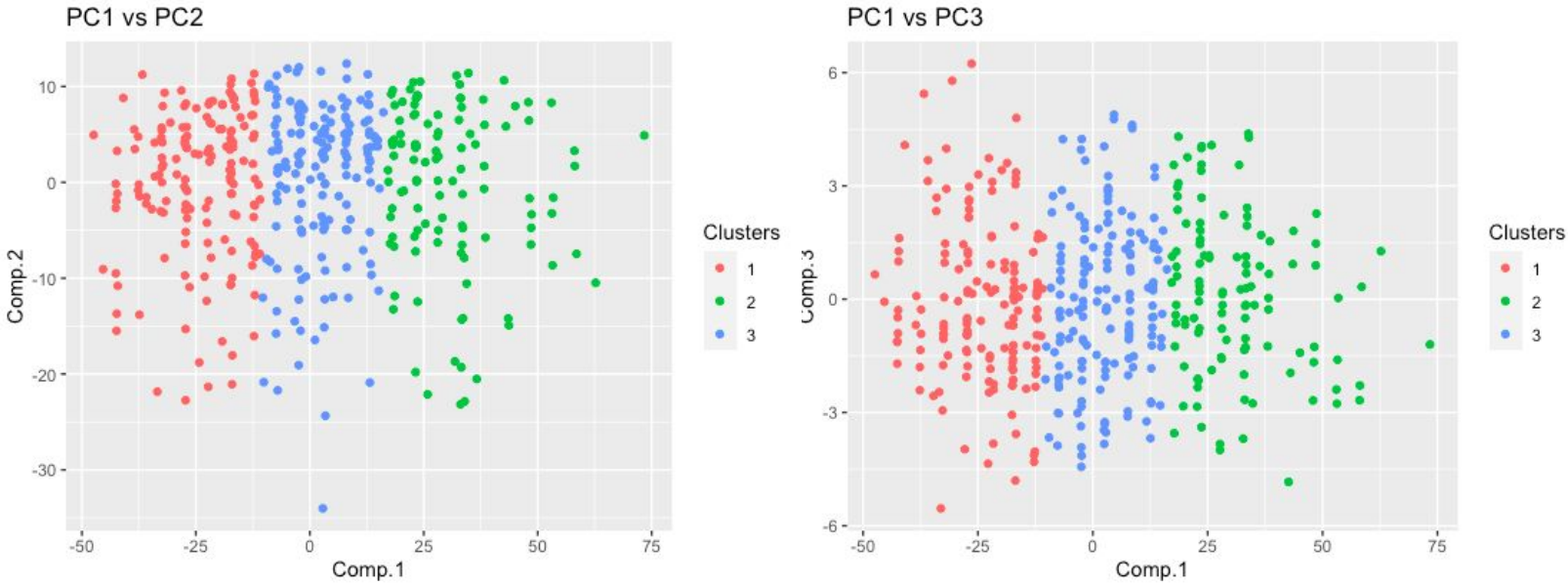



**Figure 6. PC1 vs. PC2. Plot**

We observe that there are 3 distinct groupings in the first plot using the first and second components, this is mostly because of the first component. There are also 3 relatively distinct groups when plotting the first and third components, largely due to component 1 again. When the second and third components were plotted against each other, it was difficult to discern where the groupings lie since there was a lot of overlap. Therefore, this plot was not displayed in the report. Overall, we observe that the K-means clusters are distinctly separated and defined by the first principal component.

Next, we will explore the cluster centroid values in order to further interpret the meaning of our 3 clusters.

```
knitr::kable(round(km$centers,2))
```

| Height | Weight | Points | Blocks | Steals | Assists | Rebounds | FT% | FTA | FG3% | FG3A | FG% | FGA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| -0.31 | -0.28 | 1.32 | 0.13 | 0.97 | 1.12 | 0.54 | 0.42 | 1.14 | 0.38 | 1.12 | 0.03 | 1.36 |
| -0.35 | -0.34 | -0.59 | -0.50 | -0.39 | -0.36 | -0.61 | -0.08 | -0.57 | 0.05 | -0.22 | -0.45 | -0.55 |
| 1.10 | 1.07 | -0.03 | 0.99 | -0.12 | -0.34 | 0.84 | -0.26 | 0.13 | -0.52 | -0.67 | 1.00 | -0.15 |

**Table 6: K-means Cluster Centroid Values**

Each row in the above table represents a cluster outputted from K-means clustering. We observe that the first cluster represents a group of NBA players that is below average in stature and well above average in points, steals and assists. The second cluster represents a group that is generally below average across all variables, this most likely represents a group that has less playing time. Finally, the third cluster represents a group with much larger stature than average, and that has above average block and rebound values.

## Model-based Clustering

The final clustering technique we used was model-based clustering using the `mclust` library available in R. Utilizing this technique increased the balance of players in each group as shown in Table 6.

```
mc <- Mclust(scaled, G = 3)
table(mc$classification)
```

| Group | Players |
|---|---|
| 1 | 182 |
| 2 | 103 |
| 3 | 149 |

**Table 6. Model-based Groupings**

We can observe the groupings by plotting the classification results from the model-based technique but with so many variables it is very condensed and would be difficult to view. However, we can visualize the uncertainty of players based on a couple

of variables, this plot is shown in Appendix B. Appendix B shows a closer look at the Free Throws and 3 Point Field Goals. There, we can see the most uncertain player classification is Scott Machado. We could do this with more variables but we limit ourselves to a single example just to get the idea.

When comparing these clustering methods against each other, we observe that Model-based clustering provides the most balanced results. However, K-means clustering provides both the fastest computation and the most in-depth interpretation capabilities. While the interpretation of results was limited for Hierarchical and Model-based clustering due to the large number of observations, we were able to extract valuable information from the cluster centroids and principal component plots through K-means clustering. Specifically, we observed from the K-means cluster centroids that the first cluster represented groups of players that were above average in almost all performance metrics but below average in size, the second cluster represented players with below average values in all performance metrics and likely were secondary players. Finally, the third cluster represented players above average in size, rebounds, and blocks which resembles the ‘Rebounder’ factor in the EFA result outcomes.

## Confirmatory Factor Analysis (Author: Scott)

Confirmatory Factor Analysis (CFA), according to “An Introduction to Applied Multivariate Analysis with R” (Everitt & Hothorn, 2011), may come up from theoretical considerations or be based on the results of an exploratory factor analysis (EFA) where the investigator might wish to postulate a specific model for a new set of similar data. In the case of our group, CFA is performed due to the latter reason and will be used to test the hypotheses and underlying factors developed in our EFA.

We performed two CFA models on the NBA dataset we had selected for this project. Within our original CFA model, we found that it was more comprehensive

across our variables and yet did not perform well due to the many uncertain variables which couldn't be grouped into a single factor. Within the second and more condensed CFA model we created, we found that with all the uncertain variables removed, we observed better "goodness of fit" measures.

## Original CFA Model

Our first CFA model was designed to be more comprehensive and representative of the results found in the EFA. We incorporated three underlying factors which encaptured all the variables except for FT% and FG%, which we found were too uncertain to add into our model. The model is detailed below, and also can be observed in the path diagram.

- **Factor:** Three-point shooter - **Variables:** FG3A, FG3%
- **Factor:** Rebounder - **Variables:** Height, Weight, Blocks, Rebounds
- **Factor:** Passer - **Variables:** FGA, FTA, Assists, Steals

We computed this outlined model using the 'sem' function in R, the model, estimates, and code are shown in Appendix D. This CFA model had an RMSE of 0.201, AGFI of 0.499, GFI of 0.708, and SRMR of 0.195. As we can observe from these resulting measures, the RMSE and SRMR metrics are not below 0.05, and the AGFI and GFI metrics are not above 0.95, so this factor model does not describe the data very well.

```
semPaths(nba_sem, rotation = 2, col = 2, "est")
```

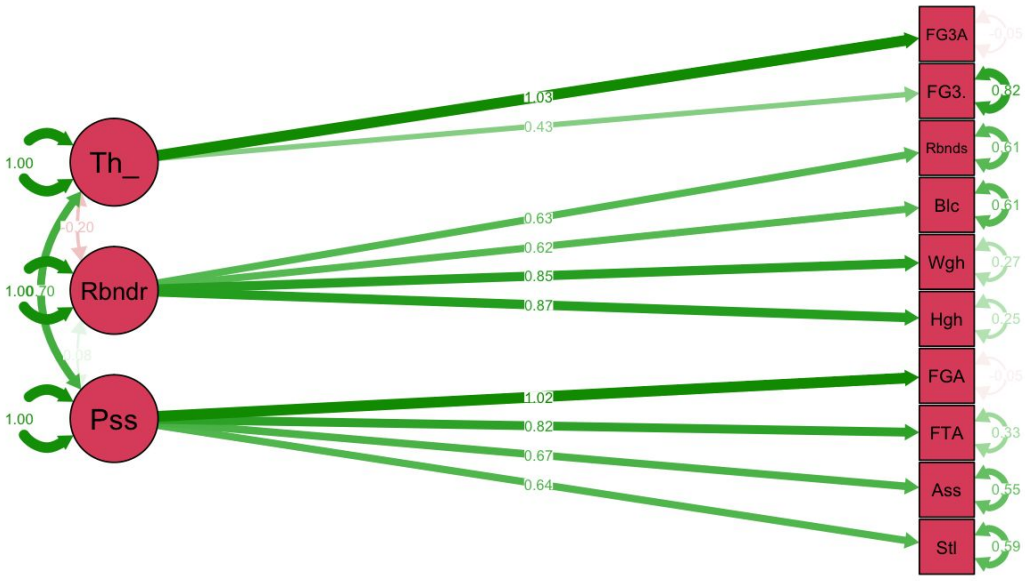


**Figure X: Path Diagram for the 1st CFA Model**

Although we had removed the FT% and FG% variables which we thought were too uncertain for CFA, we believe that there are many other uncertain variables in this model which we still assigned factors to. We believe that this is the reason that this CFA model did not perform well, therefore, we will construct a secondary model which is far more condensed and considers much fewer but more definitive variables.

## Condensed CFA Model

The second CFA model that we built has 2 underlying factors and only considers 6 variables. This model is outlined below and in the path diagram.

- **Factor:** Rebounder- **Variables:** Height, Weight, Blocks
- **Factor:** All-rounder - **Variables:** FG3A, Assists, Steals

We create a subset of our dataset with only these 6 variables and again perform CFA using the 'sem' function. The resulting model and estimates are shown in Appendix E. The second CFA model that we built had a RMSE of 0.137, GFI of 0.899, AGFI of 0.736, and SRMR of 0.127 . Although the factors fit our model better, they still do not confirm our model fits the data well. The resulting path diagram is shown below.

```
semPaths(nba_sem, rotation = 2, col = 2, "est")
```

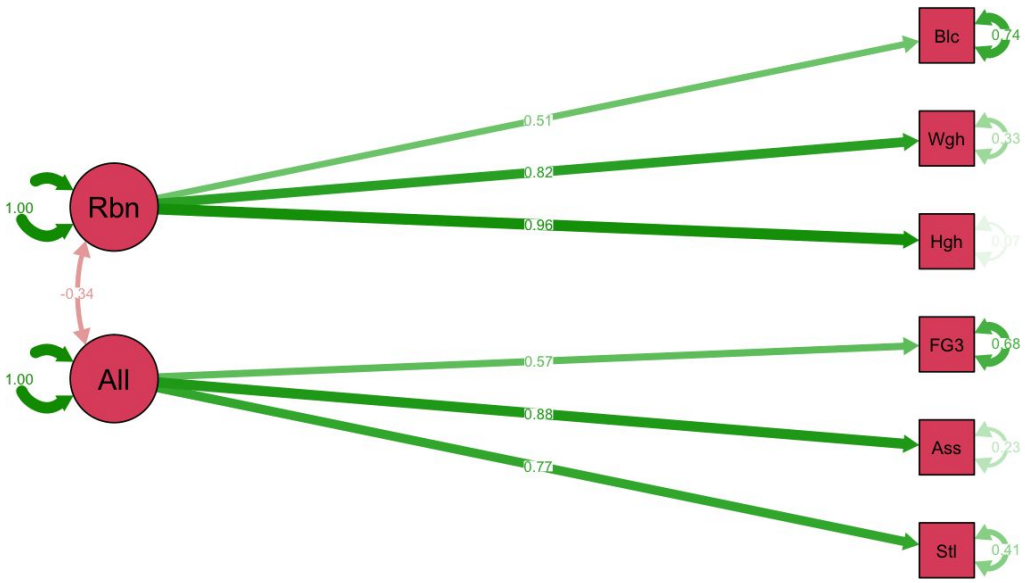


**Figure X: Path Diagram for the 1st CFA Model**

Although removing factors and uncertain variables from our model has improved our RMSE, GFI, AGFI, and SRMR metrics compared to the previous CFA model, these values are still below the sufficient requirements to confirm that our factor model accurately represents the data and that these general playstyles are the underlying factors which describe the performance metrics.

# Conclusion (Author: Jishen)

## Project Summary

Throughout this project, our team has conducted a multivariate analysis on NBA performance statistics in order to draw inference on general playstyles in the NBA. Our first step was to perform data cleaning and to initialize transformations that were utilized as inputs in later sections. Afterwards, our team performed an exploratory data analysis where the correlation matrix was calculated and several histograms and scatter plots were created, from these plots we observed that many performance statistics are positively correlated and that player salary generally increases as these performance statistics increase.

Next, our group performed dimension reduction through Exploratory Factor Analysis. In this section we observed that all performance metrics can be condensed into four underlying factors which highlight general playstyles. Furthermore, we compared the EFA models for high and low salary groups and observed that the 3-point shooter playstyle appears to be more prevalent in higher salary individuals. Afterwards, we performed clustering with three separate algorithms (hierarchical, K-means, model-based) in order to group similar players together and draw inference from these clusters. When comparing and contrasting the algorithms, we observed that although model-based clustering gave the most balanced clusters, K-means was the preferred algorithm due to its fast computation and easy interpretability.  When interpreting the K-means cluster centroids we observed that the three clusters can be separated as:

all-around above average players, all-around below average players, and larger players who excel in rebounding. Finally, our group performed Confirmatory Factor Analysis in order to test the underlying factors observed in the EFA. Generally, our team observed that our original EFA factor model did not perform well due to many uncertain variables that did not fit into a single factor. However, as we removed more of these uncertain variables and reduced the number of factors, we observed a better model fit measure.

## Pros and Cons

Overall, our study had many advantages and disadvantages. One advantage of our study is that it is comprehensive, the analysis was performed on the population of all NBA players in the season. The primary drawback of our study was that our CFA models did not reach acceptable goodness-of-fit levels (above 0.95 for GFI, and AGFI, and below 0.05 for SRMR and RMSE) in order to confirm our model hypothesis, even after removing uncertain variables and condensing our model. Another disadvantage of our study is that it is fixed in time and only encompasses a single season in the NBA, analyzing results across multiple seasons over time may provide more informative and accurate results.

## Future Considerations

One consideration for future projects would be to replicate this analysis with a larger number of performance metrics. Although there are far more performance metrics, such as VORP (Value Over Replacement Player) and PER (Player Efficiency Rating) available, our team did not consider any of these and only used the 15 basic metrics mentioned in the Introduction. More performance metrics may result in more accurate groupings when performing multivariate analysis. Now that we have gained a more thorough understanding of the relationships between variables, a follow-up analysis would be to predict NBA salaries based on the performance metrics that were considered. Therefore, performing a predictive analysis on NBA salaries is another future consideration which could be very valuable for both players and coaches in assessing value.

# References

[1] Schmadamco (2019). *NBA Regular Season Stats 2018-2019*. Retrieved from https://www.kaggle.com/schmadam97/nba-regular-season-stats-20182019/.

[2] Merrimack College (2020). *How NBA Analytics is Changing Basketball.* Retrieved from https://onlinedsa.merrimack.edu/nba-analytics-changing-basketball/

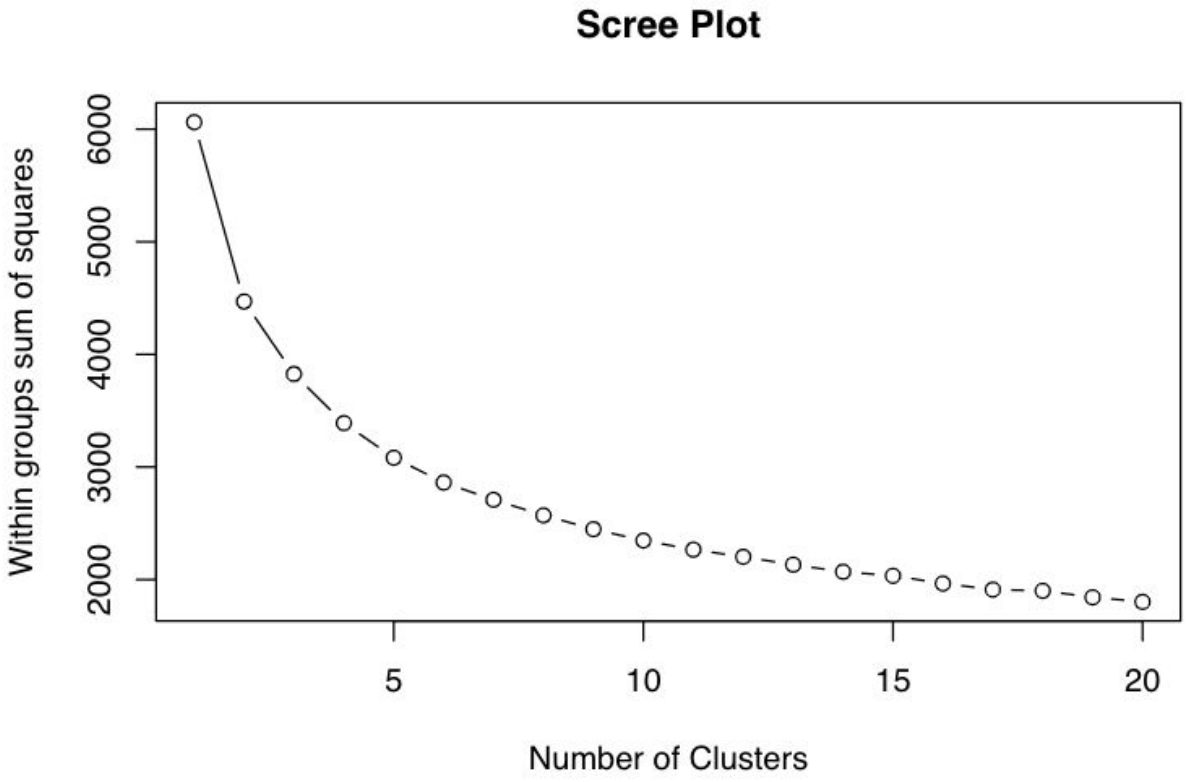
[3] Everitt, B. S., & Hothorn, T. (2011). *An introduction to applied multivariate analysis with R*. New York: Springer.

# Appendix A: List of Variables

| Variable | Description | Type |
|---|---|---|
| **Player Description** | | |
| Name | Name of the player | String |
| Height | Height of the player in inches | Inches |
| Weight | Weight of the player in pounds | Pounds |
| Team | Team the player played on | String |
| Age | Age of the player | Integer |
| **Player Value** | | |
| Salary | Yearly salary of the player | Integer |
| **Player Performance** | | |
| Points | Average number of points per game for the season | Decimal |
| Blocks | Average number of blocks per game for the season | Decimal |
| Steals | Average number of steals per game for the season | Decimal |
| Assists | Average number of assists per game for the season | Decimal |
| Rebounds | Average number of rebounds per game for the season | Decimal |
| FT% | % of freethrows made | Decimal |
| FTA | Average number of freethrows attempted | Decimal |
| FG3% | % of three point field goals made | Decimal |
| FG3A | Average number of three point field goals attempted | Decimal |
| FG% | % of field goals (excluding freethrows) made | Decimal |
| FGA | Average number of field goals (excluding freethrows) attempted | Decimal |

# Appendix B: K-Means Scree Plot

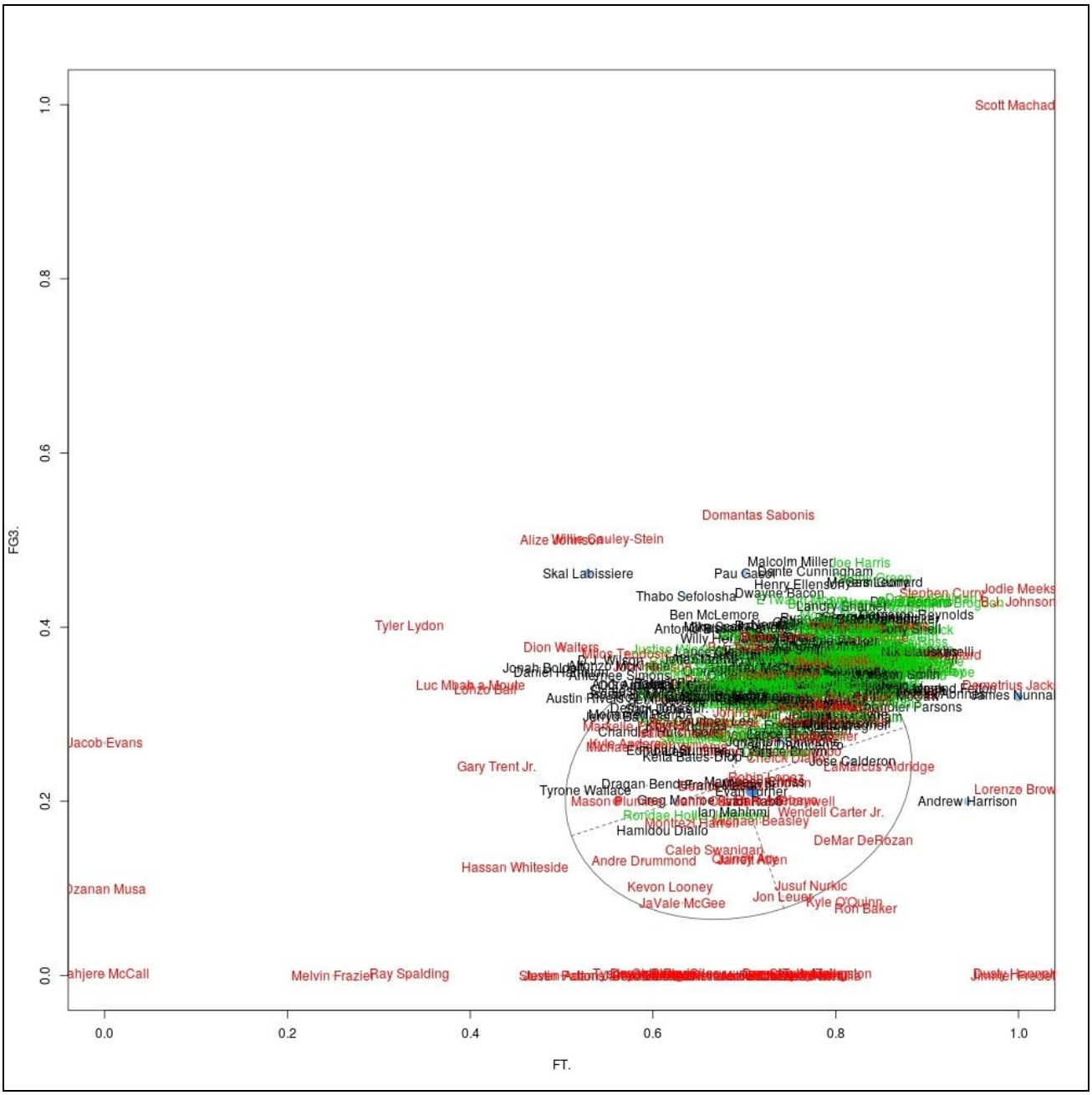
```
#Scree plot
plot.wgss = function(mydata, maxc) {
  wss = numeric(maxc)
  for (i in 1:maxc)
      wss[i] = kmeans(mydata, centers=i, nstart=10)$tot.withinss
  plot(1:maxc, wss, type="b", xlab="Number of Clusters",
  ylab="Within groups sum of squares", main="Scree Plot")
}

plot.wgss(scaled, 20)
```




*The scree plot appears to level off after 3 clusters. Therefore, we select 3 clusters for our K-means analysis.*

# Appendix C: Uncertainty Plot Between FT and FG3 Variables

```
plot(mc, what = "uncertainty", dimens = c(6,8))
text(mc$data[,c(6,8)], labels = rownames(mc$data), col = mc$classification)
```




*This plot shows uncertainty values outputted by Model-based clustering. The X-axis represents free throws while the Y-axis represents 3-point field goals.*

# Appendix D: Original CFA Model Code and Estimates

```
nba_omit = nba[,c(2,3,8,9,10,11,13,14,15,17)]
scaled = scale(nba_omit)
NBA_sem <- specifyModel(text = '
Rebounder -> Height, lambda1, NA
Rebounder -> Weight, lambda2, NA
Rebounder -> Rebounds, lambda3, NA
Rebounder -> Blocks, lambda4, NA
Passer -> Steals, lambda5, NA
Passer -> Assists, lambda6, NA
Passer -> FTA, lambda8, NA
Passer -> FGA, lambda9, NA
Three_shooter -> FG3., lamba10, NA
Three_shooter -> FG3A, lamba11, NA
Rebounder <-> Passer, rho1, NA
Three_shooter <-> Passer, rho2, NA
Three_shooter <-> Rebounder, rho3, NA
Height<-> Height, theta1, NA
Weight <-> Weight, theta2, NA
Rebounds <-> Rebounds, theta3, NA
Blocks <-> Blocks, theta4, NA
Steals <-> Steals, theta5, NA
Assists <-> Assists, theta6, NA
FTA <-> FTA, theta8, NA
FGA <-> FGA, theta9, NA
FG3. <-> FG3., theta10, NA
FG3A <-> FG3A, theta11, NA
Three_shooter <-> Three_shooter, NA, 1
Passer <-> Passer, NA, 1
Rebounder <-> Rebounder, NA,1')

options(fit.indices = c("GFI", "AGFI", "SRMR"))
nba_sem = sem(NBA_sem, cor(scaled), nrow(scaled))
summary(nba_sem)
##
##  Model Chisquare =  931.916   Df =  32 Pr(>Chisq) = 3.629914e-175
##  Goodness-of-fit index =  0.7123618
##  Adjusted goodness-of-fit index =  0.5056218
##  SRMR =  0.1919742
##
##  Normalized Residuals
##     Min.   1st Qu.    Median      Mean   3rd Qu.    Max.
```

```
## -8.334892 -1.185344 -0.000007  0.737532  1.385482 13.126290
##
##  R-square for Endogenous Variables
##   Height   Weight Rebounds   Blocks   Steals  Assists    FTA   FGA
##   0.7539   0.7674   0.4088   0.3980   0.3794   0.4315   0.6675   1.0510
##    FG3.  FG3A
##   0.1820   1.0806
##
##  Parameter Estimates
##          Estimate    Std Error  z value      Pr(>|z|)
## lambda1  0.86824822 0.04070189 21.3318916 5.743769e-101
## lambda2  0.87600121 0.04054049 21.6080560 1.508680e-103
## lambda3  0.63937721 0.04519203 14.1480071  1.921251e-45
## lambda4  0.63090729 0.04535167 13.9114455  5.397744e-44
## lambda5  0.61596239 0.04282690 14.3826066  6.654044e-47
## lambda6  0.65686548 0.04227173 15.5391211  1.885515e-54
## lambda8  0.81701210 0.03950193 20.6828382  4.944454e-95
## lambda9  1.02518864 0.03391761 30.2258486 1.083805e-200
## lamba10  0.42660465 0.04921132  8.6688319  4.365758e-18
## lamba11  1.03953999 0.05455453 19.0550617  5.964616e-81
## rho1     0.06345286 0.04938371  1.2848946  1.988291e-01
## rho2     0.68876132 0.03799206 18.1290876  1.878503e-73
## rho3     -0.21626192 0.04779942 -4.5243633  6.057761e-06
## theta1   0.24614542 0.03059796  8.0445051  8.659468e-16
## theta2   0.23262221 0.03042679  7.6453090  2.084442e-14
## theta3   0.59119707 0.04415262 13.3898521  6.931866e-41
## theta4   0.60195588 0.04475670 13.4495152  3.098972e-41
## theta5   0.62059121 0.04150223 14.9532005  1.484403e-50
## theta6   0.56852803 0.03809138 14.9253735  2.253700e-50
## theta8   0.33249044 0.02384630 13.9430639  3.467297e-44
## theta9  -0.05101201 0.01553893 -3.2828523  1.027625e-03
## theta10  0.81800904 0.05766656 14.1851532  1.132183e-45
## theta11 -0.08064299 0.09113707 -0.8848539  3.762355e-01

sqrt(mean((nba_sem$C-nba_sem$S)^2))
## [1] 0.2013442
```

# Appendix E: Condensed CFA Model Code and Estimates

```
nba = nba_omit[,c(2,3,8,9,10,15)]
nba_orig = data.frame(nba)
scaled = scale(nba_orig)
NBA_sem <- specifyModel(text = '
Rebounder -> Height, lambda1, NA
Rebounder -> Weight, lambda2, NA
Rebounder -> Blocks, lambda3, NA
Allrounder -> Steals, lambda5, NA
Allrounder -> Assists, lambda6, NA
Allrounder -> FG3A, lamba7, NA
Rebounder <-> Allrounder, rho, NA
Height<-> Height, theta1, NA
Weight <-> Weight, theta2, NA
Blocks <-> Blocks, theta4, NA
Steals <-> Steals, theta5, NA
Assists <-> Assists, theta6, NA
FG3A <-> FG3A, theta7, NA
Allrounder <-> Allrounder, NA, 1
Rebounder <-> Rebounder, NA,1')

nba_sem = sem(NBA_sem, cor(scaled), nrow(scaled))
options(fit.indices = c("GFI", "AGFI", "SRMR"))
summary(nba_sem)
##
##  Model Chisquare =  153.3336   Df =  8 Pr(>Chisq) = 3.951798e-29
##  Goodness-of-fit index =  0.9019199
##  Adjusted goodness-of-fit index =  0.7425396
##  SRMR =  0.1236782
##
##  Normalized Residuals
##     Min.   1st Qu.      Median   Mean   3rd Qu.    Max.
## -1.909306 -0.314743  0.000002  0.913030  1.792467  9.211602
##
##  R-square for Endogenous Variables
##  Height  Weight  Blocks  Steals Assists    FG3A
##  0.8944  0.7215  0.2810  0.5720  0.7615  0.3211
##
##  Parameter Estimates
##           Estimate   Std Error  z value   Pr(>|z|)
## lambda1  0.9457409 0.04255226 22.225401 1.951344e-109 Height <--- Rebounder
## lambda2  0.8493958 0.04368997 19.441439  3.443698e-84 Weight <--- Rebounder
## lambda3  0.5300827 0.04673623 11.342008  8.125818e-30 Blocks <--- Rebounder
## lambda5  0.7563163 0.04804536 15.741714  7.829420e-56 Steals <--- Allrounder
## lambda6  0.8726532 0.04788259 18.224853  3.277543e-74 Assists <--- Allrounder
## lamba7   0.5666380 0.04861373 11.655925  2.140438e-31 FG3A <--- Allrounder
```

```
## rho       -0.3477428 0.04952821 -7.021105  2.201210e-12 Allrounder <--> Rebounder
## theta1   0.1055740 0.04429623  2.383362  1.715530e-02 Height <--> Height
## theta2   0.2785266 0.04007920  6.949405  3.668289e-12 Weight <--> Weight
## theta4   0.7190126 0.05110641 14.068930  5.895472e-45 Blocks <--> Blocks
## theta5   0.4279857 0.04852662  8.819608  1.148609e-18 Steals <--> Steals
## theta6   0.2384766 0.05376187  4.435795  9.173294e-06 Assists <--> Assists
## theta7   0.6789213 0.05171391 13.128408  2.263642e-39 FG3A <--> FG3A

sqrt(mean((nba_sem$C-nba_sem$S)^2))
## [1] 0.1335877
```