Final Report

Mohammed Khan

Brief Introduction

This will be an exploratory data analysis. This dataset is based on advanced statistics taken for NBA player during the 2019-2020 season. This dataset has been sources from Basketball Reference. This dataset contains a unique identifier 'Rk', denoting rank. This variable is the primary key, made in case there happen to be two player with the same first and last name in the dataset.

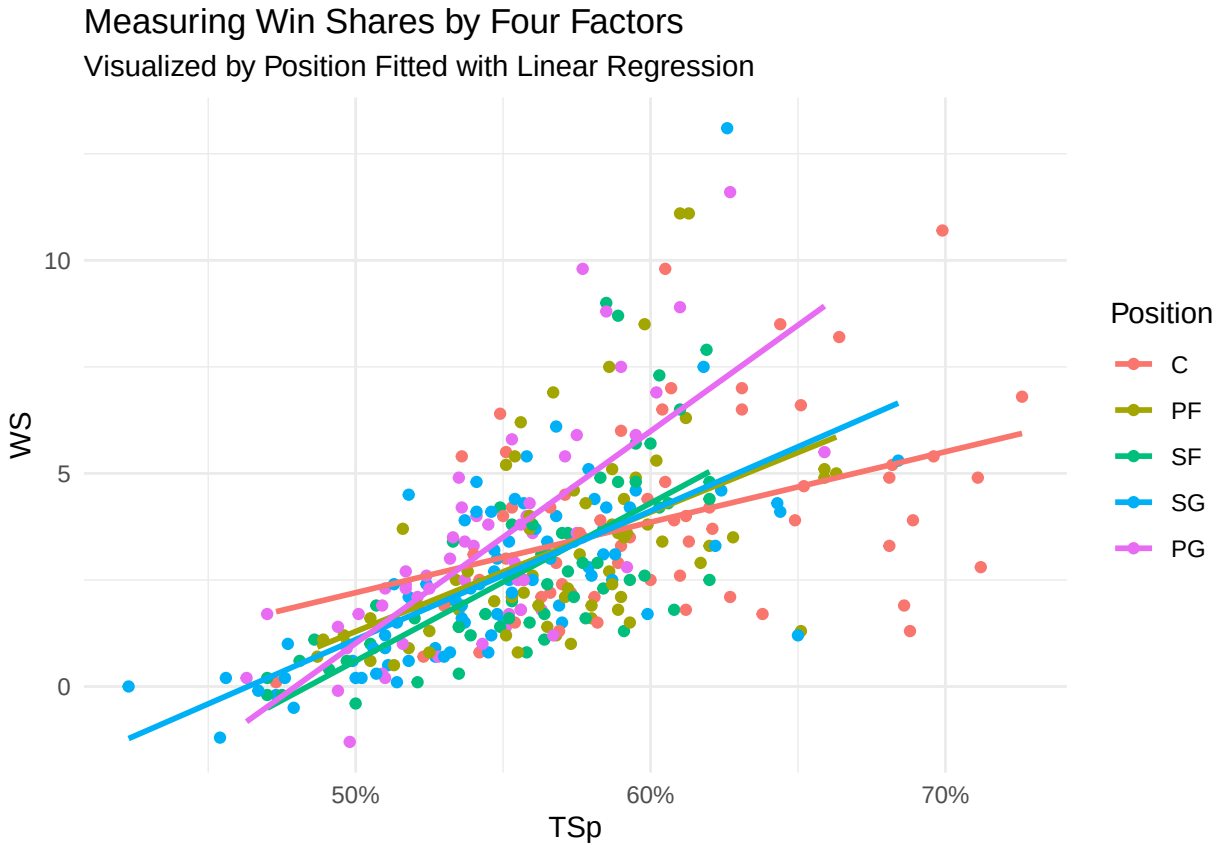
The advanced dimensional stats take into account how efficient a player is while he is on the court. If a player is known for getting a high number of steals per possession played, then he will have a high steal percentage. This is an overly simple method of looking at this dataset, as the calculations behind the numbers are out of the scope of this project. The advanced wholesale stats try to sum up a player's impact in a one neat number. There are concerns with attempting to dissolve a slew of statistics into one impact metric, however as long as the numbers can be confirmed with the eye test, they can be safe to use as a rough outline to rank players. Again, the math behind the metrics are out of the scope of this project.

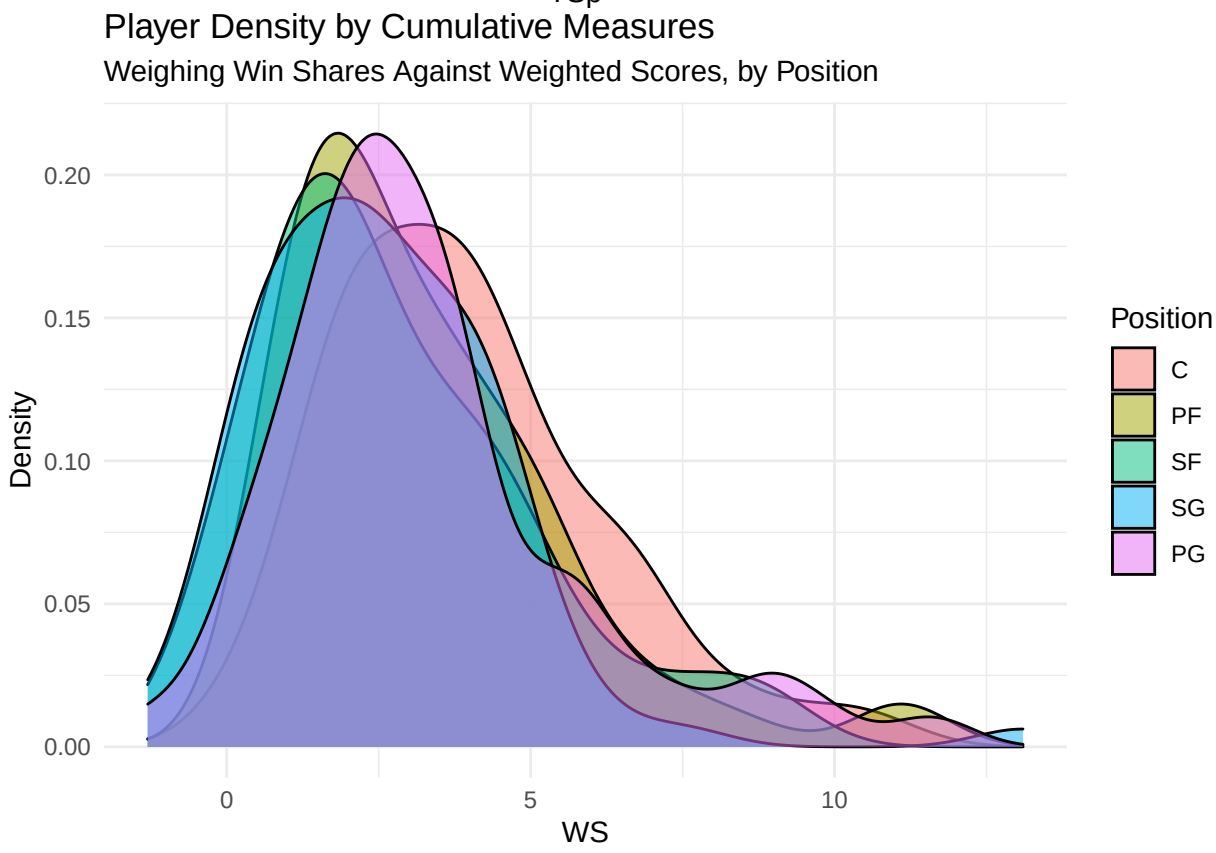
This data contains advanced statistics for various dimensions of the game, along with wholesale statistics to sum up a player's impact in one number. The dimensional stats we will be using are related to the Four Factors, which is a method of weighing dimensional metrics to predict which teams are likely to win. This concept of Four Factors is intended to weigh NBA teams as a whole. The objective of this project is to see if Four Factors can be used to project value in players as reflected in real NBA play.

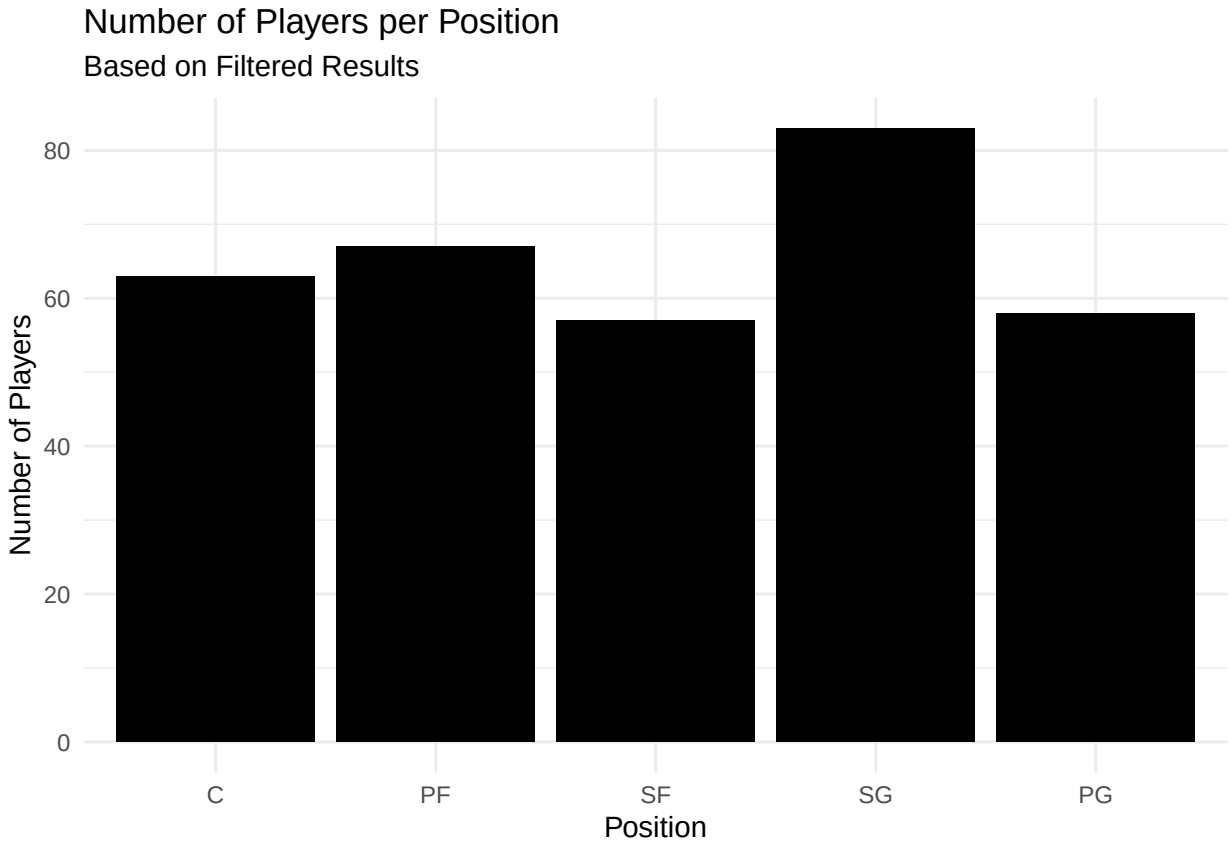
3-5 Plots

```
# import in libraries
library(readr)
library(dplyr)
library(janitor)
library(rsq)
library(ggplot2)
library(scales)
library(plotly)

# read in data
NBA <- read_csv("NBAadv2020.csv")

# dataframe setup
NBAFF <- NBA %>%
    select(Rk, Player, Pos, G, 'USG%', 'TS%', FTr, 'TRB%', 'TOV%', WS) %>%
    filter(G > 40) %>%
    rename('ID' = Rk) %>%
    rename('Position' = Pos) %>%
    rename('TSp' = `TS%`) %>%
    rename('USGp' = `USG%`) %>%
    rename('TRBp' = `TRB%`) %>%
    rename('TOVp' =`TOV%`) %>%
    mutate('TSp' = `TSp` * 100) %>%
    mutate('FTr' = `FTr` * 100) %>%
    mutate('weightedScoreWS' = `TSp` * 0.7084 + `FTr` * 0.0875 + `TRBp` * 0.1494 - `TOVp` * 0.0537) %>%
    mutate('weightedScoreFF' = `TSp` * 0.40 + `FTr` * 0.15 + `TRBp` * 0.20 - `TOVp` * 0.25)

# change double position to single positions, choosing the smaller position
NBAFF$Position[NBAFF$Position == 'SF-SG'] <- 'SG'
NBAFF$Position[NBAFF$Position == 'SF-PF'] <- 'SF'
NBAFF$Position[NBAFF$Position == 'PF-C'] <- 'PF'
NBAFF$Position <- factor(NBAFF$Position, levels = c('C', 'PF', 'SF', 'SG', 'PG'))

# clear NAs
#View(is.na(NBAFF))
NBAFF <- na.omit(NBAFF)
#View(NBAFF)

# r-squared values for win shares attribute in terms of FF attributes
rsq_WS <- rsq(lm(WS ~ TSp, NBAFF), TRUE) +
    rsq(lm(WS ~ FTr, NBAFF), TRUE) +
    rsq(lm(WS ~TRBp, NBAFF), TRUE) +
    rsq(lm(WS ~ TOVp, NBAFF), TRUE)
```

```
## `geom_smooth()` using formula 'y ~ x'
```


Measuring Win Shares by Four Factors
Visualized by Position Fitted with Linear Regression
WS
10
5
0
50%
60%
70%
TSp
Position
C
PF
SF
SG
PG


Player Density by Cumulative Measures
Weighing Win Shares Against Weighted Scores, by Position
Density
0.20
0.15
0.10
0.05
0.00
0
5
10
WS
Position
C
PF
SF
SG
PG



Link to Dashboard

shinyapps.io Link

The above is a clickable link but in case it does not work:

https://mohammed-khan-17.shinyapps.io/Final_Project/

Executive Summary

Overview

The reason advanced stats are considered for this project over traditional stats is to attempt to place players on an even playing field. The only filtering done on the original dataset is to establish a minimum number of games played to take away outliers. These outliers make a notable impact through the advanced stats, but their lack of opportunity to show this impact on the court means they do not affect winning games to the extent the advanced stats might suggest. If a player has been on the court for 41 games, which is 50% of an entire season's worth of games, then he will qualify for this project. This still gives us a robust 328 observations from the original 652. These 328 observations are players who impact winning games at the highest level for this given season.

The Four Factors (weight in parenthesis) of basketball are shooting (40%), turnovers (25%), rebounding (20%) and free throws (15%). For the purpose of this project, these factors will be represented by true shooting percentage (TSp), turnover percentage (TOVp), total rebound percentage (TRBp), and free throw rate (FTr), respectively. Based on these four metrics, we will compare the R-squared value of each against Win Share,

the chosen wholesale metric for this project. Other wholesale metrics include VORP, (value over replacement player) and BPM (box plus/minus). This project will be looking at which players contribute most to winning games through the lens of the Four Factors.

The R-squared value of each dimensional metric will be used to find a weighing proportion based on Win Share. This will be labeled as the wholesale metric 'weighedScoreWS'. There will be another metric named 'weighedScoreFF', which will utilize the percentages above in the original Four Factors concept. We will find which positions provide the most value through these Four Factors through visualizations.

Methodology

There are three types of plots utilized for this project. The first is a scatterplot, fit with linear regression. This visualization is grouped by player position, to provide clarity in how each of the Four Factor metrics affect position when measured by Win Share. The x-axis is one of the Four Factor measures, displaying a relationship with Win Shares and how each position fares. The purpose of this visualization is to provide basic familiarity with the data.

The second visualization is a density plot. The purpose of this plot is to illustrate, by position, where players fall on the spectrum of the wholesale metrics we are measuring. Typically, we would expect peaks of density to be near the average figure for each position, with the troughs of the plot toward the right portion of the visualization denoting the impact figures of the most elite players by a given metric.

The final visualization is a simple histogram that can be filtered by the three wholesale metrics to find the number of players, by position, that qualify under the filtered results. This visualization is best interacted with as part of the dashboard.

Results

The results of the analysis heavily favor players who play the Center and Power Forward positions. This result is true when analysis is done through 'weightedScoreWS' and 'weightedScoreFF'. When analysis is done through the original Win Share metric, there is more of an equal distribution of impact between all five positions. Centers and Power Forwards dominate the metrics made for this project because the Four Factors are dimensional metrics that reflect positively on the taller players of the game.

Taller players are usually the ones closer to the basket, which gives them more quantity of easier scoring chances. Taller players also rebound better than smaller ones for the same reason. Taller players typically do not handle the basketball for the duration of the game, which give them a boost in keeping turnovers low. Taller players tend to be worse free throw shooters than smaller player, however the dimensional metric that was measured is free throw rate, which means actually converting free throws makes no bearing on this metric. Therefore, the players that play the other three positions who grade highly in 'weightedScoreWS' and 'weightedScoreFF' are even more valuable based purely on positional scarcity.

Recommendations

In the end, the results returned by 'weightedScoreWS' and 'weightedScoreFF' are not scalable to the practical game of basketball. Having a roster of the top 5 players in these metrics would mean having five Centers, who likely lack in other basketball skills, such as passing, steals and long-distance shooting. Four Factors is a viable method to measure competency by team, but it falls shorts in assigning value to players with the same metrics. This analysis can be enriched by trying to compare the dimensional metrics to other wholesale variables such as VORP and BPM. The R-squared value of the Four Factors accounted for 65.86% of the variance in Win Shares. While this is not a terrible figure, clearly there is room for improvement. Ideally, an adequate model will value each of the positions similarly, but being limited to only four metrics is limiting.

Appendix of Code (if not already above)

Server

```
# Define server logic required to draw a histogram
shinyServer(function(input, output) {

    output$Scatter <- renderPlot({

        NBAFF <- NBAFF %>%
            filter(USGp >= input$USGp) %>%
            filter(G >= input$G)

        Scatter <- ggplot(NBAFF, aes_string(x = input$toggle,  y = "WS", color = "Position"))
        Scatter +
            labs(title = 'Measuring Win Shares by Four Factors',
                 subtitle = 'Visualized by Position Fitted with Linear Regression') +
            geom_point() +
            geom_smooth(method = 'lm', se = FALSE) +
            scale_x_continuous(labels = percent_format(scale = 1)) +
            theme_minimal()

    })

    output$Density <- renderPlot({

        NBAFF <- NBAFF %>%
            filter(USGp >= input$USGp) %>%
            filter(G >= input$G)

        Density <- ggplot(NBAFF, aes_string(input$toggle2))
        Density +
            labs(title = 'Player Density by Cumulative Measures',
                 subtitle = 'Weighing Win Shares Against Weighted Scores, by Position',
                 y = 'Density') +
            geom_density(aes(fill = factor(Position)), alpha = 0.5) +
            scale_fill_discrete(name = 'Position') +
            theme_minimal()

    })

    output$Hist <- renderPlot({

        NBAFF <- NBAFF %>%
            filter(USGp >= input$USGp) %>%
            filter(G >= input$G) %>%
            filter(WS >= input$WS) %>%
            filter(weightedScoreWS >= input$wsWS) %>%
            filter(weightedScoreFF >= input$wsFF)

        Hist <- ggplot(NBAFF, aes(Position))
        Hist +
            labs(title = 'Number of Players per Position',
```

```
                subtitle = 'Based on Filtered Results',
                y = 'Number of Players') +
            geom_bar(fill = 'black') +
            theme_minimal()

    })

    output$NBAFF <- renderDataTable({

        if (input$USGp != 'All') {
            NBAFF <- NBAFF[NBAFF$USGp >= input$USGp,]
        }
        if (input$G != 'All') {
            NBAFF <- NBAFF[NBAFF$G >= input$G,]
        }
        if (input$USGp != 'All') {
            NBAFF <- NBAFF[NBAFF$WS >= input$WS,]
        }
        if (input$G != 'All') {
            NBAFF <- NBAFF[NBAFF$weightedScoreWS >= input$wsWS,]
        }
        if (input$USGp != 'All') {
            NBAFF <- NBAFF[NBAFF$weightedScoreFF >= input$wsFF,]
        }
        NBAFF

    })

})
```

UI

```
library(shiny)
library(readr)
library(janitor)
library(tidyverse)
library(ggplot2)
library(scales)

NBA <- read_csv("NBAadv2020.csv")

shinyUI(fluidPage(

    # Application title
    titlePanel('NBA Four Factors'),

    # Sidebar with a slider input for number of bins
    sidebarLayout(
        sidebarPanel(
            sliderInput('USGp',
                        'Usage Percentage',
                        min = 8.0,
```

```
                        max = 37.5,
                        value = 20.0),

            sliderInput('G',
                        'Games Played',
                        min = 41,
                        max = 74,
                        value = 41)
        ),
        # Show a plot of the generated distribution
        mainPanel(
            tabsetPanel(type = 'tabs',
                        tabPanel('Measuring Win Shares by Four Factors',
                                 plotOutput('Scatter'),
                                 selectInput('toggle', 'Four Factors:',
                                             c('TSp', 'FTr', 'TRBp', 'TOVp'))
                                 ),
                        tabPanel('Player Density by Cumulative Measures',
                                 plotOutput('Density'),
                                 selectInput('toggle2', 'Cumulative Scores:',
                                             c('WS', 'weightedScoreWS', 'weightedScoreFF'))
                                 ),
                        tabPanel('Number of Players per Position',
                                 plotOutput('Hist'),
                                 sliderInput('WS',
                                             'Win Shares',
                                             min = -1.3,
                                             max = 13.1,
                                             value = -1.3),
                                 sliderInput('wsWS',
                                             'Weighted Win Shares',
                                             min = 31,
                                             max = 59,
                                             value = 31),
                                 sliderInput('wsFF',
                                             'Weighted Four Factors',
                                             min = 16,
                                             max = 40,
                                             value = 16)
                                ),
                        tabPanel('NBA Four Factors Data',
                                 dataTableOutput('NBAFF'))
                    )
                )
    )))
```