Midrange

Kevin Chen

December 25, 2018

Webscraping Data

```
library(rvest)
```

```
## Loading required package: xml2
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(stringr)


datalist = list()

#webscraping for years 2007:2019
for (i in 2007:2019){

#widgets.sports-reference.com/wg.fcgi?css=1&site=bbr&url=%2Fleagues%2FNBA_2019.html&div=div_team_shooti

main <- "https://widgets.sports-reference.com/wg.fcgi?css=1&site=bbr&url=%2Fleagues%2FNBA_"
tail <- ".html&div=div_team_shooting"

url <- paste0(main, i, tail) #combines correct url according to year


first <- url %>%
read_html() %>%
html_table()

first <- first[[1]]
first[2,] <- paste(first[1,], first[2,], sep=" ")

colnames(first) <- first[2,]

first <- first[-c(1:2),]
first$i <- i
datalist[[i]] <- first
```

```
#names(first) <- as.character(unlist(first[1,]))
#first <- first[-1,]

}
teams_shooting <- bind_rows(datalist) # this is for fga% etc

#scrape offensive rating
datalist2 = list()
for(i in 2007:2019){
  main <- "https://widgets.sports-reference.com/wg.fcgi?css=1&site=bbr&url=%2Fleagues%2FNBA_"
  tail2 <- ".html&div=div_misc_stats"
  url <- paste0(main, i, tail2)

  first <- url %>%
    read_html() %>%
    html_table()

  first <- first[[1]]
  colnames(first) <- first[1,]
  first <- first[-1,]
  first$i <- i
  datalist2[[i]] <- first
}

#Combining data
teams_stats <- bind_rows(datalist2)
teams_stats$Team <- str_remove_all(teams_stats$Team, "\\*") #remove * from teams
names(teams_shooting) <- trimws(names(teams_shooting))
```

Rename teams

```
#bobcats
teams_stats[teams_stats[,"Team"] == "Charlotte Bobcats", "Team"] = "Charlotte Hornets"
teams_shooting[teams_shooting[,"Team"] == "Charlotte Bobcats", "Team"] = "Charlotte Hornets"

#seattle
teams_stats[teams_stats[,"Team"] == "Seattle Supersonics", "Team"] = "Oklahoma City Thunder"
teams_shooting[teams_shooting[,"Team"] == "Seattle Supersonics", "Team"] = "Oklahoma City Thunder"

#new orelans
teams_stats[teams_stats[,"Team"] == "New Orleans Hornets", "Team"] = "New Orleans Pelicans"
teams_shooting[teams_shooting[,"Team"] == "New Orleans Hornets", "Team"] = "New Orleans Pelicans"
```

Select relevant columns for mid range data

```
#selecting relevant columns
teams_midrange<-teams_shooting[,c(1,2,5,9,10,11,12,15,16, 17, 18)]
```

```
#create year column
teams_midrange$Year<-teams_shooting%>%select(., i) %>% unlist()

#make columns numeric
teams_midrange[,3:ncol(teams_midrange)]<-sapply(teams_midrange[,3:ncol(teams_midrange)],as.numeric)

#create %FGA by midrange column
teams_midrange$`% of FGA by Midrange` <- rowSums(teams_midrange[,c(5,6)])
```

Arranging team stats each season in alphabetical order

```
end = 30
start = 1
c=0
#1-30 for 2007, then again for 2008 onward
while(c<13){
  teams_stats[start:end,]<-teams_stats[start:end,]%>% arrange(.,Team)
  start = end+2
  end = start+29
  c=c+1
}
```

Add Ortg to Midrange data

```
teams_midrange$ORtg <- as.numeric(teams_stats$ORtg) #adding ORtg from team_stats to team_shooting
teams_midrange<-teams_midrange[,c(12,1:6,13,7:11,14)] #rearrange columns
teams_midrange$Team <- str_remove_all(teams_midrange$Team, "\\*")
```

sort according to ortg in desc order

```
end = 30
start = 1
c=0
while(c<13){
  teams_midrange[start:end,]<-teams_midrange[start:end,]%>% arrange(desc(ORtg))
  start = end+2
  end = start+29
  c=c+1
}
```

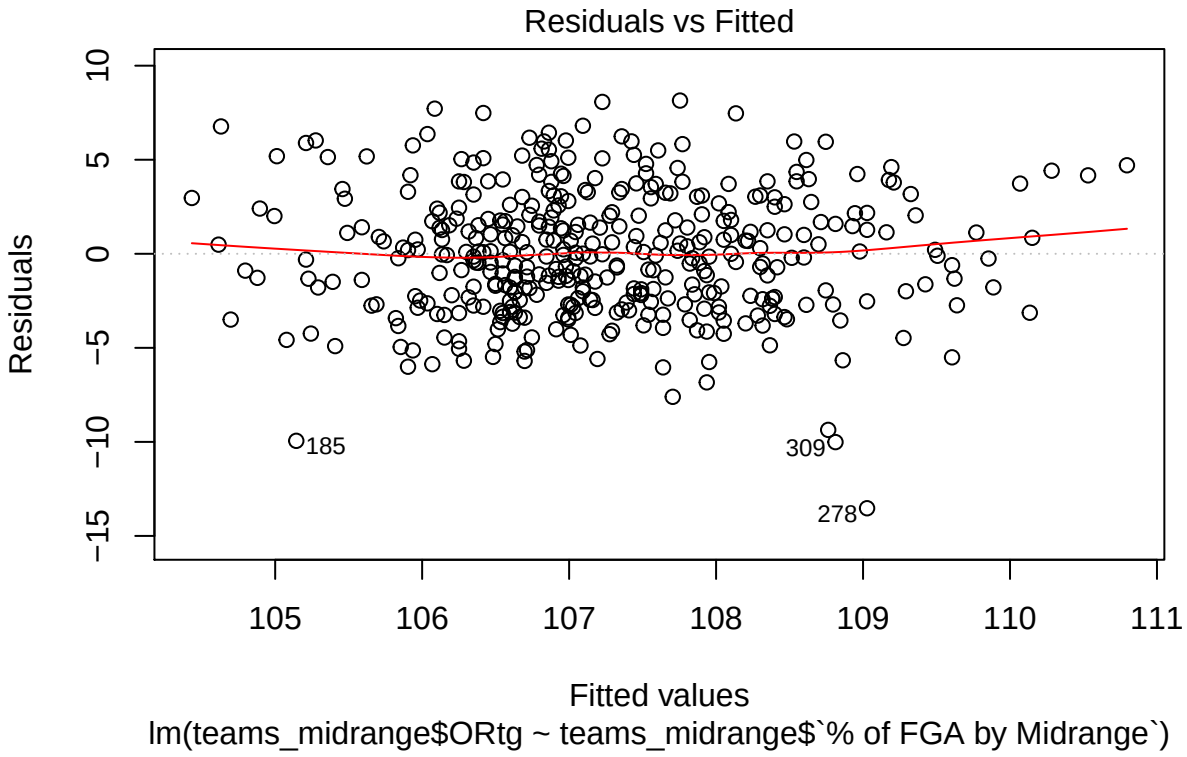
Lm predicting Ortg from FGA Midrange. Residuals looks good

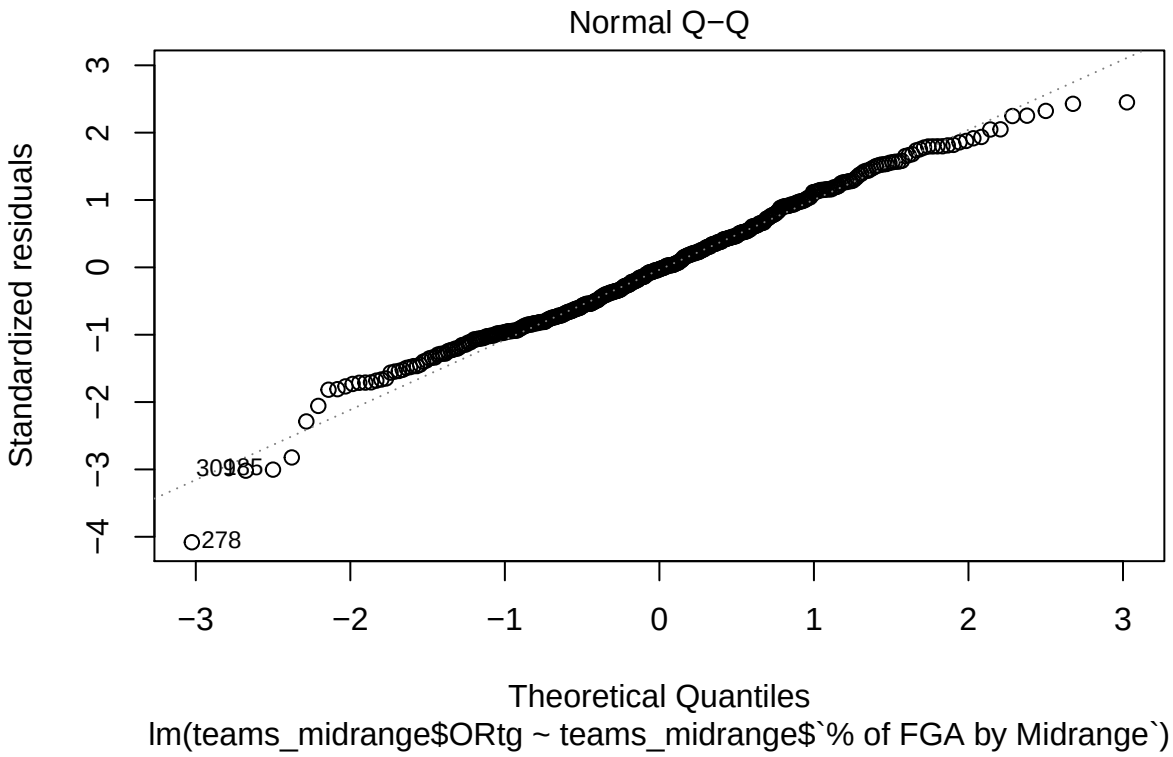
```
offrtg<-lm(teams_midrange$ORtg~teams_midrange$`% of FGA by Midrange`)
summary(offrtg)
```

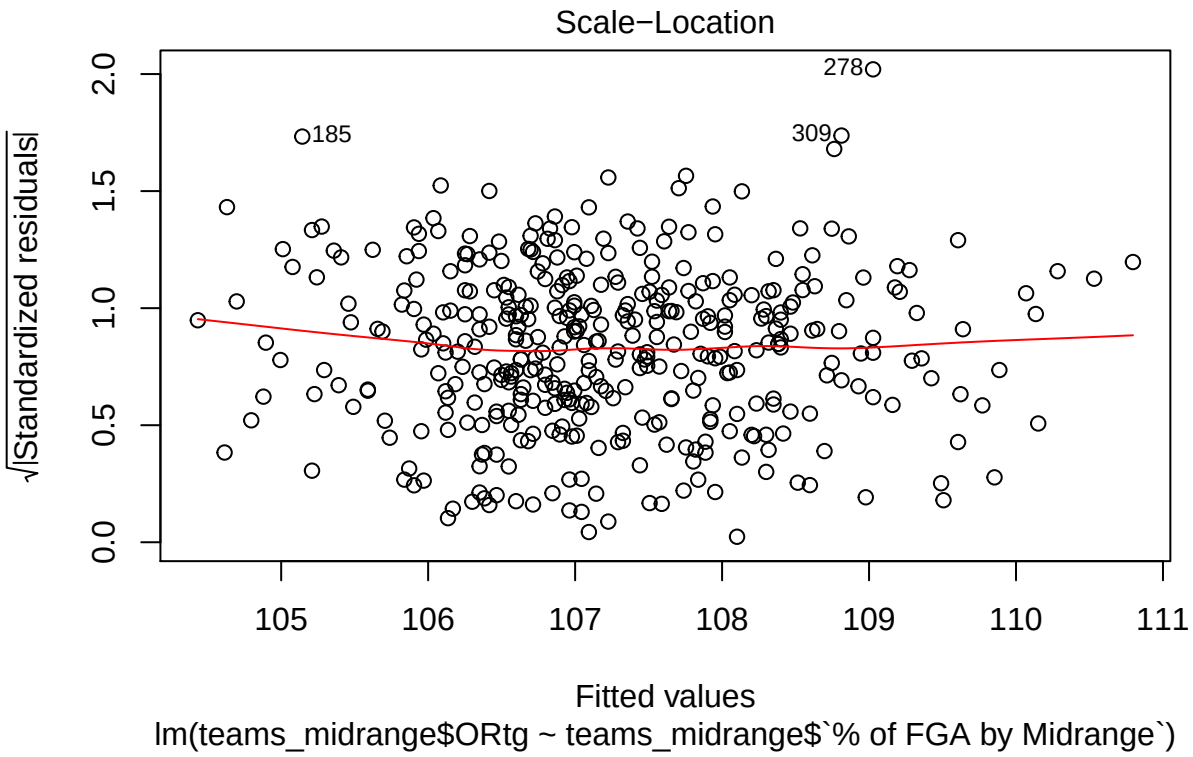
```
##
## Call:
## lm(formula = teams_midrange$ORtg ~ teams_midrange$`% of FGA by Midrange`)
##
## Residuals:
```

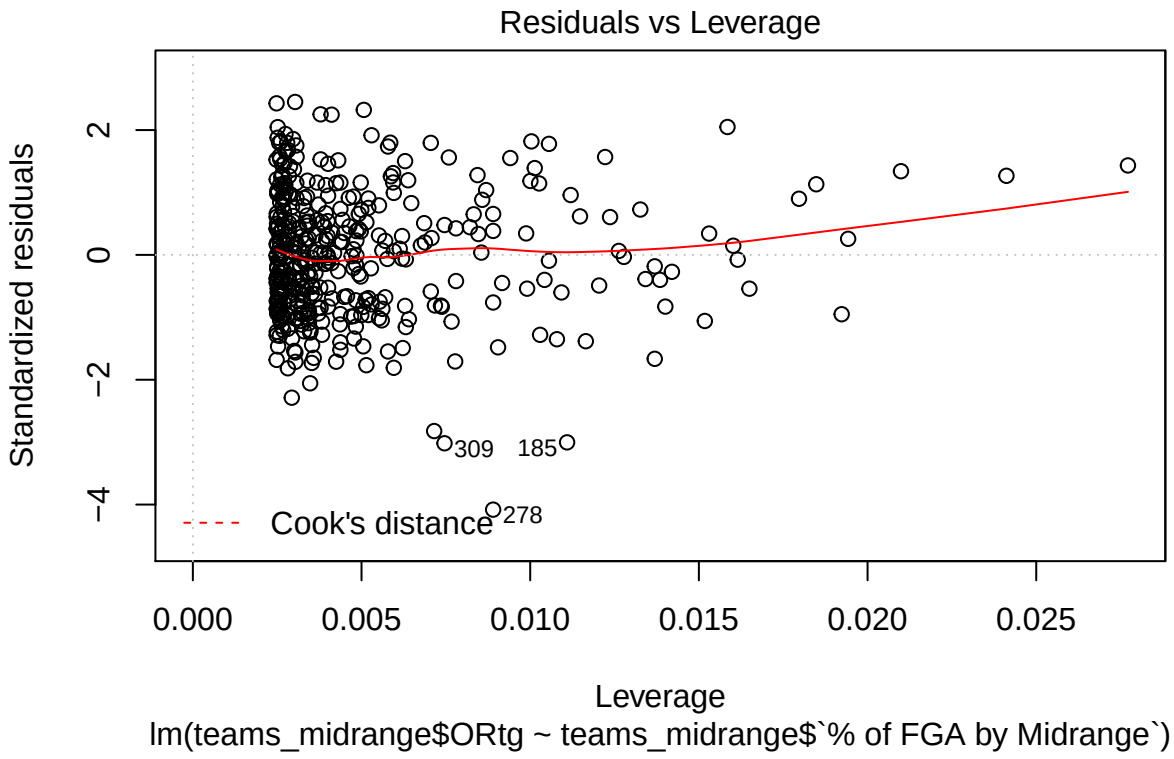
```
##      Min       1Q   Median       3Q      Max
## -13.5273  -2.4515  -0.0895   2.2113   8.1452
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        112.0352     0.7317 153.111  < 2e-16
## teams_midrange$`% of FGA by Midrange` -16.5268     2.4500  -6.746 5.34e-11
##
## (Intercept)                        ***
## teams_midrange$`% of FGA by Midrange` ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.329 on 401 degrees of freedom
## Multiple R-squared:  0.1019, Adjusted R-squared:  0.09967
## F-statistic:  45.5 on 1 and 401 DF,  p-value: 5.344e-11
```

```
plot(offrtg)
```



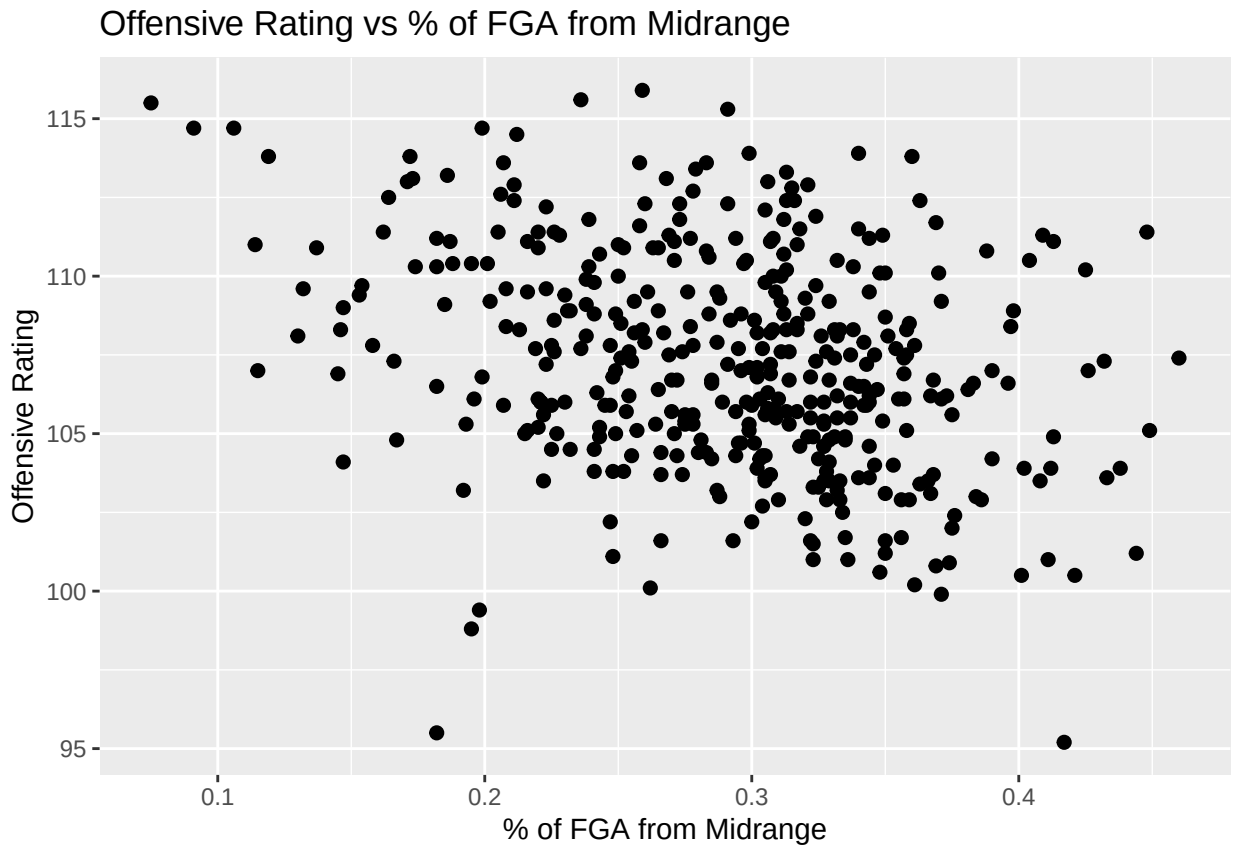

Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(teams_midrange$ORtg ~ teams_midrange$`% of FGA by Midrange`)
278


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
185
278
309
105
106
107
108
109
110
111
Fitted values
lm(teams_midrange$ORtg ~ teams_midrange$`% of FGA by Midrange`)

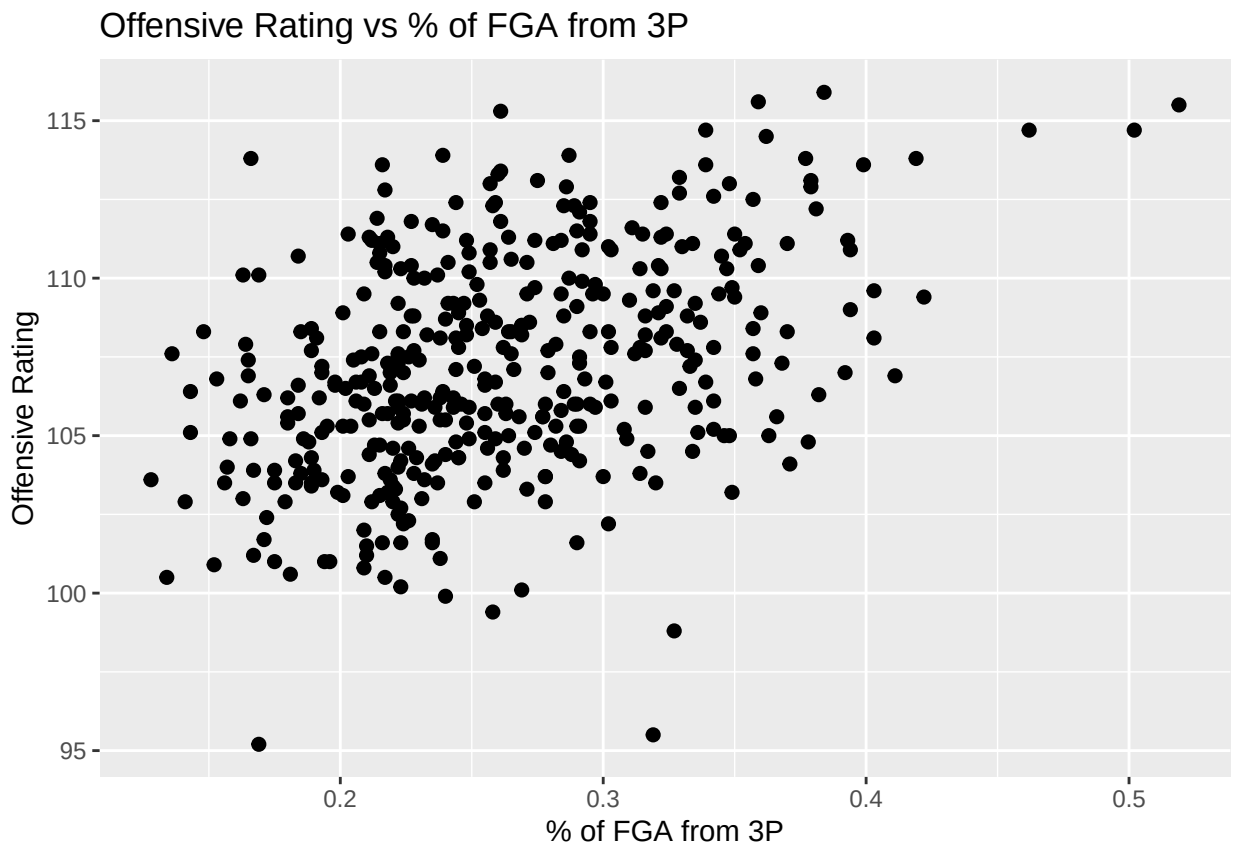


Plot offensive rating and % of FGA from midrange

```
library(ggplot2)
ggplot(teams_midrange, aes(x = `% of FGA by Midrange` , y = ORtg)) + geom_point(size = 2) + labs(title =
```




```
ggplot(teams_midrange, aes(x = `% of FGA by Distance 3P` , y = ORtg)) + geom_point(size = 2) + labs(tit
```



% FGA from 3 by year

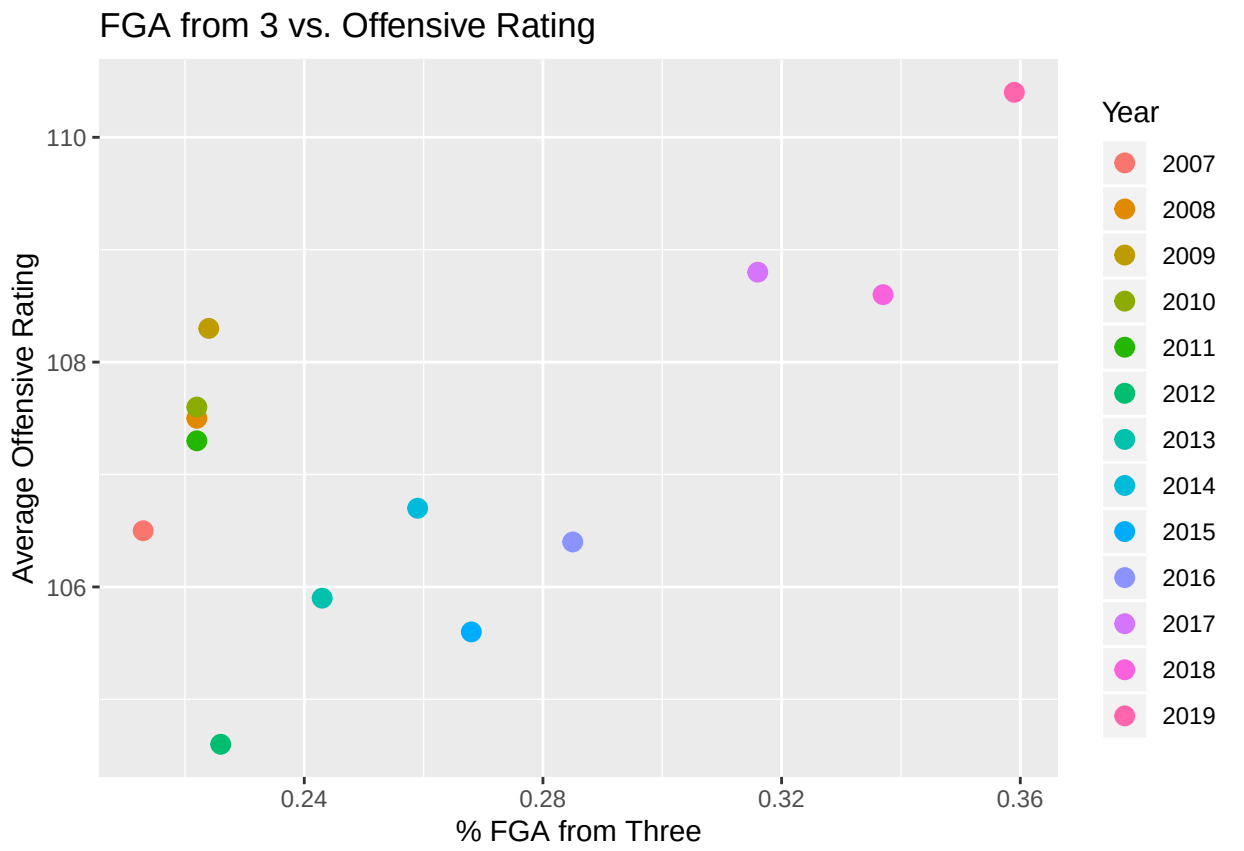
```
League_Avg3<-teams_midrange[seq(31,403,31), 9] #taking the league avg every year of FGA from 3P
Year <- 2007:2019
Three_Avg <- data.frame(Year = Year, Average = League_Avg3)
```

Avg Offensive Rating by year

```
Avg_OffRtg <- data.frame(Year = Year, Avg_Off_Rtg = teams_midrange[seq(31,403,31),14])
Avg_13 <- inner_join(Three_Avg, Avg_OffRtg, by = "Year")
Avg_13$Year <- factor(Avg_13$Year)
```

Plot of Avg Offensive Rating vs Avg. % FGA from 3

```
ggplot(Avg_13, aes(Average ,Avg_Off_Rtg,colour = Year)) + geom_point(size =3) + labs(x = "% FGA from Th
```

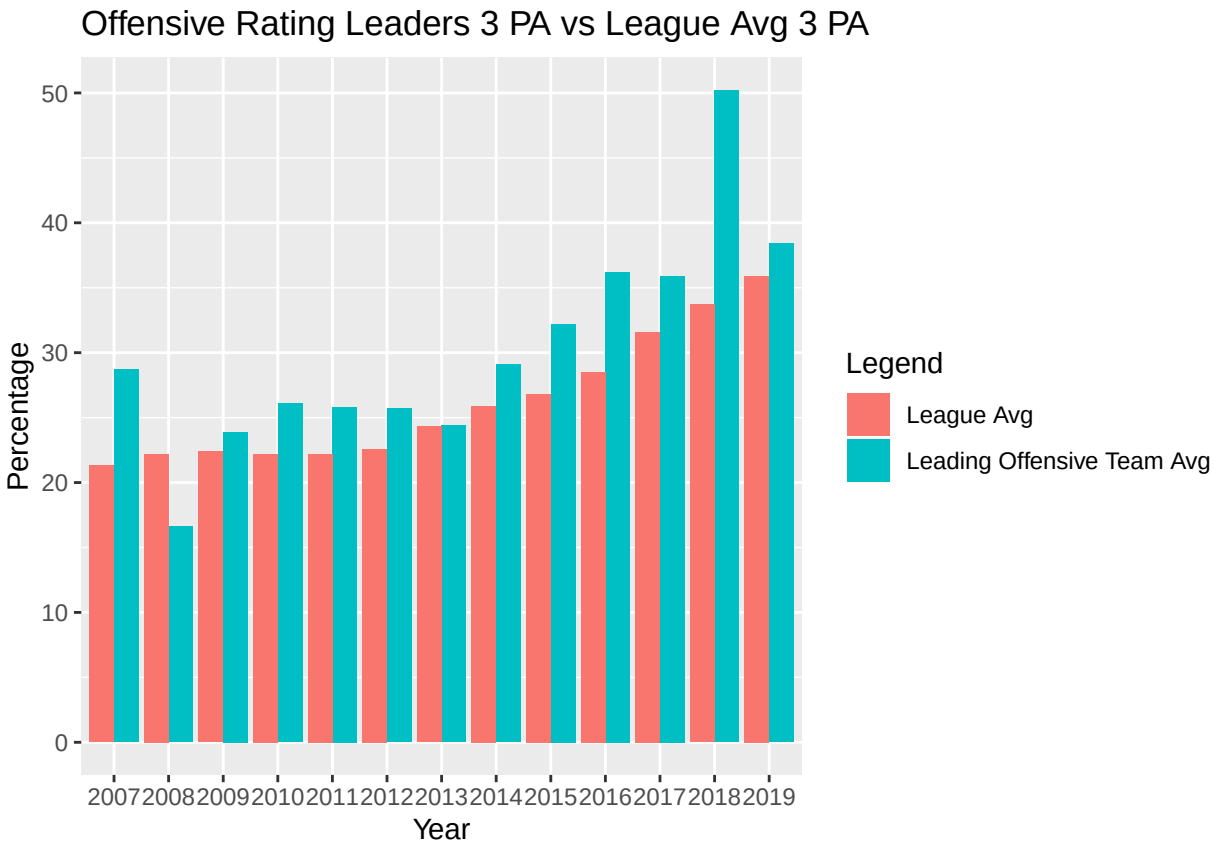


Off rating leaders 2007-19

```
Off_Rank<- data.frame(teams_midrange[seq(1,403,31),])
Off_Rank$Year<-unlist(Off_Rank$Year)
names(Off_Rank) <- names(teams_midrange)
```

% FGA 3p compared to league average for ortg leader

```
Off_Rank3<-inner_join(Off_Rank,Three_Avg, by = "Year") ##Difference in Off Rtg leaders 3P FGA to league
Off_Rank3<- Off_Rank3 %>% select(Year, Team, `% of FGA by Distance 3P`, ORtg, Average)
names(Off_Rank3)[5] <- "League 3 PT FGA"
Off_Rank3$Year <- factor(Off_Rank3$Year)

stacked_bar <- rbind(as.matrix(Off_Rank3[,c("Year","League 3 PT FGA")]), as.matrix(Off_Rank3[,c("Year",
names(stacked_bar) <- c("Year", "Percentage")
category <- c(rep("League Avg 3 Pt Att", 13), rep("Team Avg 3 Pt Att",13))
stacked_bar$Category <- category
stacked_bar$Year <- factor(stacked_bar$Year)
stacked_bar$Percentage <- stacked_bar$Percentage %>% as.character() %>% as.numeric()
stacked_bar$Percentage <- stacked_bar$Percentage * 100
stacked_bar <- stacked_bar %>% arrange(Year)
ggplot(data = stacked_bar, aes(Year,Percentage, fill = Category)) + geom_bar(position = "dodge", stat =
```

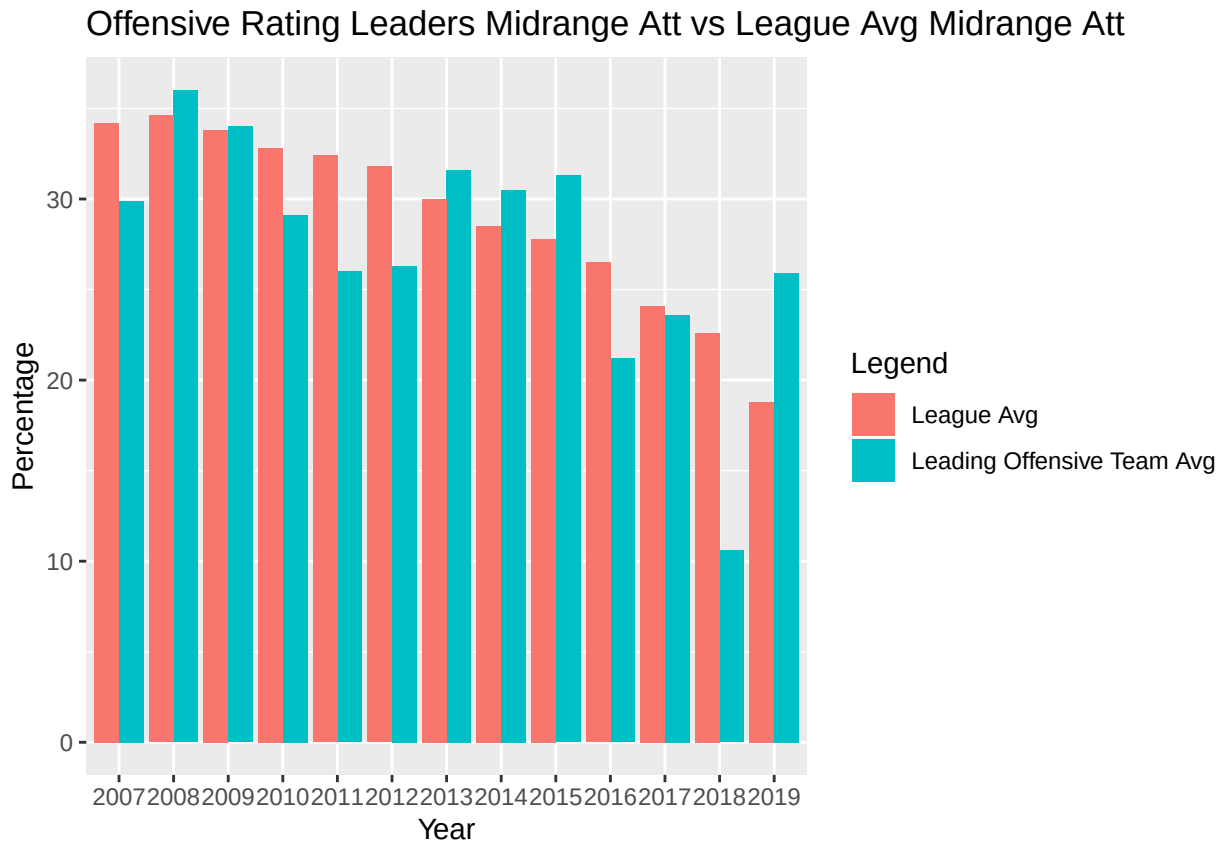


% field goal attempts from mid range by year

```
League_Avg_Mid<- teams_midrange[seq(31,403,31), 8]
```

Compare ortg leaders to FGA from midrange avg for league

```
Off_LeadMid<-cbind(Off_Rank,League_Avg_Mid) %>% select(Year, Team, `% of FGA by Midrange`, League_Avg_Mi
side_bar <- rbind(as.matrix(Off_LeadMid[,c("Year","League_Avg_Mid")]), as.matrix(Off_Rank[,c("Year","% o
names(side_bar) <- c("Year", "Percentage")
category <- c(rep("League Avg Midrange Att", 13), rep("Team Avg Midrange Att",13))
side_bar$Category <- category
side_bar$Year <- factor(side_bar$Year)
side_bar$Percentage <- side_bar$Percentage %>% as.character() %>% as.numeric()
side_bar$Percentage <- side_bar$Percentage * 100
side_bar <- side_bar %>% arrange(Year)
ggplot(data = side_bar, aes(Year,Percentage, fill = Category)) + geom_bar(position = "dodge", stat = "id
```



How many times did team with best offense finsih inside top 10 midrange FGA

```
#rank teams by FGA % frm midrange
top10_mid <- teams_midrange %>% filter(Team != "League Average") %>% group_by(Year) %>% mutate(Mid_Rank
#convert year to numeric from factor
top10_mid$Year <- top10_mid$Year %>% as.character() %>% as.numeric()
#convert offensive rating leaders' midrange fga % "year" to numeric
Off_LeadMid$Year <- Off_LeadMid$Year %>% as.character() %>% as.numeric()

n = 2007
c=0
while(n <= 2019){
  temp <- top10_mid %>% filter(Year == n, Mid_Ranking <=10) %>% arrange(Mid_Ranking) %>% select(Year, T
  temp_team <- Off_LeadMid %>% filter(Year == n)
  team <- temp_team$Team
  if(team %in% temp$Team) {
    c = c+1
    cat(c, " ", n, " ", team)
  }
    n = n+1
}
```

```
## 1   2008   Utah Jazz2   2015   Los Angeles Clippers3   2019   Golden State Warriors
```

How many times did teams with highest FGA from 3 finished inside the top

```
#rank teams be FGA % from 3
top10_three <- teams_midrange %>% filter(Team != "League Average") %>% group_by(Year) %>% mutate(Three_
#convert year to numeric
top10_three$Year <- top10_three$Year %>% as.character() %>% as.numeric()
#offensive leaders by year convert year to numeric
Off_Rank$Year <- Off_Rank$Year %>% as.character() %>% as.numeric()

c = 0
n = 2007
new_df <- data.frame(matrix(ncol = 2, nrow = 0)) #stores data frame
while(n <= 2019){
  #create temporary vriable that ranks 3 pt leaders and arranges it in descending order
  temporary <- top10_three %>% filter(Year == n, Three_Pt_Ranking <= 10) %>% arrange(Three_Pt_Ranking) 
  #create temporary df of offensrive rating leader for the year n
  temp_team <- Off_Rank %>% filter(Year == n)
  #choose only the team for the year n
  team <- temp_team$Team
  #if statement to check if team is in that year's top 10
  if(team %in% temporary$Team){
    c = c+1
    new_df[c,1] <- n
    new_df[c,2] <- team
  }
 n = n+1
}
new_df
```

```
##      X1                   X2
## 1  2007          Phoenix Suns
## 2  2010          Phoenix Suns
## 3  2011        Denver Nuggets
## 4  2012     San Antonio Spurs
## 5  2014  Los Angeles Clippers
## 6  2015  Los Angeles Clippers
## 7  2016 Golden State Warriors
## 8  2017 Golden State Warriors
## 9  2018       Houston Rockets
## 10 2019 Golden State Warriors
```

Among teams in top 10 midrange FGA every year, how many of them ranked in the top 10 in offense that year?

- out of all the teams in that span, how many times did team with highest % of midrange FGA ranked in the top 10 in offense?

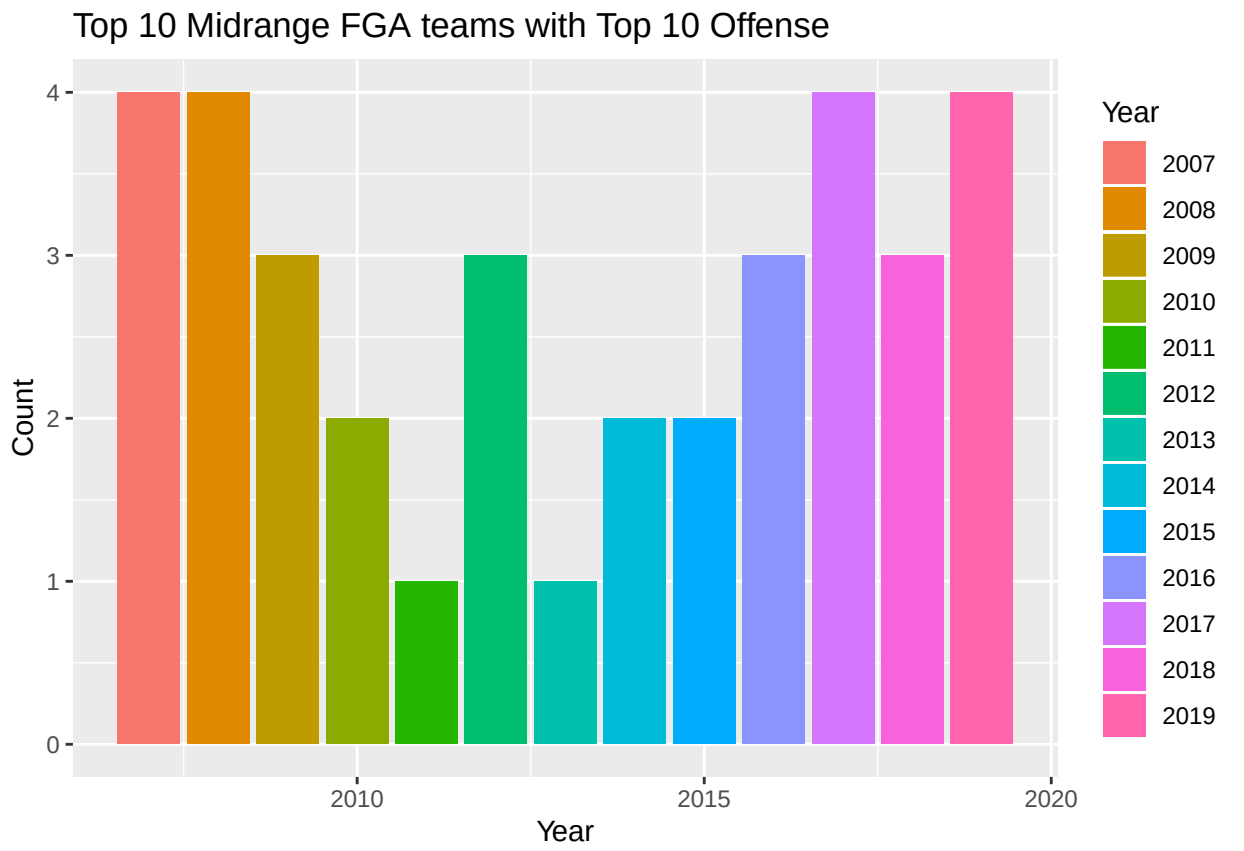
```
#rank top10 offenses
top10_off <- teams_midrange %>% filter(Team != "League Average") %>% group_by(Year) %>% mutate(Offense_
top10_off <- top10_off %>% filter(Offense_Ranking <= 10) %>% arrange(Year, Offense_Ranking)
#convert yer to numeric
top10_off$Year <- top10_off$Year %>% as.character() %>% as.numeric()
```

```
#midrange att leaders
midrange_leaders <- top10_mid %>% filter(Mid_Ranking == 1)

c = 0
n = 2007
#storesall occurrences where midrange att leaders finished in top 10 in offense
top10mid_df <- data.frame(matrix(ncol = 2, nrow =0))
while(n <= 2019){
  team_mid <- midrange_leaders %>% filter(Year == n)
  temp_top10 <- top10_off %>% filter(Year == n)
  if(team_mid$Team %in% temp_top10$Team){
    c=c+1
    top10mid_df[c,1] <- n
    top10mid_df[c,2] <- team_mid$Team
  }
  n = n+1
}
top10mid_df
```

```
##     X1                X2
## 1 2008   Detroit Pistons
## 2 2017 San Antonio Spurs
## 3 2019 San Antonio Spurs
```

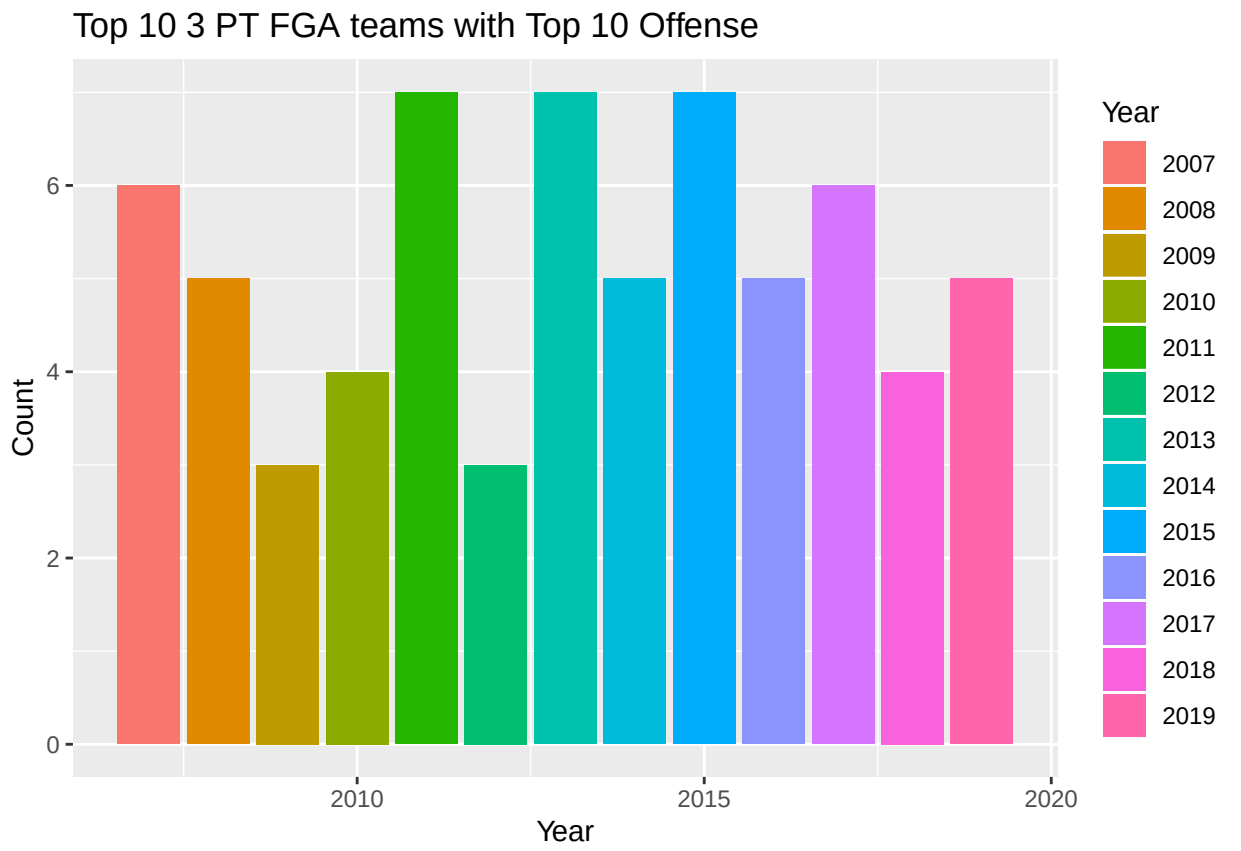
Top 10 midrange FGA% in top 10 offense by year

```
temp_top10_offense <- top10_off %>% select(Year, Team, Offense_Ranking)
temp_top10_midrange <- top10_mid %>% select(Year, Team, Mid_Ranking)
temp_top10_midrange <- temp_top10_midrange %>% filter(Mid_Ranking <= 10) %>% arrange(Year, Mid_Ranking)

n = 2007
common_df <- data.frame(matrix(ncol = 2, nrow = 0))
c = 0
while(n <= 2019){
  x <- temp_top10_offense %>% filter(Year == n)
  y <- temp_top10_midrange %>% filter(Year == n)
  z <- intersect(x$Team,y$Team)
  if(length(z) > 0){
  c = c+ 1
  common_df[c,1] <- n
  common_df[c,2] <- length(z)
  }
  n = n+1
}
ggplot(common_df, aes(x=X1, y=X2, fill = factor(X1))) + geom_bar(stat = "identity") + labs(title = "Top
```



Top 10 3 Pt FGA% in top 10 offense by year

```
temp_top10_three <- top10_three %>% filter(Three_Pt_Ranking<=10) %>% arrange(Year, Three_Pt_Ranking) %>%

n = 2007
common_df <- data.frame(matrix(ncol = 2, nrow = 0))
c = 0
while(n <= 2019){
  x <- temp_top10_offense %>% filter(Year == n)
  y <- temp_top10_three %>% filter(Year == n)
  z <- intersect(x$Team,y$Team)
  if(length(z) > 0){
  c = c+ 1
  common_df[c,1] <- n
  common_df[c,2] <- length(z)
  }
  n = n+1
}
ggplot(common_df, aes(x=X1, y=X2, fill = factor(X1))) + geom_bar(stat = "identity") + labs(title = "Top
```


Top 10 3 PT FGA teams with Top 10 Offense
Count
0
2
4
6
2010
2015
2020
Year
Year
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019