Problem Set 6

Subeom Lee

2019-03-10

Questions

```r
#http://asbcllc.com/nbastatR/index.html

library(nbastatR)
library(future)
library(stringi)
library(tidyverse)
library(lubridate)
library(texreg)
library(broom)
library(knitr)
library(ggpubr)
library(ggrepel)
library(janitor)
library(plotly)
library(reticulate)

plan(multiprocess)

# Run only when needed
# game_logs(seasons = 1947:2019, result_types = c("team", "player"))
# dataGameLogsTeam$Team = substring(dataGameLogsTeam$slugMatchup, 1, 3)

# Run when you updated data
# save(df_nba_player_dict, file='df_nba_player_dict.Rdata')
# save(dataGameLogsTeam, file='dataGameLogsTeam.Rdata')
# save(dataGameLogsPlayer, file='dataGameLogsPlayer.Rdata')

# load('df_nba_player_dict.Rdata')
# load('dataGameLogsTeam.Rdata')
# load('dataGameLogsPlayer.Rdata')

load('BaseEnvironment.Rdata')



# avg <- aggregate(dataGameLogsTeam[, 24:46], list(dataGameLogsTeam$yearSeason, dataGameLogsTeam$Team), mean)
# colnames(avg)[1] <- "Year"
# colnames(avg)[2] <- "Team"
#
#
#
# avgplot <- avg %>%
#             filter(Team %in% c('GSW', 'CHI', 'HOU', 'LAL')) %>%
#             ggplot(aes(x=Year, y=pctFG3Team, colour=Team)) +
#             geom_line()
# avgplot
#
# avg2 <- aggregate(dataGameLogsTeam[, 24:46], list(dataGameLogsTeam$yearSeason), mean)
# colnames(avg2)[1] <- "Year"
# min2 <- aggregate(dataGameLogsTeam[, 24:46], list(dataGameLogsTeam$yearSeason), min)
# colnames(min2)[1] <- "Year"
# max2 <- aggregate(dataGameLogsTeam[, 24:46], list(dataGameLogsTeam$yearSeason), max)
# colnames(max2)[1] <- "Year"
#
```

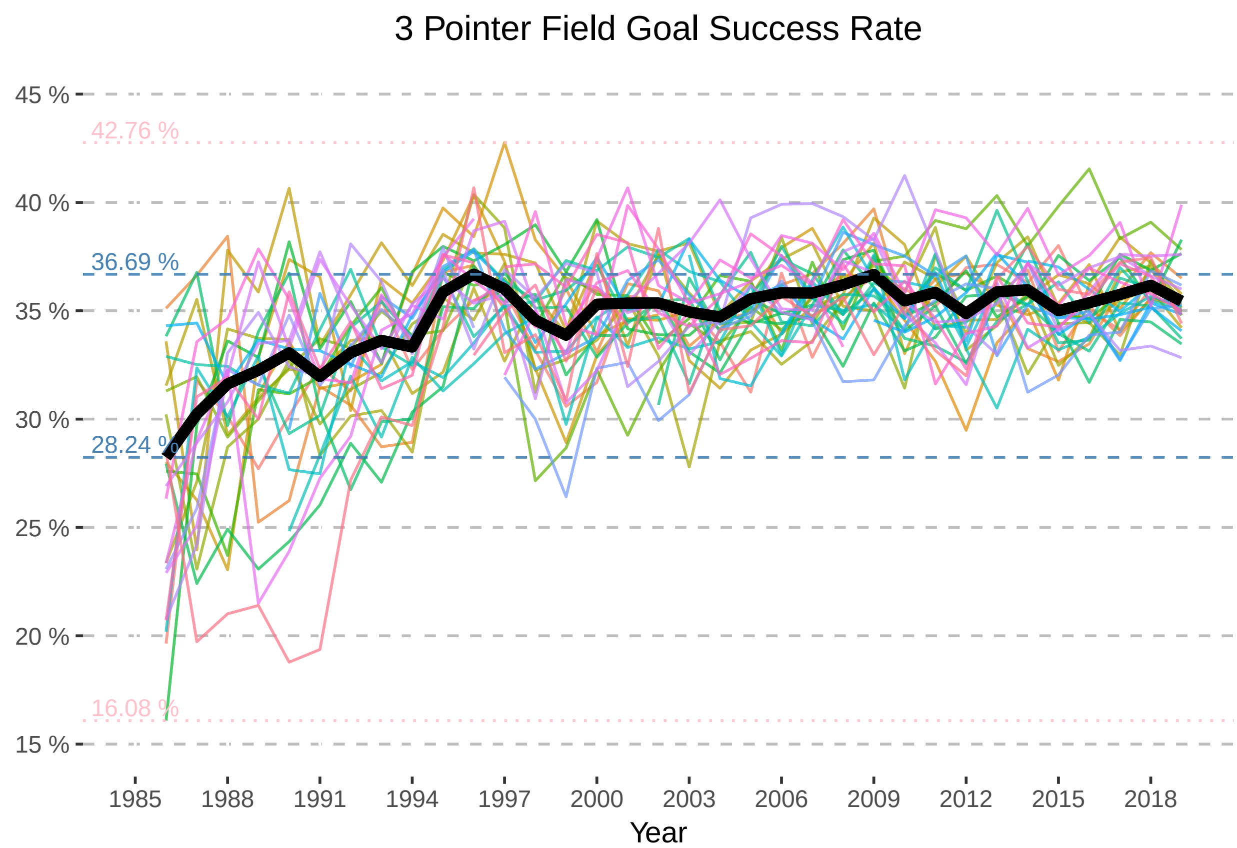
```r
# avgplot2 <- avg2 %>%
#               filter(Year >= 1986) %>%
#               ggplot(aes(x=Year, y=pctFG3Team*100)) +
#               geom_path(colour='violet', size=2)
# avgplot2
#
# avgplot3 <- avg2 %>%
#               ggplot(aes(x=Year, y=pctFG2Team*100)) +
#               geom_path(colour='red', size=2)
# avgplot3
#
# avgminmax <- ggplot() +
#                 geom_line(data=min2, aes(x=Year, y=pctFG3Team*100), size=1, colour='green') +
#                 geom_line(data=max2, aes(x=Year, y=pctFG2Team*100), size=1, colour='red') +
#                 geom_smooth(data=avg2, aes(x=Year, y=pctFG3Team*100), size=2, colour='black')
# avgminmax
#
# avg1986 <- avg %>% filter(Year>=1986)
# avg21986 <- avg2 %>% filter(Year>=1986)
#
# avgcombinedall <- ggplot() +
#                 geom_line(data=avg1986, aes(x=Year, y=pctFG3Team*100, colour=Team), size=0.5, show.legend=FALSE) +
#                 geom_line(data=avg1986, aes(x=Year, y=pctFG2Team*100, colour=Team), size=0.5, show.legend=FALSE) +
#                 geom_line(data=avg21986, aes(x=Year, y=pctFG3Team*100), size=2, colour='black')+
#                 geom_line(data=avg21986, aes(x=Year, y=pctFG2Team*100), size=2, colour='black')
# avgcombinedall
#
# avgfiltered <- avg %>% filter(Team %in% c('GSW', 'CHI', 'HOU', 'LAL'))
#
# avgcombined <- ggplot() +
#                 geom_line(data=avgfiltered, aes(x=Year, y=pctFG3Team*100, colour=Team), size=1, show.legend=FALSE) +
#                 geom_line(data=avgfiltered, aes(x=Year, y=pctFG2Team*100, colour=Team), size=1, show.legend=FALSE) +
#                 geom_smooth(data=avg2, aes(x=Year, y=pctFG3Team*100), size=2, colour='black')
# avgcombined
#
# avgcombined2 <- avgcombined +
#                 geom_smooth(data=avg2, aes(x=Year, y=pctFG2Team*100), size=2, colour='red')
# avgcombined2
#
# avgcombined3 <- avgcombined2 +
#                 geom_smooth(data=avg2, aes(x=Year, y=pctFTTeam*100), size=2, colour='green')
# avgcombined3
#
# avgcombined4 <- avgcombined3 +
#                 geom_smooth(data=avg2, aes(x=Year, y=ptsTeam), size=2, colour='purple')
# avgcombined4

# ggplotly(p=ggplot2::last_plot())

# library(ggplot2)
# library(ggpubr)
# theme_set(theme_pubr())
#
# figure <- ggarrange(avgplot, avgplot2,
#                     labels = c("Each Team", "All Teams"),
#                     ncol = 1, nrow = 2)
# figure

# climate <- read.csv('ps5_data.csv')
# a <- ggplot(climate) +
#       xlab('Year') +
#       ylab('Temperature(°C)') +
#       theme(panel.border=element_rect(colour="black", fill=NA), panel.background=element_rect(fill=NA),
#             panel.grid=element_line(color="grey")) +
#       geom_smooth(aes(Year, Lowess.5.), colour="blue", size=1) +
#       geom_line(aes(Year, No_Smoothing), colour="grey", size=1) +
#       geom_point(aes(Year, No_Smoothing), shape=1, size=3)
```

```
#
```

Team level questions

Q1. It seems that players are getting better at making 3-pointers than 20 years ago (both on average and also top 3-pointer shooters vs. top 3-pointer shooters) Is it true?

```
fg3year <- aggregate(dataGameLogsTeam[, 35:36], list(dataGameLogsTeam$yearSeason), sum)
colnames(fg3year)[1] <- "Year"
fg3year <- fg3year %>% filter (Year >= 1986)
fg3year$pctfg3 <- fg3year$fg3mTeam / fg3year$fg3aTeam * 100

fg3yearteam <- aggregate(dataGameLogsTeam[, 35:36], list(dataGameLogsTeam$yearSeason, dataGameLogsTeam$T
colnames(fg3yearteam)[1] <- "Year"
colnames(fg3yearteam)[2] <- "Team"
fg3yearteam <- fg3yearteam %>% filter (Year >= 1986)
fg3yearteam$pctfg3 <- fg3yearteam$fg3mTeam / fg3yearteam$fg3aTeam * 100

xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(15, 45, by=5)

Q1 <- ggplot() +
  geom_line(data=fg3yearteam, aes(x=Year, y=pctfg3, colour=Team), size=0.5, show.legend=FALSE, alpha=0.7
  geom_line(data=fg3year, aes(x=Year, y=pctfg3), size=2, colour='black') +
  xlab('Year') +
  ylab(NULL) +
  ggtitle('3 Pointer Field Goal Success Rate') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=2
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(15, 45), breaks=yaxisbreaks, labels=paste(yaxisbreaks,"%")) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks) +
  geom_hline(yintercept=min(fg3year$pctfg3), linetype=2, color="steelblue", size=0.5, alpha=0.9) +
  geom_hline(yintercept=max(fg3year$pctfg3), linetype=2, color="steelblue", size=0.5, alpha=0.9) +
  geom_hline(yintercept=min(fg3yearteam$pctfg3), linetype=3, color="pink", size=0.5, alpha=0.9) +
  geom_hline(yintercept=max(fg3yearteam$pctfg3), linetype=3, color="pink", size=0.5, alpha=0.9) +
  annotate("text", x=1985, y=min(fg3year$pctfg3)+0.6, label=paste(toString(round(min(fg3year$pctfg3), di
  annotate("text", x=1985, y=max(fg3year$pctfg3)+0.6, label=paste(toString(round(max(fg3year$pctfg3), di
  annotate("text", x=1985, y=min(fg3yearteam$pctfg3)+0.6, label=paste(toString(round(min(fg3yearteam$pct
  annotate("text", x=1985, y=max(fg3yearteam$pctfg3)+0.6, label=paste(toString(round(max(fg3yearteam$pct

Q1
```



Yes, the success rate of 3 point field goal has been increased by about 9% since 1986.

Q2. If true, what could be the reasons for that? - What are the expected average points of 3-pointers and 2-pointers? Show the historical data. - If the expected average point from 3-pointers is getting higher than that of 2-pointers, how should each team's strategy changes

https://www.nytimes.com/2016/01/21/sports/basketball/how-the-nba-3-point-shot-went-from-gimmick-to-game-changer.html

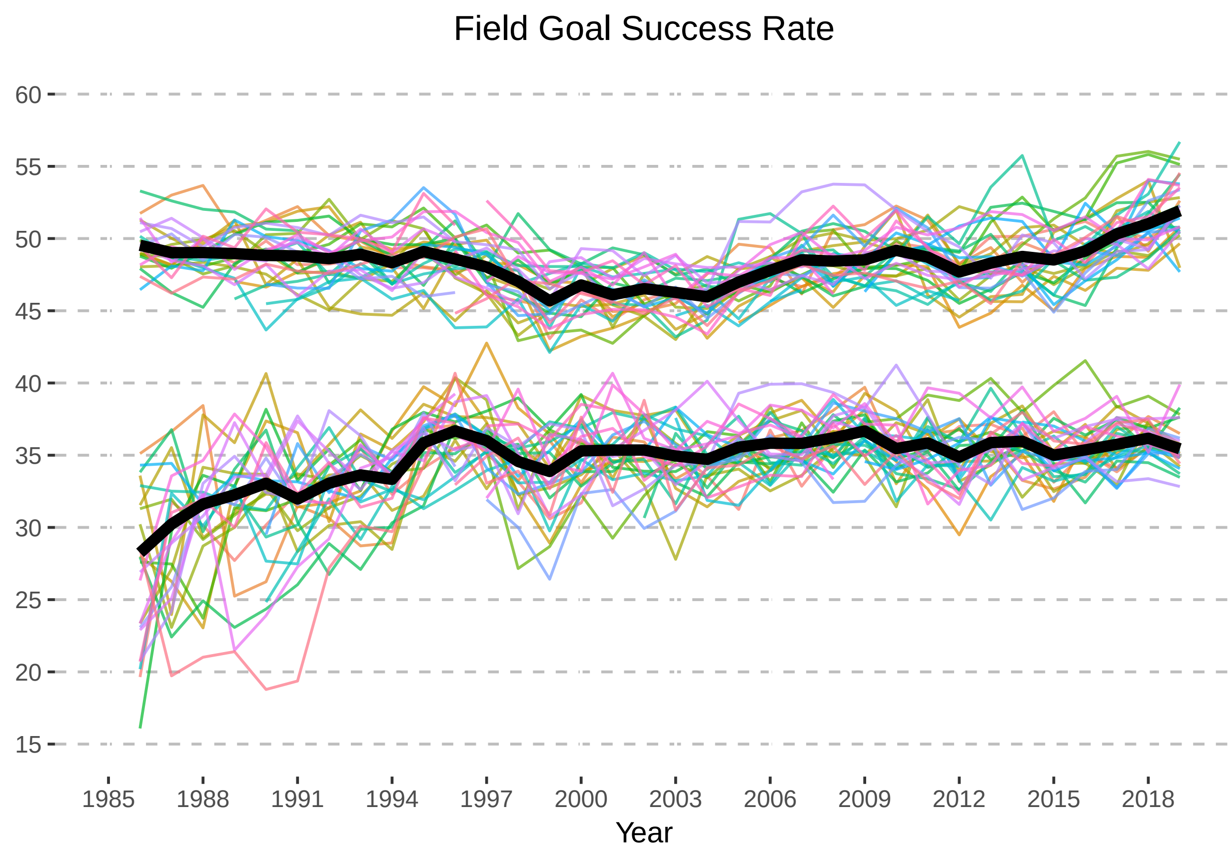
Its debut, in the 1979-80 season, was inauspicious.

There are many reasons for the rise of the 3-point shot, but one may simply be math. It took a while, but coaches finally stopped listening to the traditionalist naysayers and realized that a shot that is worth 50 percent more pays off, even if that shot is a little harder to make.

"Teams have all caught on to the whole points-per-possession argument," Lawrence Frank, the Nets' coach at the time, said in 2009 as the 3 rate began to rapidly increase.

```
fgyear <- aggregate(dataGameLogsTeam[, 35:38], list(dataGameLogsTeam$yearSeason), sum)
colnames(fgyear)[1] <- "Year"
fgyear <- fgyear %>% filter (Year >= 1986)
fgyear$pctfg3 <- fgyear$fg3mTeam / fgyear$fg3aTeam * 100
fgyear$pctfg2 <- fgyear$fg2mTeam / fgyear$fg2aTeam * 100

fgyearteam <- aggregate(dataGameLogsTeam[, 35:38], list(dataGameLogsTeam$yearSeason, dataGameLogsTeam$Te
colnames(fgyearteam)[1] <- "Year"
colnames(fgyearteam)[2] <- "Team"
fgyearteam <- fgyearteam %>% filter (Year >= 1986)
fgyearteam$pctfg3 <- fgyearteam$fg3mTeam / fgyearteam$fg3aTeam * 100
```

```
fgyearteam$pctfg2 <- fgyearteam$fg2mTeam / fgyearteam$fg2aTeam * 100

xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(15, 60, by=5)

Q2_1 <- ggplot() +
  geom_line(data=fgyearteam, aes(x=Year, y=pctfg3, colour=Team), size=0.5, show.legend=FALSE, alpha=0.7)
  geom_line(data=fgyear, aes(x=Year, y=pctfg3), size=2, colour='black') +
  geom_line(data=fgyearteam, aes(x=Year, y=pctfg2, colour=Team), size=0.5, show.legend=FALSE, alpha=0.7)
  geom_line(data=fgyear, aes(x=Year, y=pctfg2), size=2, colour='black') +
  xlab('Year') +
  ylab(NULL) +
  ggtitle('Field Goal Success Rate') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=2
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(15, 60), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)# +
  # geom_hline(yintercept=min(fg3year$pctfg3), linetype=2, color="steelblue", size=0.5, alpha=0.9) +
  # geom_hline(yintercept=max(fg3year$pctfg3), linetype=2, color="steelblue", size=0.5, alpha=0.9) +
  # geom_hline(yintercept=min(fg3yearteam$pctfg3), linetype=3, color="pink", size=0.5, alpha=0.9) +
  # geom_hline(yintercept=max(fg3yearteam$pctfg3), linetype=3, color="pink", size=0.5, alpha=0.9) +
  # annotate("text", x=1985, y=min(fg3year$pctfg3)+0.6, label=paste(toString(round(min(fg3year$pctfg3),
  # annotate("text", x=1985, y=max(fg3year$pctfg3)+0.6, label=paste(toString(round(max(fg3year$pctfg3),
  # annotate("text", x=1985, y=min(fg3yearteam$pctfg3)+0.6, label=paste(toString(round(min(fg3yearteam$
  # annotate("text", x=1985, y=max(fg3yearteam$pctfg3)+0.6, label=paste(toString(round(max(fg3yearteam$
  #
Q2_1
```



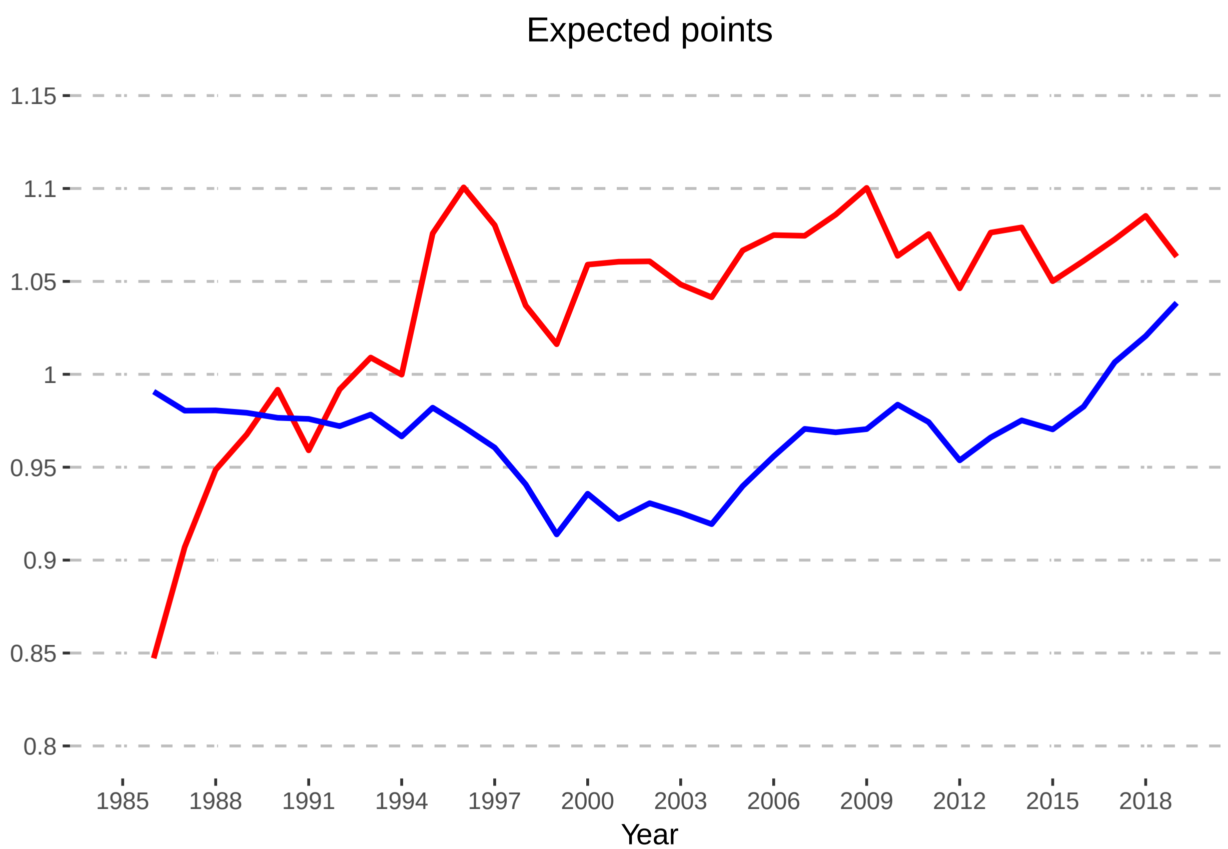
The expected points of 2-point shots in 1986 was ‘r fgyear$pctfg2[1986-1985]/100'*2 =' rfgyear$pctfg2[1986-1985]/100*2’ *The expected points of 3-point shots in 1986 was* ’r fgyear$pctfg3[1986-1985]/100' * 3 =' *rfgyear$pctfg3[1986-1985]/100*3’

The expected points of 2-point shots in 2019 was ‘r fgyear$pctfg2[2019-1985]/100'*2 =' rfgyear$pctfg2[2019-1985]/100*2’ *The expected points of 3-point shots in 2019 was* ’r fgyear$pctfg3[2019-1985]/100' * 3 =' *rfgyear$pctfg3[2019-1985]/100*3’

Teams started to focus on 3-point shots after its first introduction in 1979, because the expected points of 3-point shots are higher than that of 2-point shots since early 90’s.

```
fgyear$e2 = fgyear$pctfg2 / 100 * 2
fgyear$e3 = fgyear$pctfg3 / 100 * 3

xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(0.8, 1.15, by=0.05)

Q2_2 <- ggplot() +
  geom_line(data=fgyear, aes(x=Year, y=e3), size=1, colour='red') +
  geom_line(data=fgyear, aes(x=Year, y=e2), size=1, colour='blue') +
  xlab('Year') +
  ylab(NULL) +
  ggtitle('Expected points') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(0.8, 1.15), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
```

```
Q2_2
```

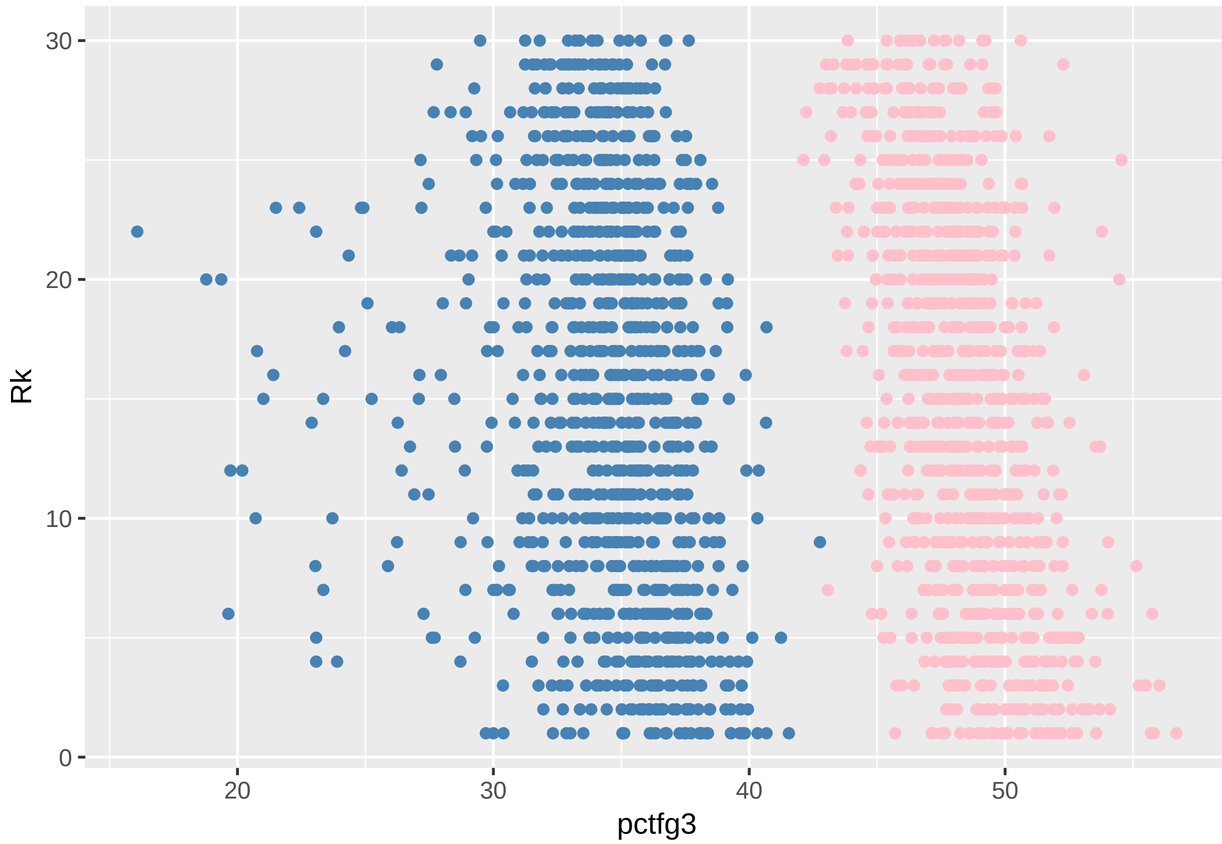


Q3. Teams with more 3-pointers tend to be the better performing teams? - Any insights between standings and 3-pointers?

```
standings <- read_csv("standings.csv")

fgyearteam <- aggregate(dataGameLogsTeam[, 35:38], list(dataGameLogsTeam$yearSeason, dataGameLogsTeam$na
colnames(fgyearteam)[1] <- "Year"
colnames(fgyearteam)[2] <- "nameTeam"
fgyearteam <- fgyearteam %>% filter (Year >= 1986)
fgyearteam$pctfg3 <- fgyearteam$fg3mTeam / fgyearteam$fg3aTeam * 100
fgyearteam$pctfg2 <- fgyearteam$fg2mTeam / fgyearteam$fg2aTeam * 100

standings2 <- left_join(standings, fgyearteam, by=c("Year" = "Year", "Team" = "nameTeam"))

Q3 <- ggplot(standings2) +
  geom_point(aes(x=pctfg3, y=Rk), color="steelblue") +
  geom_point(aes(x=pctfg2, y=Rk), color="pink")
  # geom_line(data=fgyear, aes(x=Year, y=e2), size=1, colour='blue') +
  # xlab('Year') +
  # ylab(NULL) +
  # ggtitle('Expected points') +
  # theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetyp
  #       plot.title = element_text(hjust = 0.5)) +
```

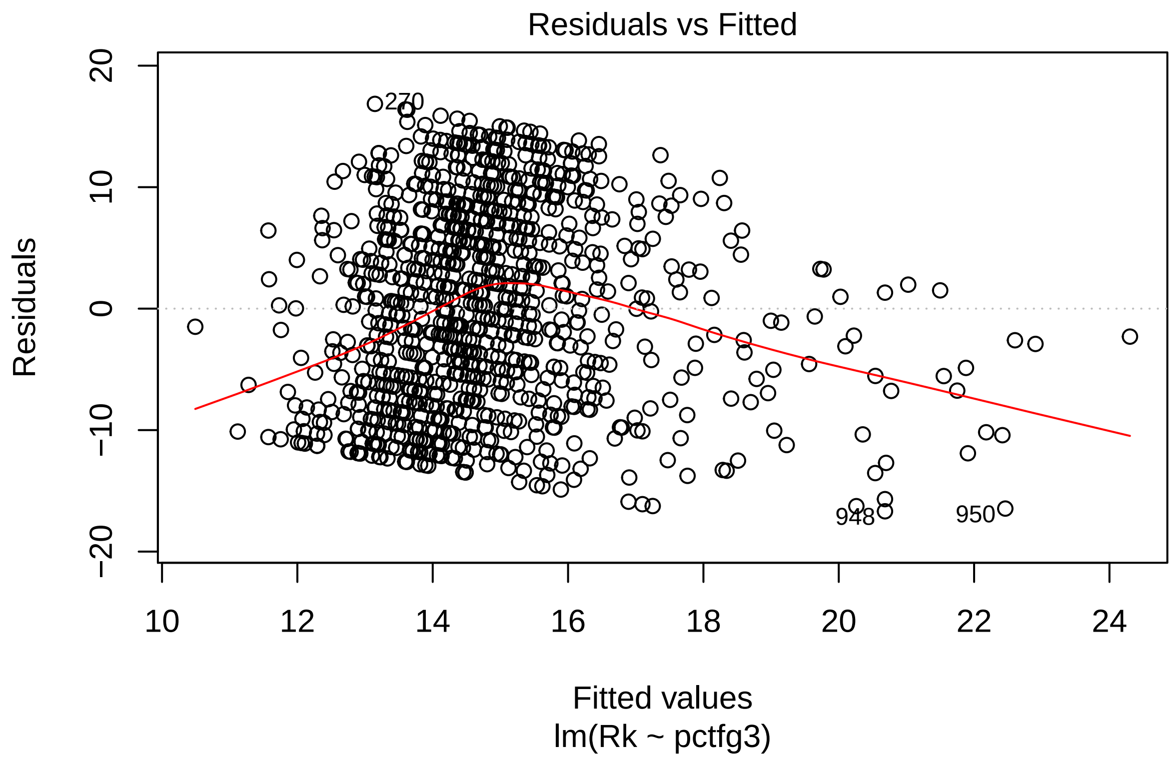
```
# scale_y_continuous(limits=c(0.8, 1.15), breaks=yaxisbreaks, labels=yaxisbreaks) +
# scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
```

Q3

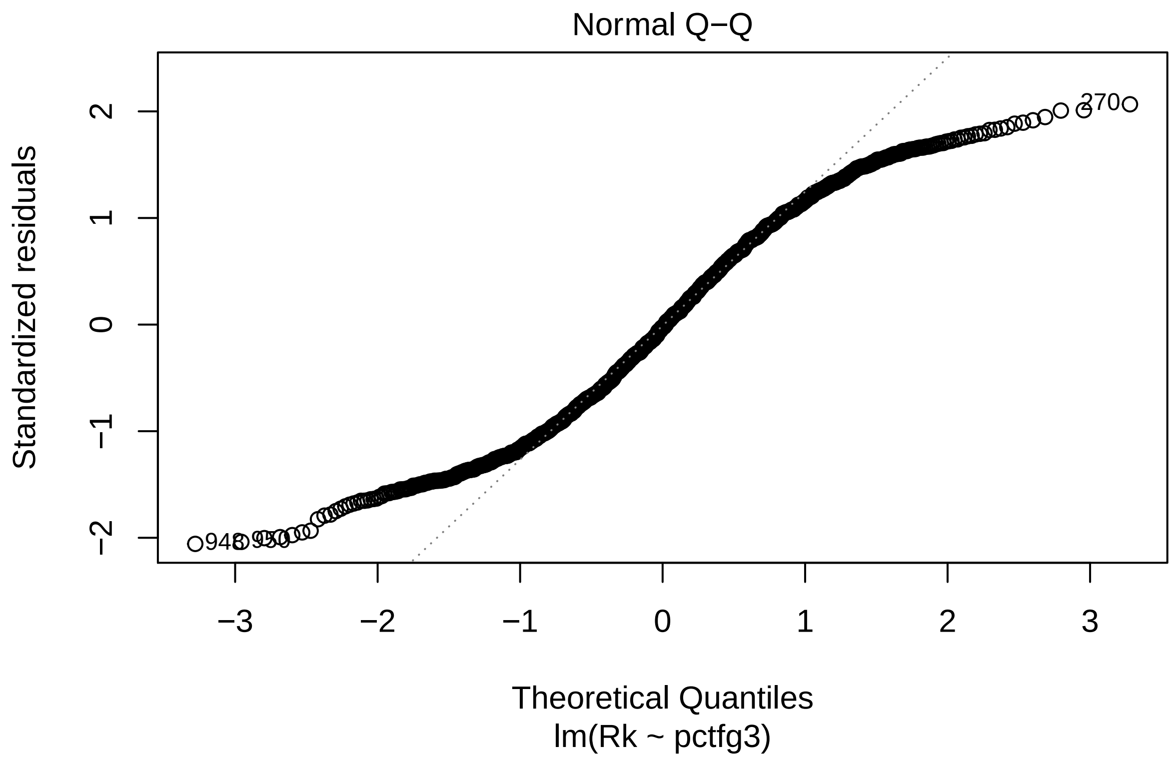



```
linearModel <- lm(Rk ~ pctfg3, data=standings2)
summary(linearModel)

Call:
lm(formula = Rk ~ pctfg3, data = standings2)

Residuals:
    Min      1Q  Median      3Q     Max
-16.683  -6.997  -0.212   6.831  16.854

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  32.6295     2.7198   12.00  < 2e-16 ***
pctfg3       -0.5177     0.0787   -6.58  7.7e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.16 on 961 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.0431,    Adjusted R-squared:  0.0421
```

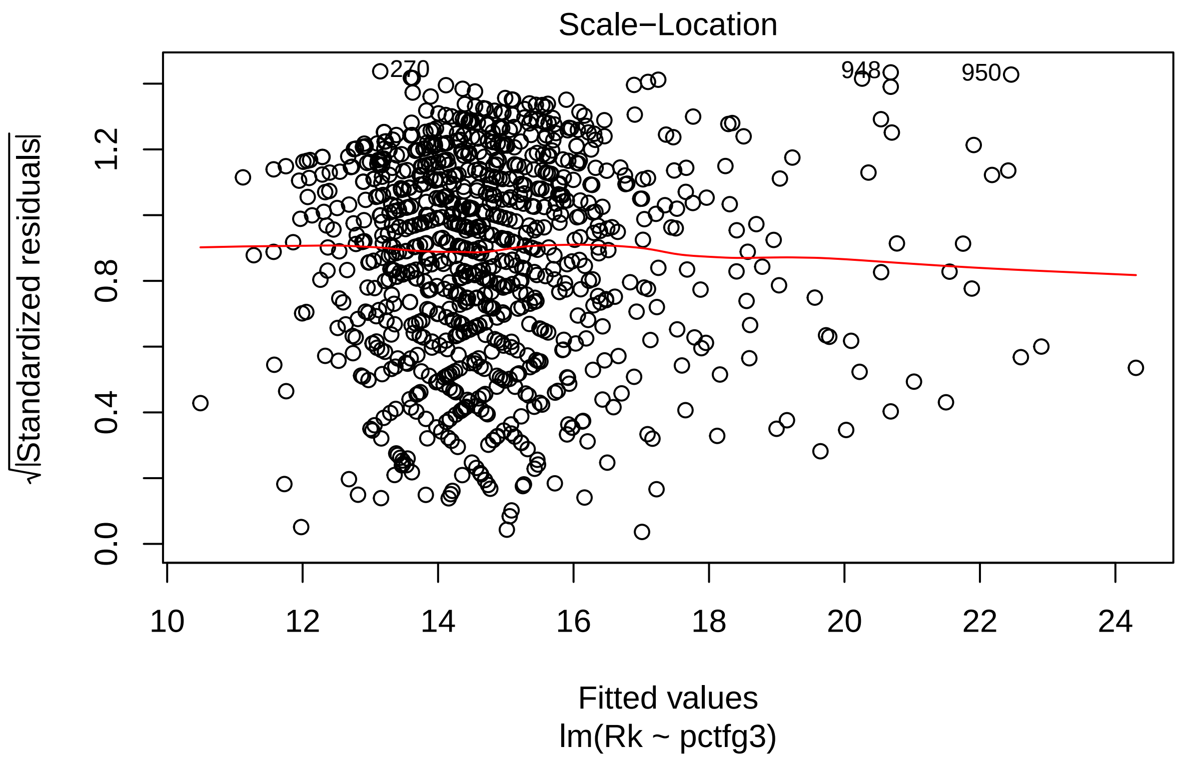
```
F-statistic: 43.3 on 1 and 961 DF,  p-value: 7.74e-11

linearModel2 <- lm(Rk ~ pctfg2, data=standings2)
summary(linearModel2)

Call:
lm(formula = Rk ~ pctfg2, data = standings2)

Residuals:
    Min      1Q  Median      3Q     Max 
-18.887  -5.418   0.012   5.334  21.975 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  107.039      4.965    21.6   <2e-16 ***
pctfg2        -1.907      0.103   -18.6   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.16 on 961 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.265, Adjusted R-squared:  0.264 
F-statistic:  346 on 1 and 961 DF,  p-value: <2e-16

linearModel3 <- lm(Rk ~ pctfg3 + pctfg2, data=standings2)
summary(linearModel3)

Call:
lm(formula = Rk ~ pctfg3 + pctfg2, data = standings2)

Residuals:
    Min      1Q  Median      3Q     Max 
-18.664  -5.402  -0.067   5.285  21.494 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 113.7368     5.1490    22.1  < 2e-16 ***
pctfg3       -0.3049     0.0694    -4.4  1.2e-05 ***
pctfg2       -1.8284     0.1031   -17.7  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.09 on 960 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.279, Adjusted R-squared:  0.278 
F-statistic:  186 on 2 and 960 DF,  p-value: <2e-16

plot(linearModel)
```

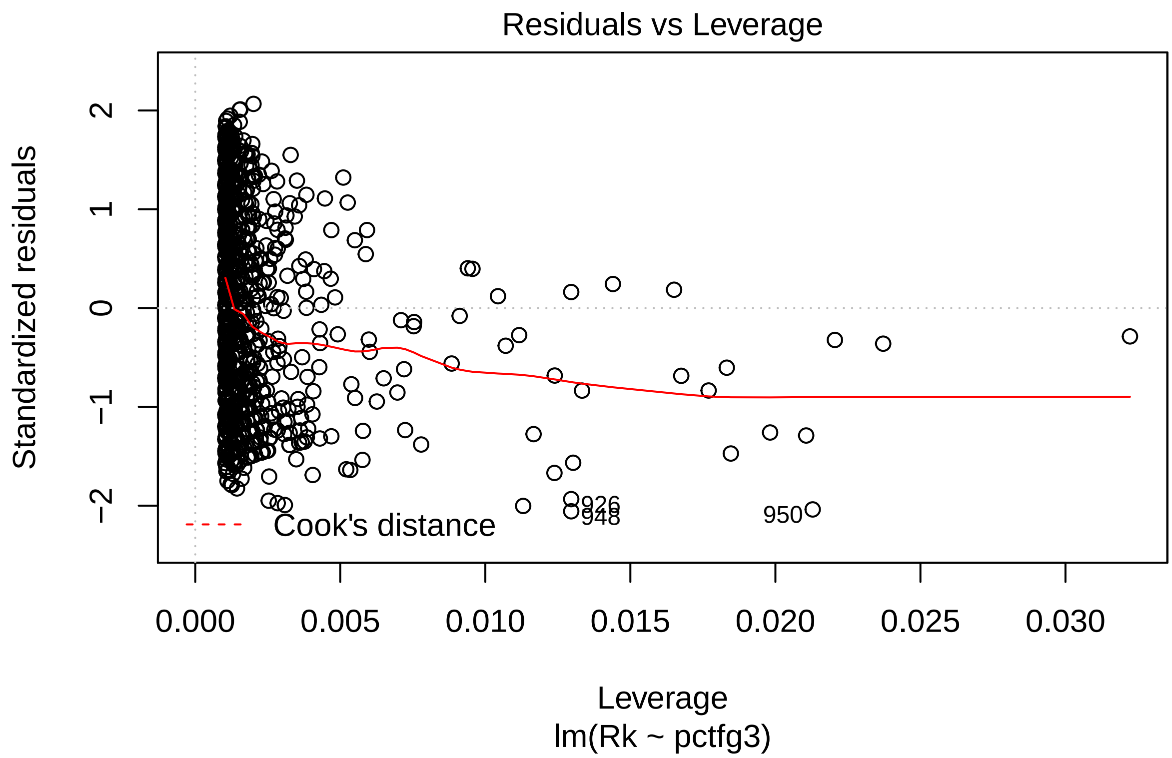


Residuals vs Fitted
Residuals
Fitted values
lm(Rk ~ pctfg3)
270
948
950
20
10
0
−10
−20
10
12
14
16
18
20
22
24

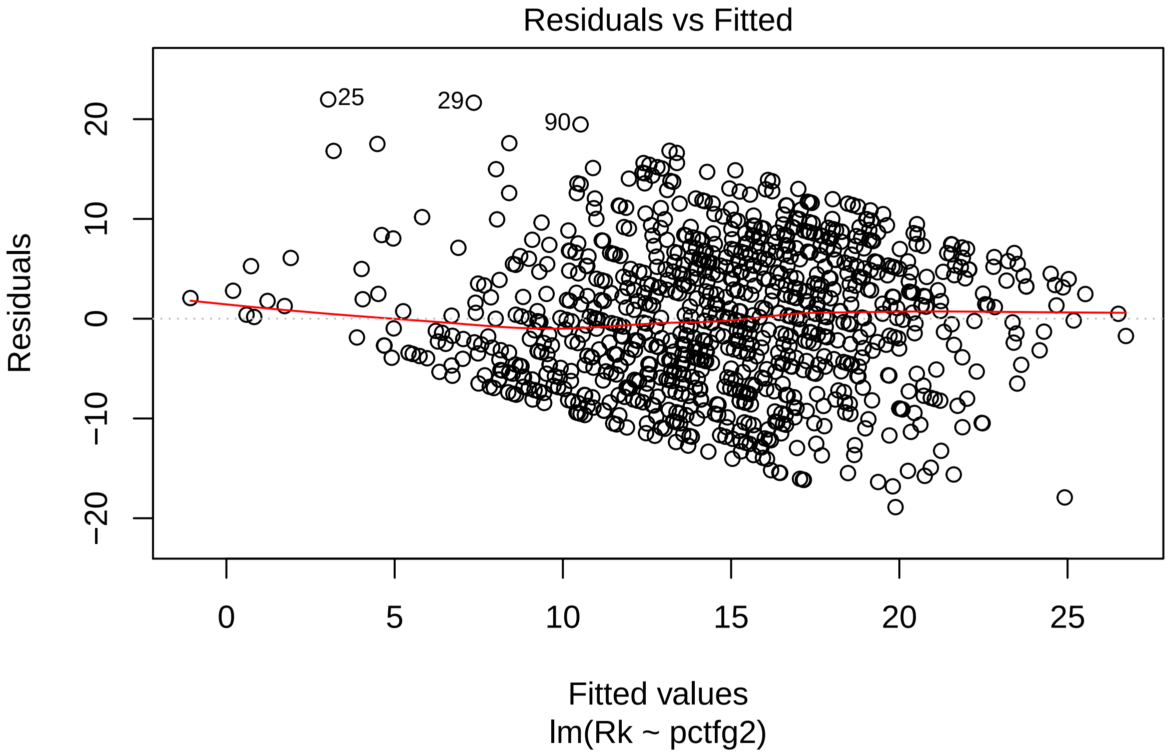


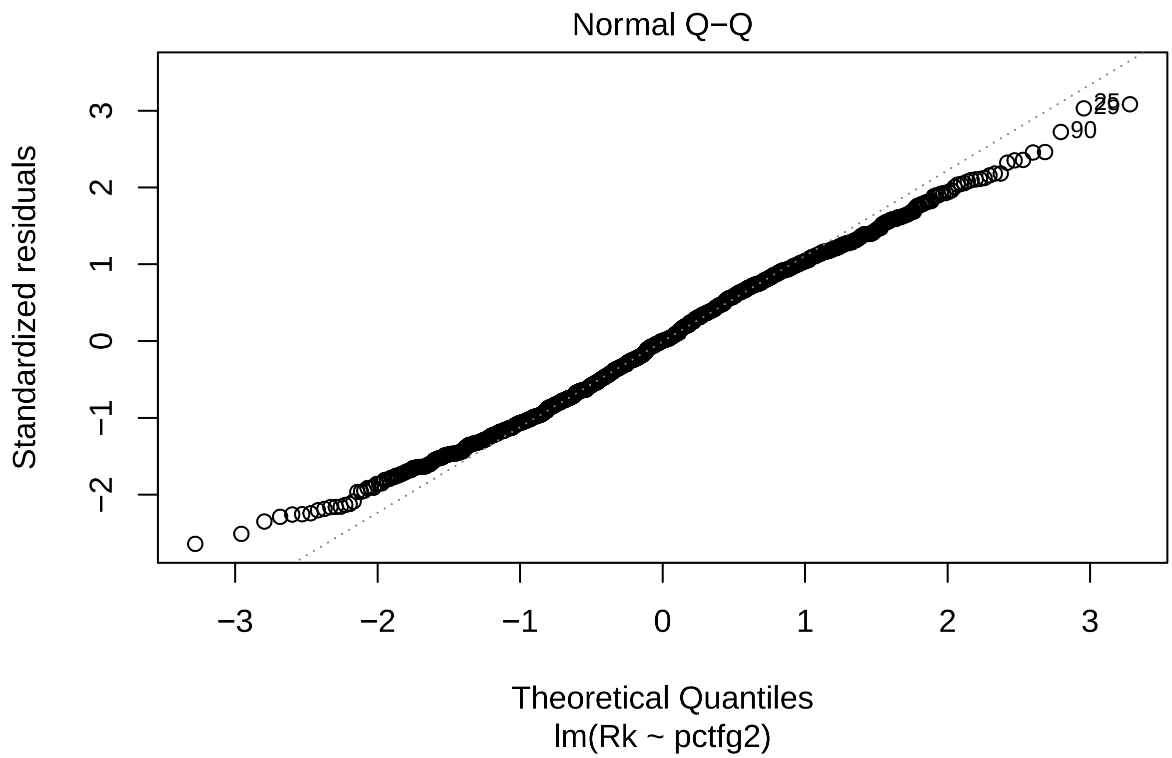
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Rk ~ pctfg3)
270
948
−3
−2
−1
0
1
2
3


Scale−Location
270
948
950
√|Standardized residuals|
0.0
0.4
0.8
1.2
10
12
14
16
18
20
22
24
Fitted values
lm(Rk ~ pctfg3)

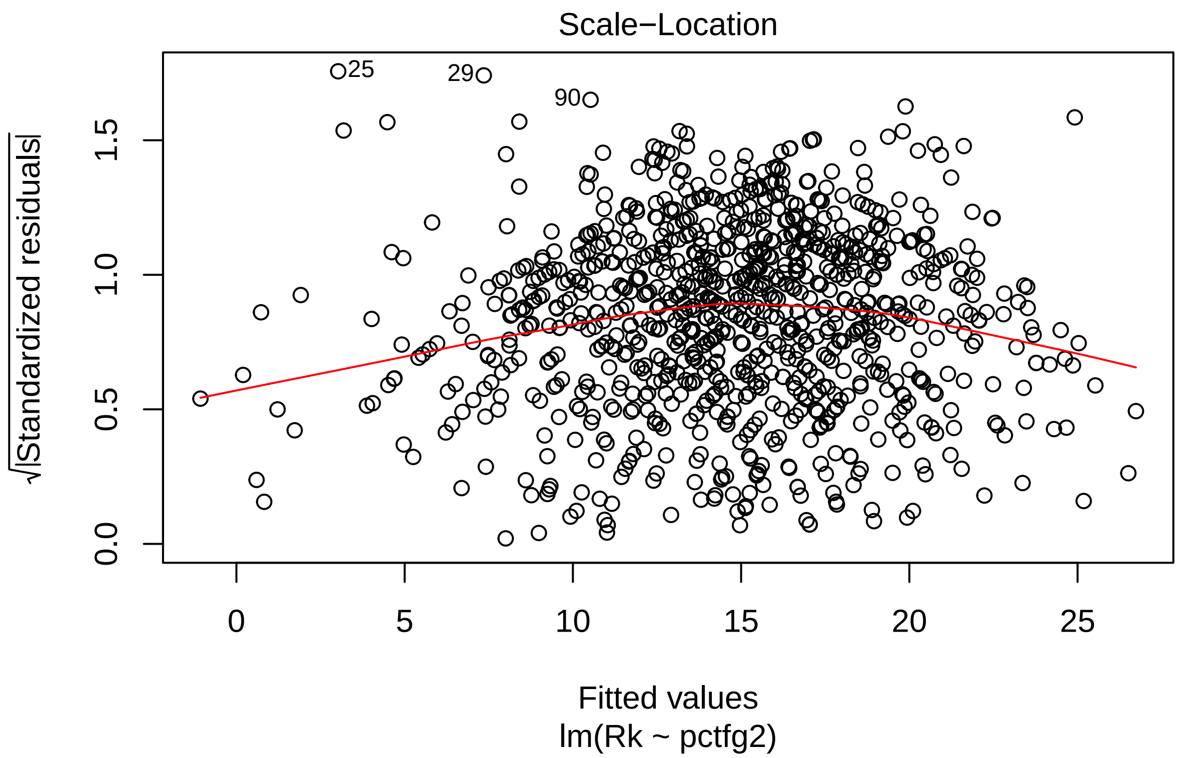



```r
plot(linearModel2)
```

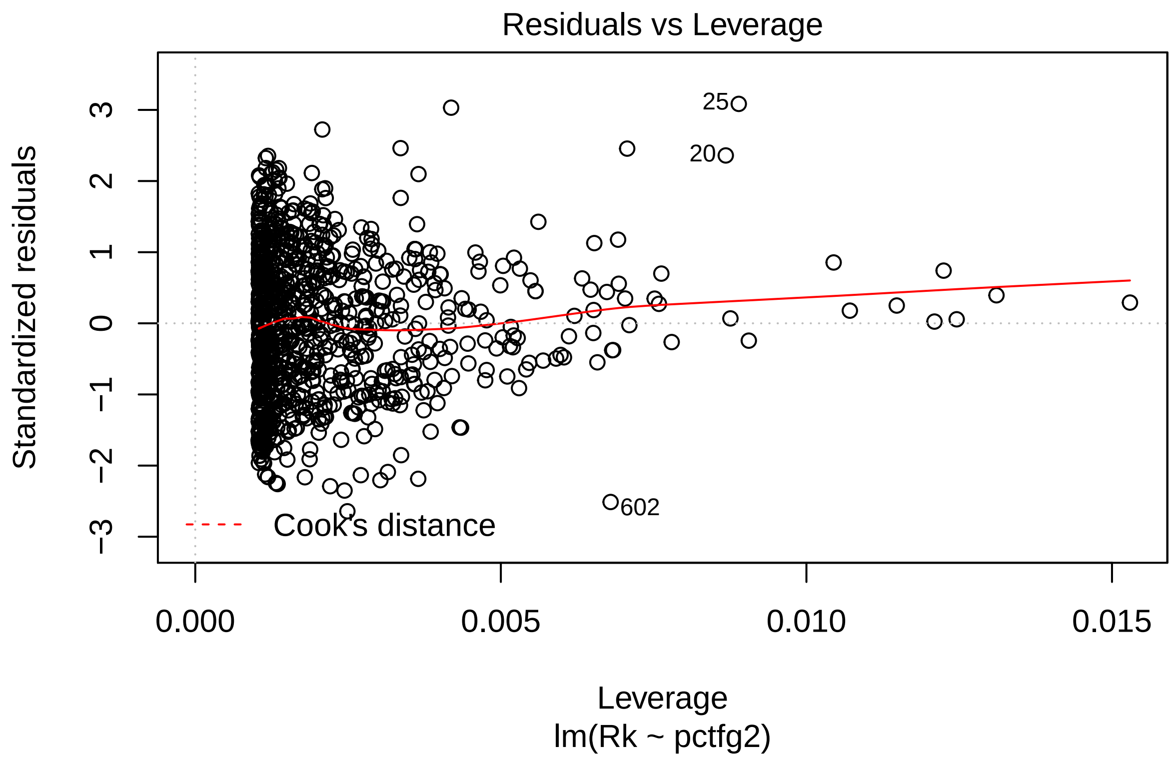

Residuals vs Fitted
Residuals
Fitted values
lm(Rk ~ pctfg2)
25
29
90
20
10
0
−10
−20
0
5
10
15
20
25



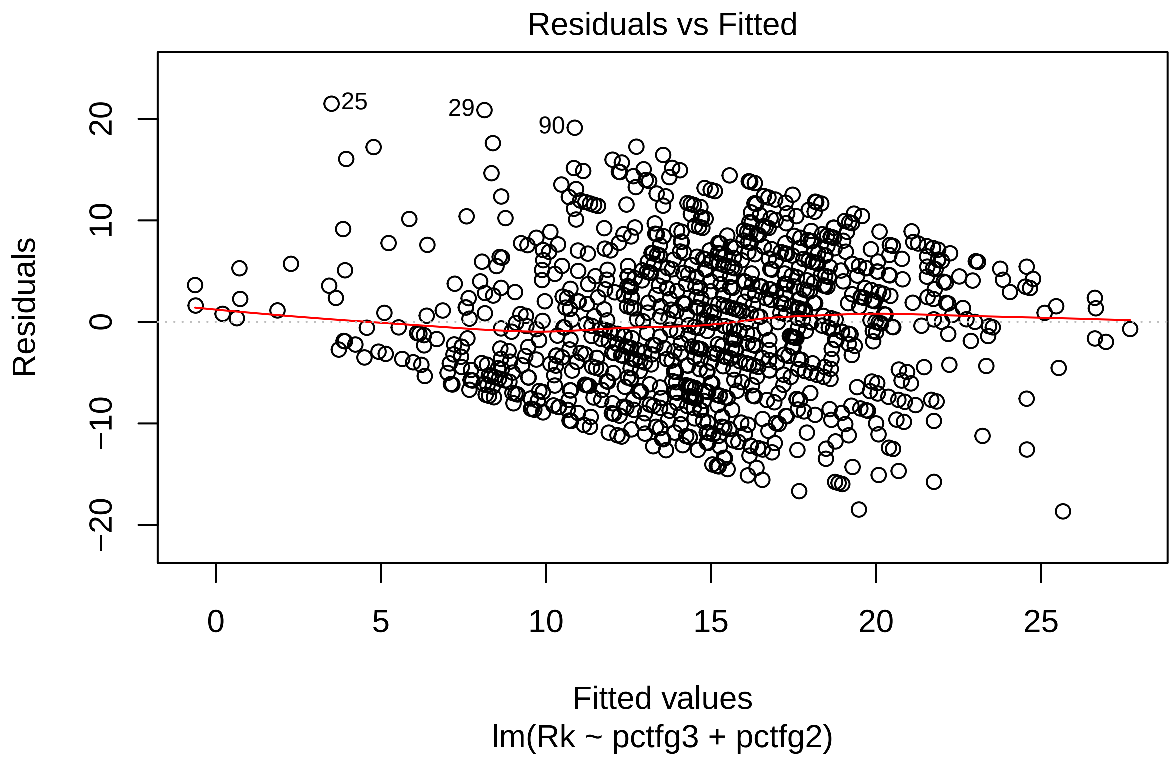
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Rk ~ pctfg2)
25
29
90
−3
−2
−1
0
1
2
3

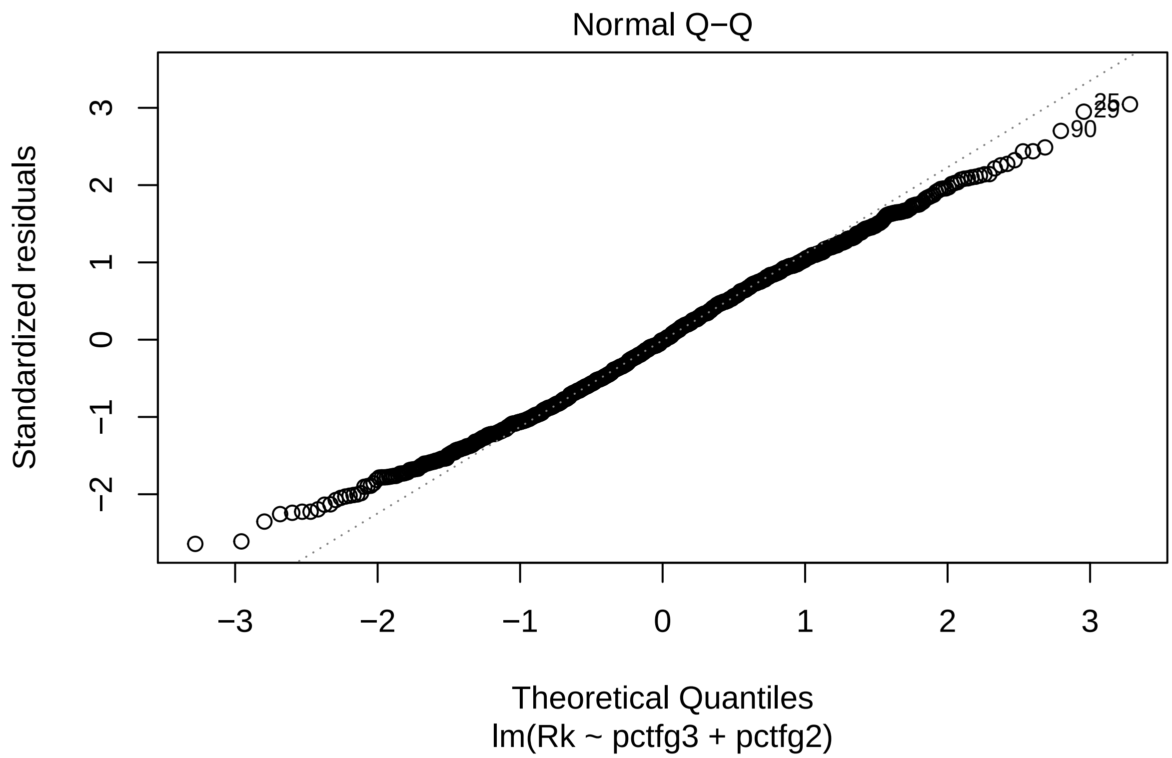


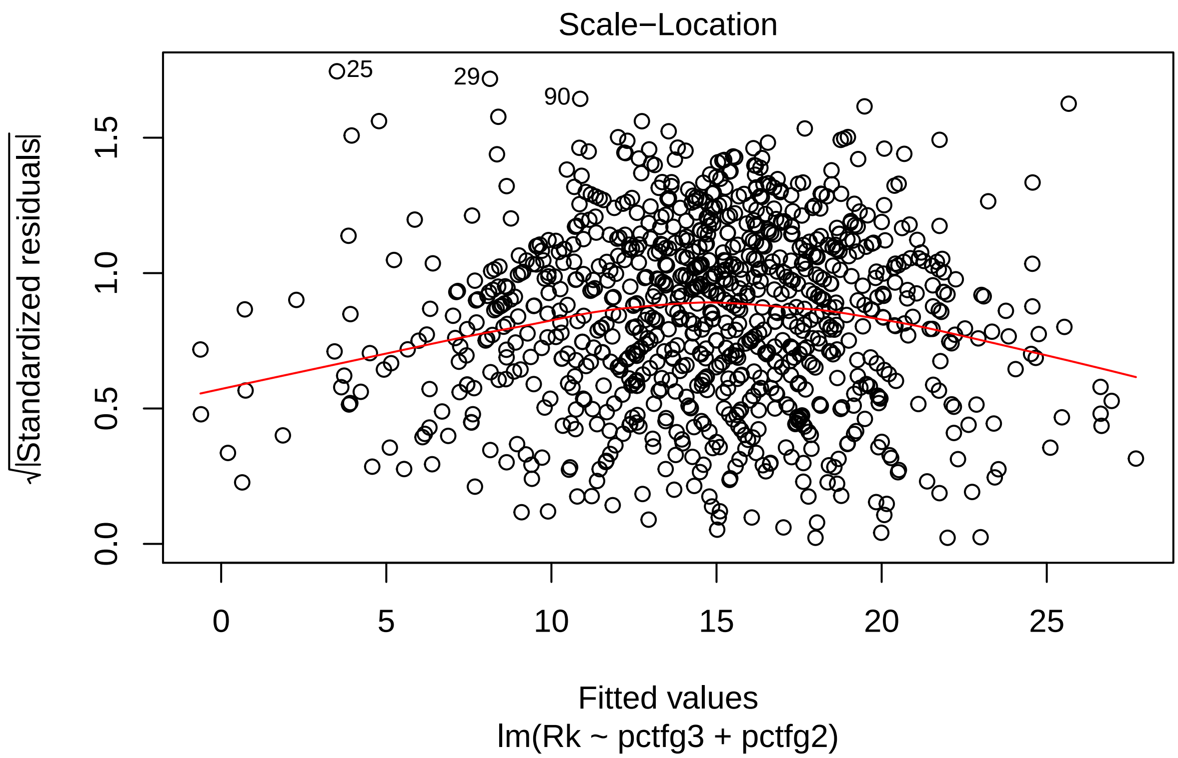
Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
25
29
90
0
5
10
15
20
25
Fitted values
lm(Rk ~ pctfg2)

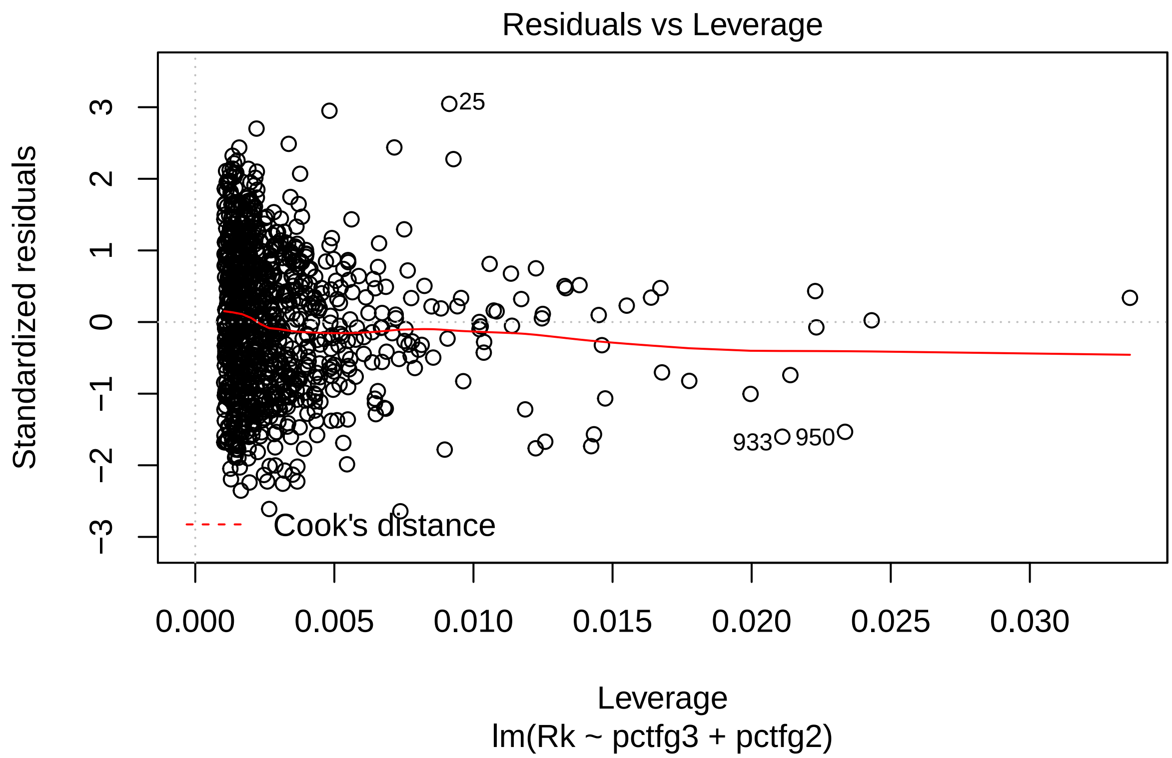



```
plot(linearModel3)
```

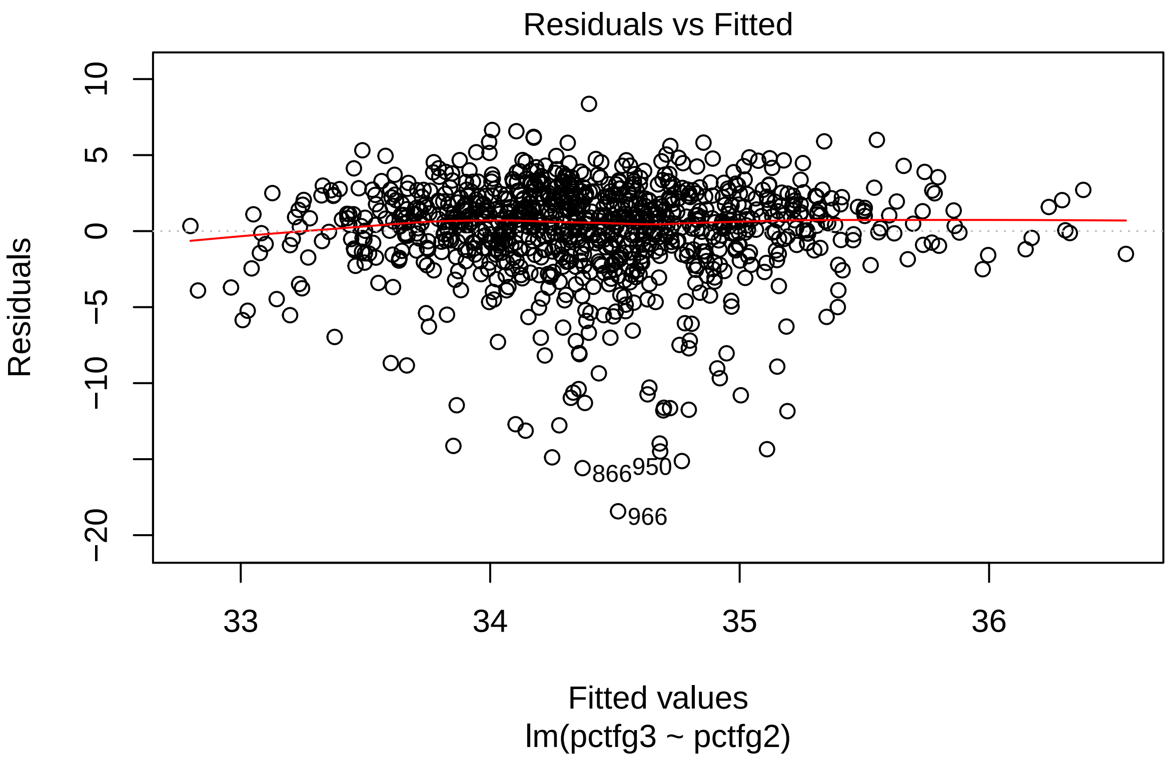

Residuals vs Fitted
Residuals
Fitted values
lm(Rk ~ pctfg3 + pctfg2)
25
29
90
20
10
0
−10
−20
0
5
10
15
20
25

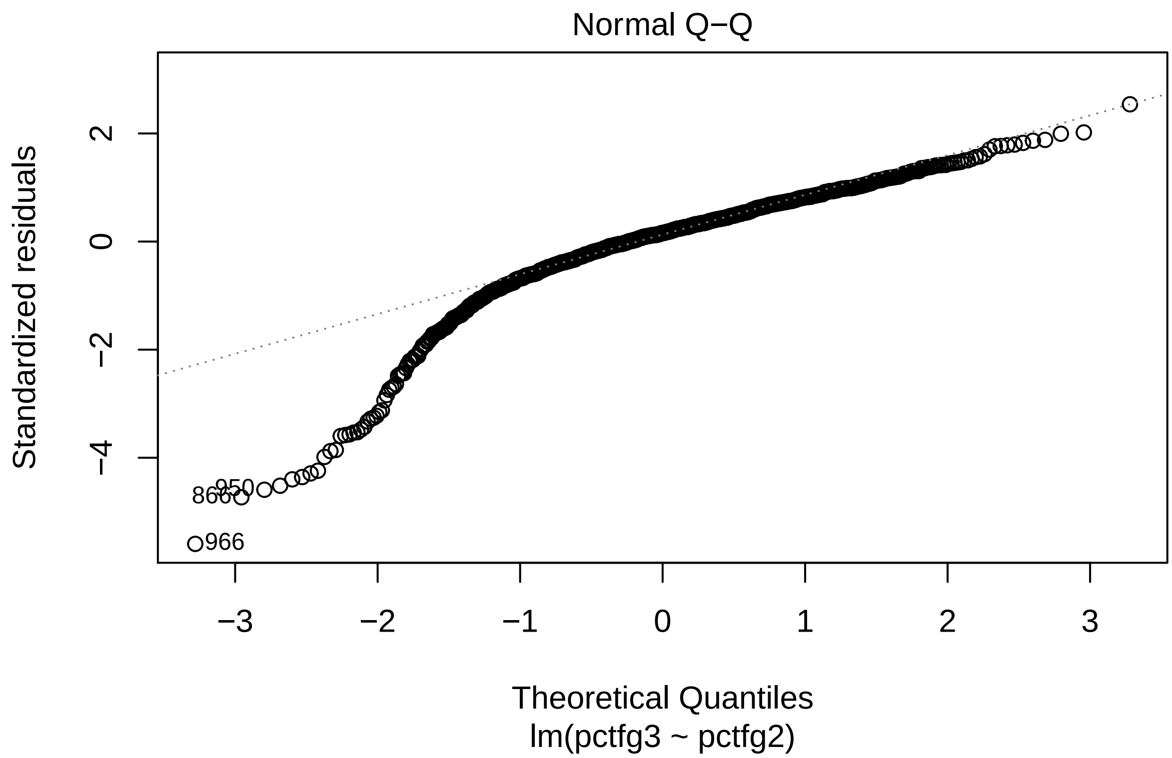

Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Rk ~ pctfg3 + pctfg2)
25
29
90
−3
−2
−1
0
1
2
3

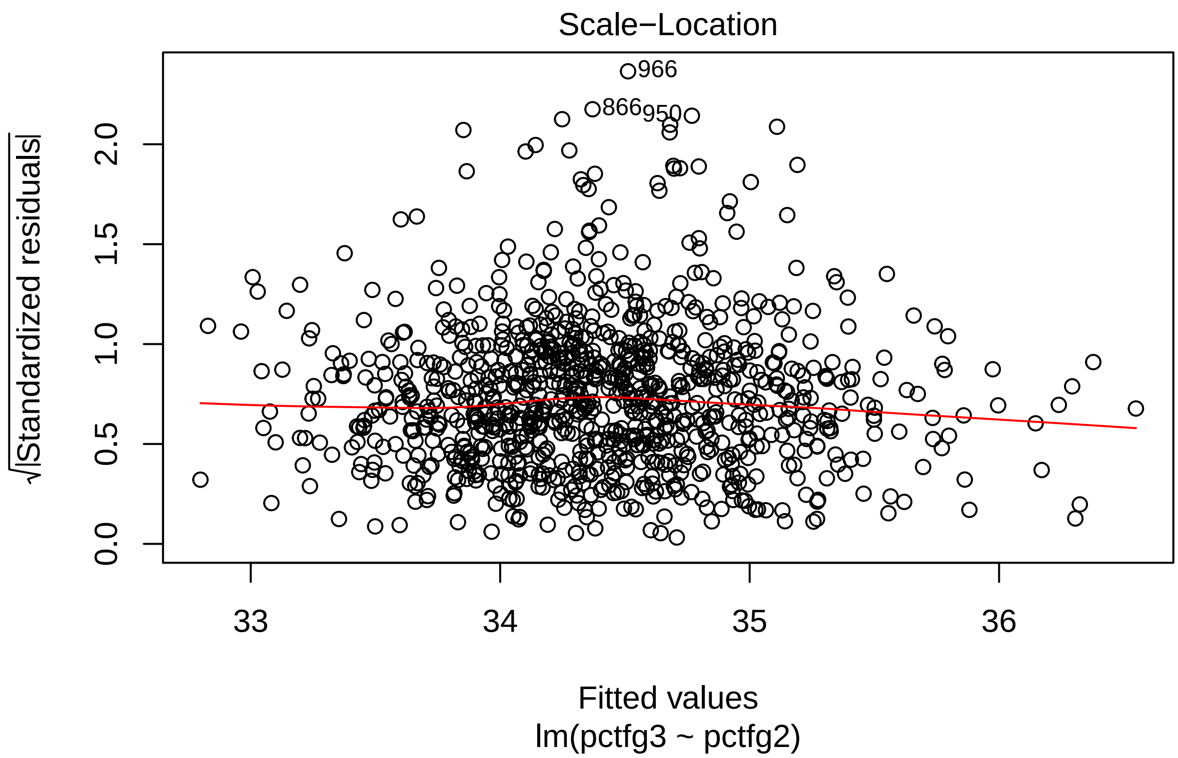

Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
0
5
10
15
20
25
25
29
90
Fitted values
lm(Rk ~ pctfg3 + pctfg2)



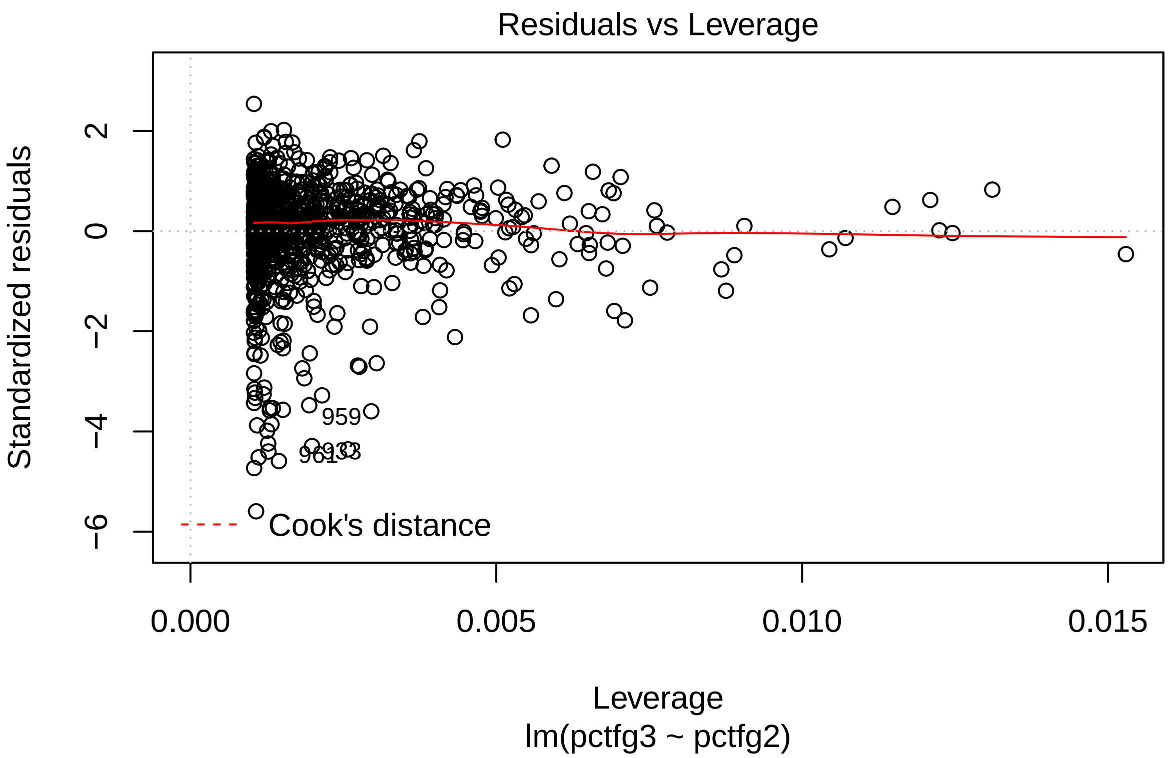

```
linearModel4 <- lm(pctfg3 ~ pctfg2, data=standings2)
summary(linearModel4)

Call:
lm(formula = pctfg3 ~ pctfg2, data = standings2)

Residuals:
    Min      1Q  Median      3Q     Max 
-18.429  -1.209   0.517   2.055   8.368 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  21.9658     2.2869    9.60  < 2e-16 ***
pctfg2        0.2572     0.0472    5.45  6.6e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.3 on 961 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.0299,	Adjusted R-squared:  0.0289 
F-statistic: 29.7 on 1 and 961 DF,  p-value: 6.57e-08
plot(linearModel4)
```


Residuals vs Fitted
Residuals
10
5
0
−5
−10
−20
33
34
35
36
866
950
966
Fitted values
lm(pctfg3 ~ pctfg2)

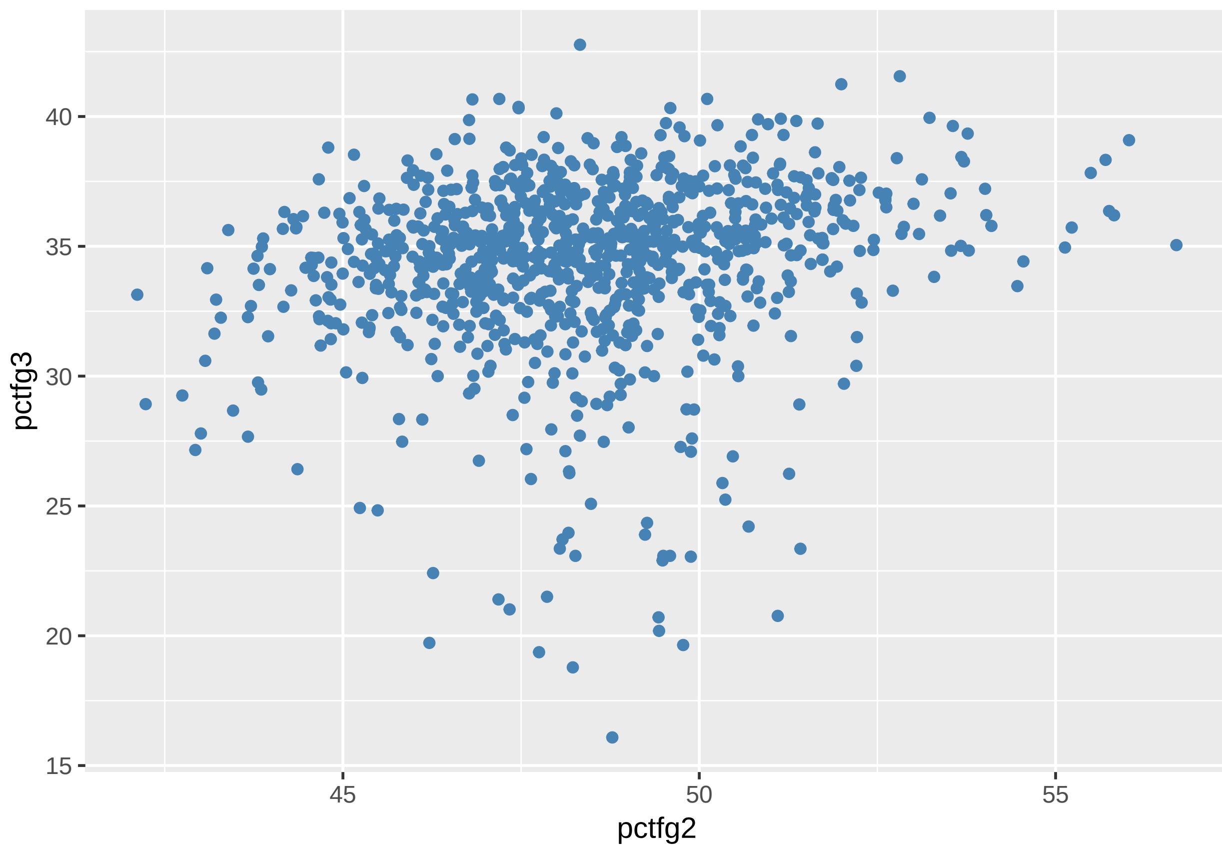

Normal Q−Q
Standardized residuals
2
0
−2
−4
866
950
966
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(pctfg3 ~ pctfg2)



Scale−Location
966
866
950
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
33
34
35
36
Fitted values
lm(pctfg3 ~ pctfg2)




```r
Q3_2 <- ggplot(standings2) +
  geom_point(aes(x=pctfg2, y=pctfg3), color="steelblue")
Q3_2
```

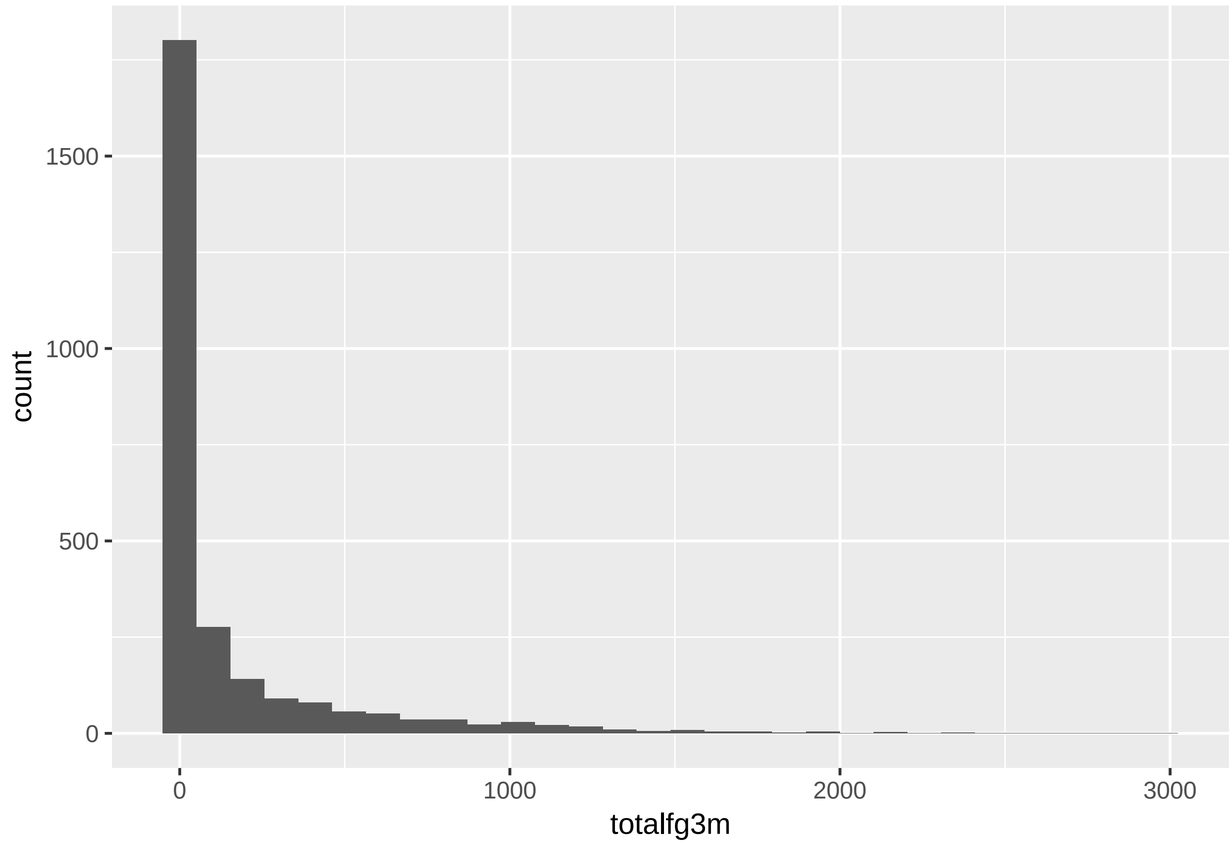


Yes. However, pctfg2 is more relevant than pctfg3

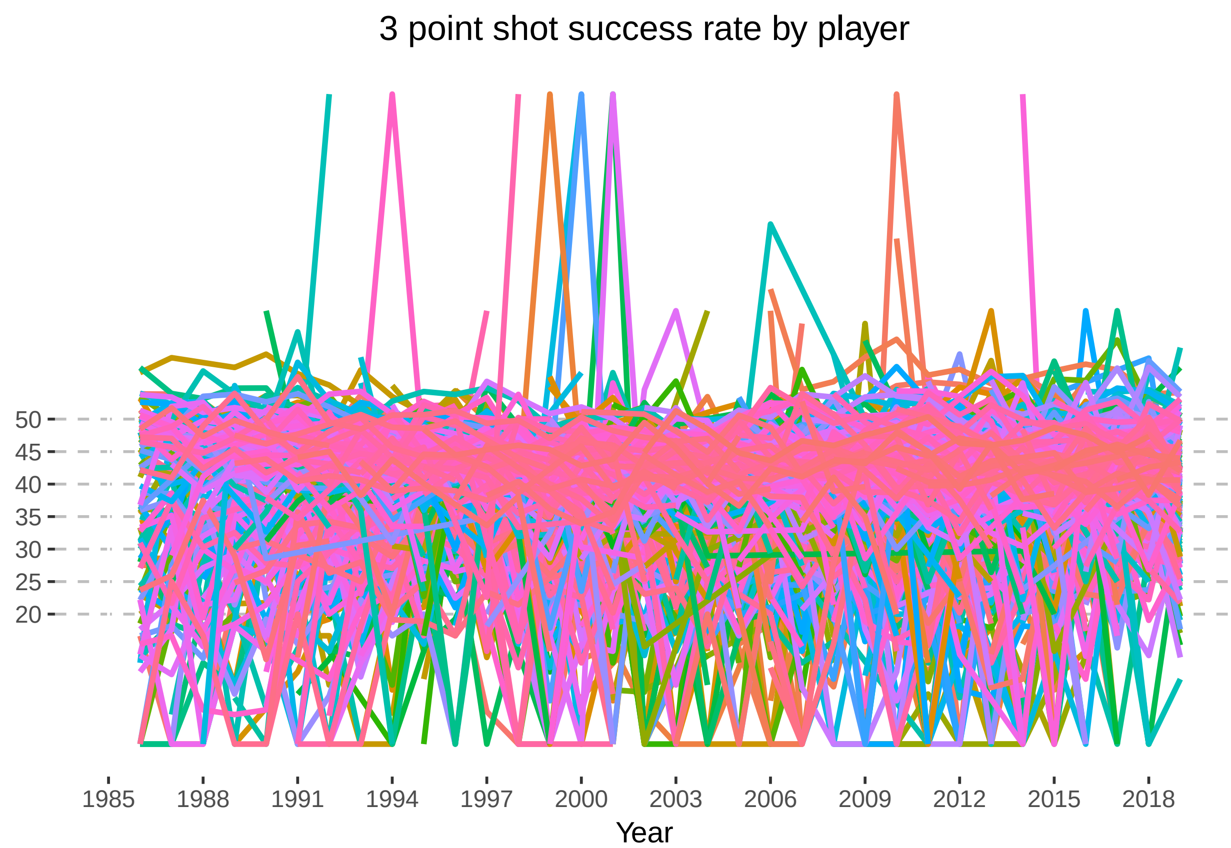
- Focus on three point shooting is a strategy that started fairly recently, we can create a map to show where this strategy initially emerged and how fast it spreaded across the entire country.

Player level questions

```
dataGameLogsPlayer1986 <- dataGameLogsPlayer %>% filter(yearSeason >= 1986)

fgyearplayer <- aggregate(dataGameLogsPlayer1986[, 19:26], list(dataGameLogsPlayer1986$yearSeason, data
colnames(fgyearplayer)[1] <- "Year"
colnames(fgyearplayer)[2] <- "Player"
fgyearplayer$pctFG = NULL
fgyearplayer$pctFG3 = NULL

fgyearplayer$pctfg3 <- fgyearplayer$fg3m / fgyearplayer$fg3a * 100
fgyearplayer$pctfg2 <- fgyearplayer$fgm / fgyearplayer$fga * 100
fgyearplayer$pctft <- fgyearplayer$ftm / fgyearplayer$fta * 100

# Meaningless...
yearplayer <- aggregate(fgyearplayer[,5], list(fgyearplayer$Player), sum)
colnames(yearplayer)[1] <- "Player"
colnames(yearplayer)[2] <- "totalfg3m"
ggplot(yearplayer, aes(totalfg3m)) + geom_histogram()
```




```
yearplayer100 <- yearplayer %>% filter (totalfg3m>=100)

xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(20, 50, by=5)

fgyearplayer100 <- fgyearplayer %>% filter(Player %in% yearplayer100$Player)
plotYearPlayer <- ggplot() +
  geom_line(data=fgyearplayer100, aes(x=Year, y=pctfg3, colour=Player), size=1, show.legend = FALSE) +
  geom_line(data=fgyearplayer100, aes(x=Year, y=pctfg2, colour=Player), size=1, show.legend = FALSE) +
  xlab('Year') +
  ylab(NULL) +
  ggtitle('3 point shot success rate by player') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=2
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(0, 100), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
plotYearPlayer
```



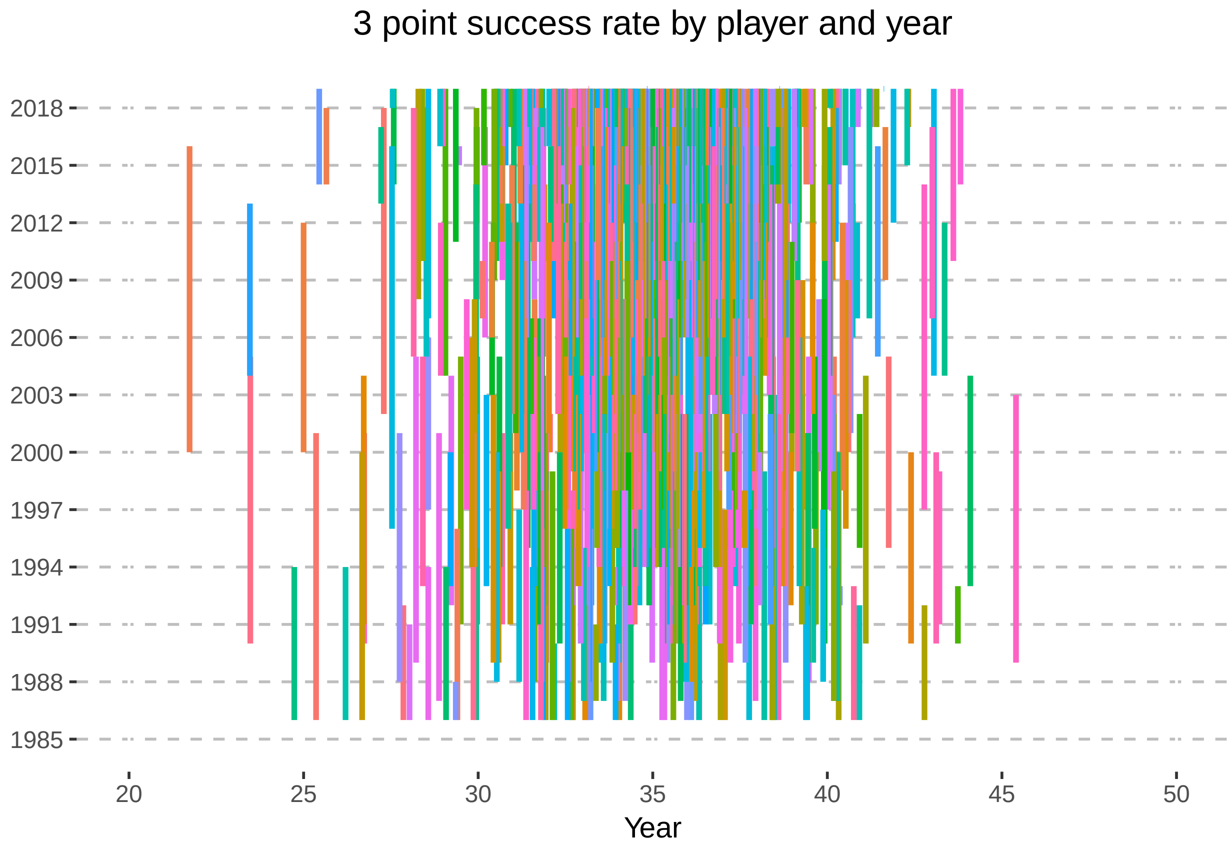

```
# Meaningless...

fgplayer <- aggregate(dataGameLogsPlayer1986[, 19:26], list(dataGameLogsPlayer1986$namePlayer), sum)
colnames(fgplayer)[1] <- "Player"
fgplayer$pctFG = NULL
fgplayer$pctFG3 = NULL

fgplayer$pctfg3 <- fgplayer$fg3m / fgplayer$fg3a * 100
fgplayer$pctfg2 <- fgplayer$fgm / fgplayer$fga * 100
fgplayer$pctft <- fgplayer$ftm / fgplayer$fta * 100

fgplayer <- fgplayer[order(-fgplayer$pctfg3),]
fgplayer100 <- fgplayer %>% filter(fg3m >= 100)
```

```
import pandas as pd

fgplayer = r.fgplayer

fgplayer['firstYear'] = 2019
fgplayer['lastYear'] = 1986

print(fgplayer.head(5))
          Player    fgm     fga    ...        pctft  firstYear  lastYear
0     Alvin Sims    4.0    10.0    ...    40.000000       2019      1986
1    Coty Clarke    2.0     4.0    ...          NaN       2019      1986
2     David Pope    9.0    19.0    ...    50.000000       2019      1986
```

```
3     Eddy Curry  2578.0  4734.0    ...     64.219474       2019      1986
4  Eric Anderson    12.0    35.0    ...     59.259259       2019      1986

[5 rows x 12 columns]
print(fgplayer.tail(5))
               Player    fgm    fga    ...          pctft  firstYear  lastYear
2720    Winston Crite   34.0   71.0    ...      76.000000       2019      1986
2721       Yinka Dare   86.0  217.0    ...      57.009346       2019      1986
2722      Yvon Joseph    0.0    0.0    ...     100.000000       2019      1986
2723   Zeljko Rebraca  488.0  926.0    ...      79.155673       2019      1986
2724  Zendon Hamilton  176.0  400.0    ...      66.005666       2019      1986

[5 rows x 12 columns]
i=0

for player in fgplayer.values:
  min = player[-2]
  max = player[-1]
  for yp in r.fgyearplayer.values:
    if player[0] == yp[1]:
      if max < yp[0]: max = yp[0]
      if min > yp[0]: min = yp[0]
  fgplayer.iloc[i,-1]=max
  fgplayer.iloc[i,-2]=min
  i += 1

print(fgplayer.head(5))
          Player     fgm     fga    ...          pctft  firstYear  lastYear
0     Alvin Sims     4.0    10.0    ...     40.000000       1999      1999
1    Coty Clarke     2.0     4.0    ...           NaN       2016      2016
2     David Pope     9.0    19.0    ...     50.000000       1986      1986
3     Eddy Curry  2578.0  4734.0    ...     64.219474       2002      2013
4  Eric Anderson    12.0    35.0    ...     59.259259       1993      1994

[5 rows x 12 columns]
print(fgplayer.tail(5))
               Player    fgm    fga    ...          pctft  firstYear  lastYear
2720    Winston Crite   34.0   71.0    ...      76.000000       1988      1989
2721       Yinka Dare   86.0  217.0    ...      57.009346       1995      1998
2722      Yvon Joseph    0.0    0.0    ...     100.000000       1986      1986
2723   Zeljko Rebraca  488.0  926.0    ...      79.155673       2002      2006
2724  Zendon Hamilton  176.0  400.0    ...      66.005666       2001      2006

[5 rows x 12 columns]
```

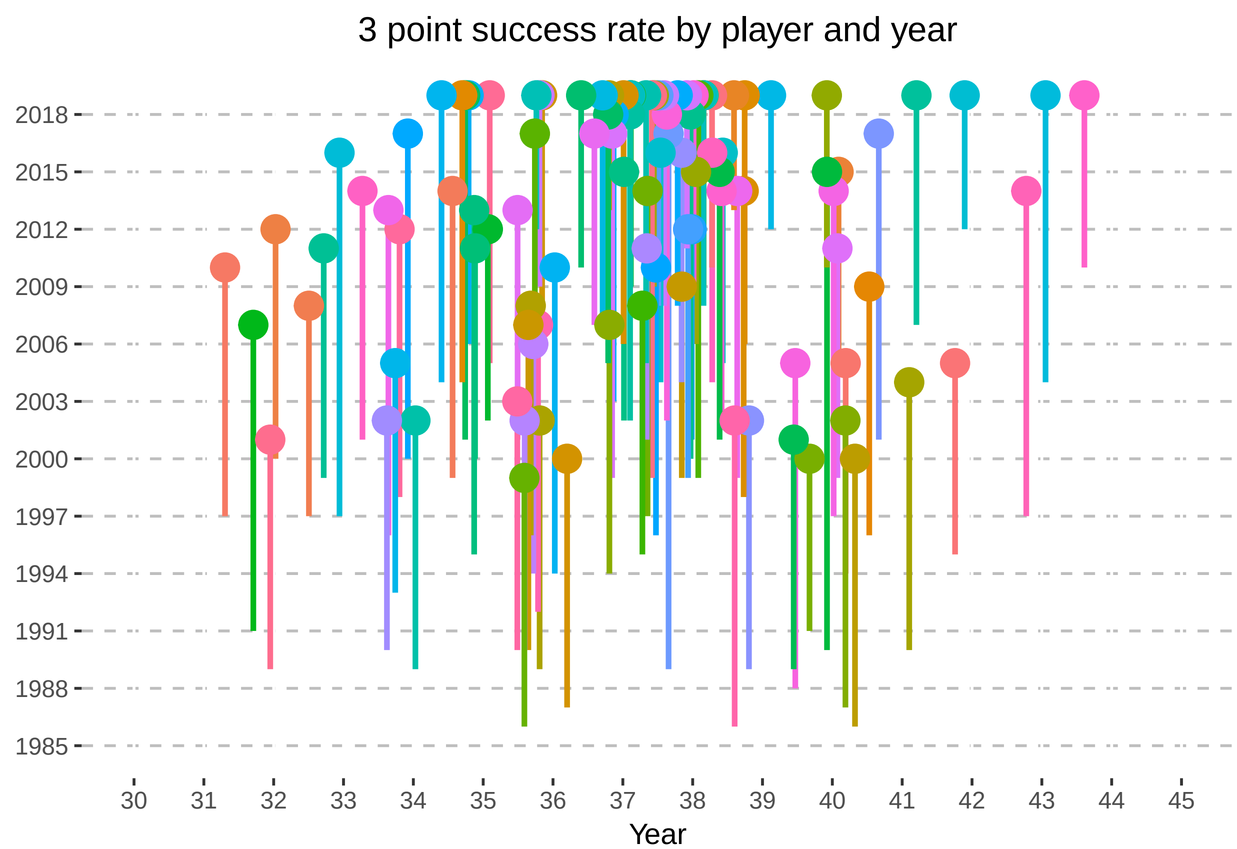
```
fgplayer <- py$fgplayer
fgplayer100 <- fgplayer %>% filter(fg3m >= 100)
fgplayer1000 <- fgplayer100 %>% filter(fg3m >= 1000)
fgplayer2000 <- fgplayer1000 %>% filter(fg3m >= 2000)


xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(20, 50, by=5)
```

```
plotPlayer100 <- ggplot() +
  geom_linerange(data=fgplayer100, aes(x=pctfg3, y=lastYear, ymin=firstYear, ymax=lastYear, colour=Playe
  # geom_point(data=fgplayer100, aes(x=lastYear, y=pctfg3, colour=Player), size=1, show.legend = FALSE)
  # geom_line
  xlab('Year') +
  ylab(NULL) +
  ggtitle('3 point success rate by player and year') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=2
        plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(limits=c(20, 50), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_y_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
plotPlayer100
```

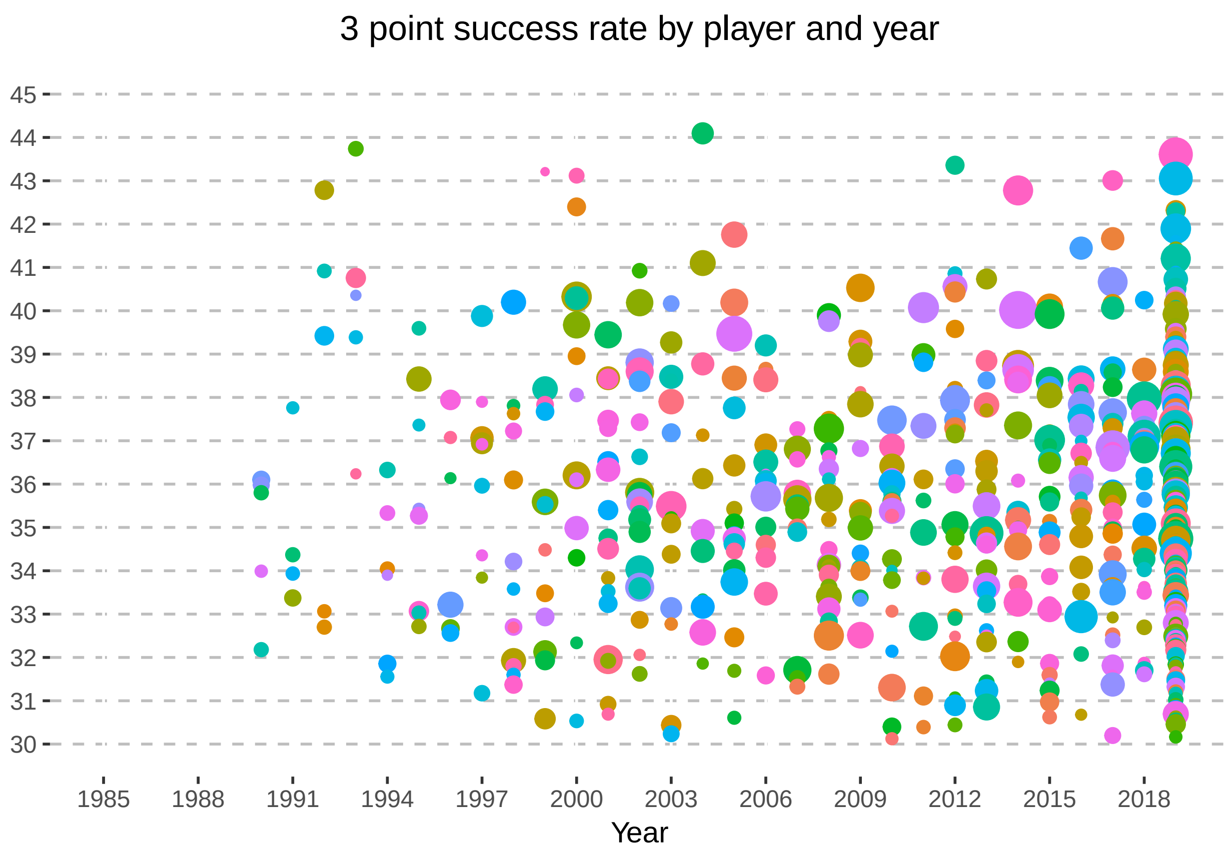



```
xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(30, 45, by=1)

fgplayer1000 <- fgplayer100 %>% filter (fg3m >= 1000)
plotPlayer1000 <- ggplot() +
  geom_pointrange(data=fgplayer1000, aes(x=pctfg3, y=lastYear, ymin=firstYear, ymax=lastYear, colour=Pla
  # geom_point(data=fgplayer100, aes(x=lastYear, y=pctfg3, colour=Player), size=1, show.legend = FALSE)
  # geom_line
  xlab('Year') +
  ylab(NULL) +
```

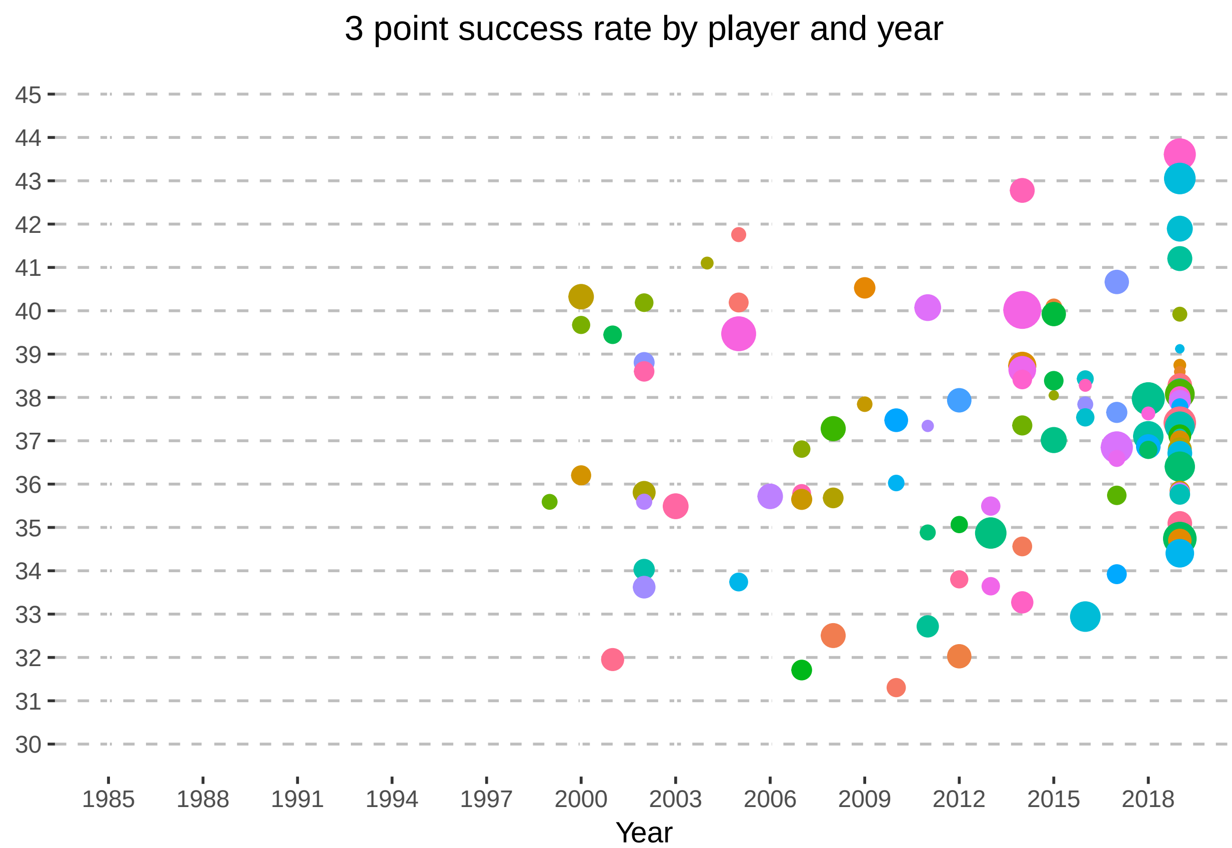
```
  ggtitle('3 point success rate by player and year') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=2
        plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(limits=c(30, 45), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_y_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
plotPlayer1000
```



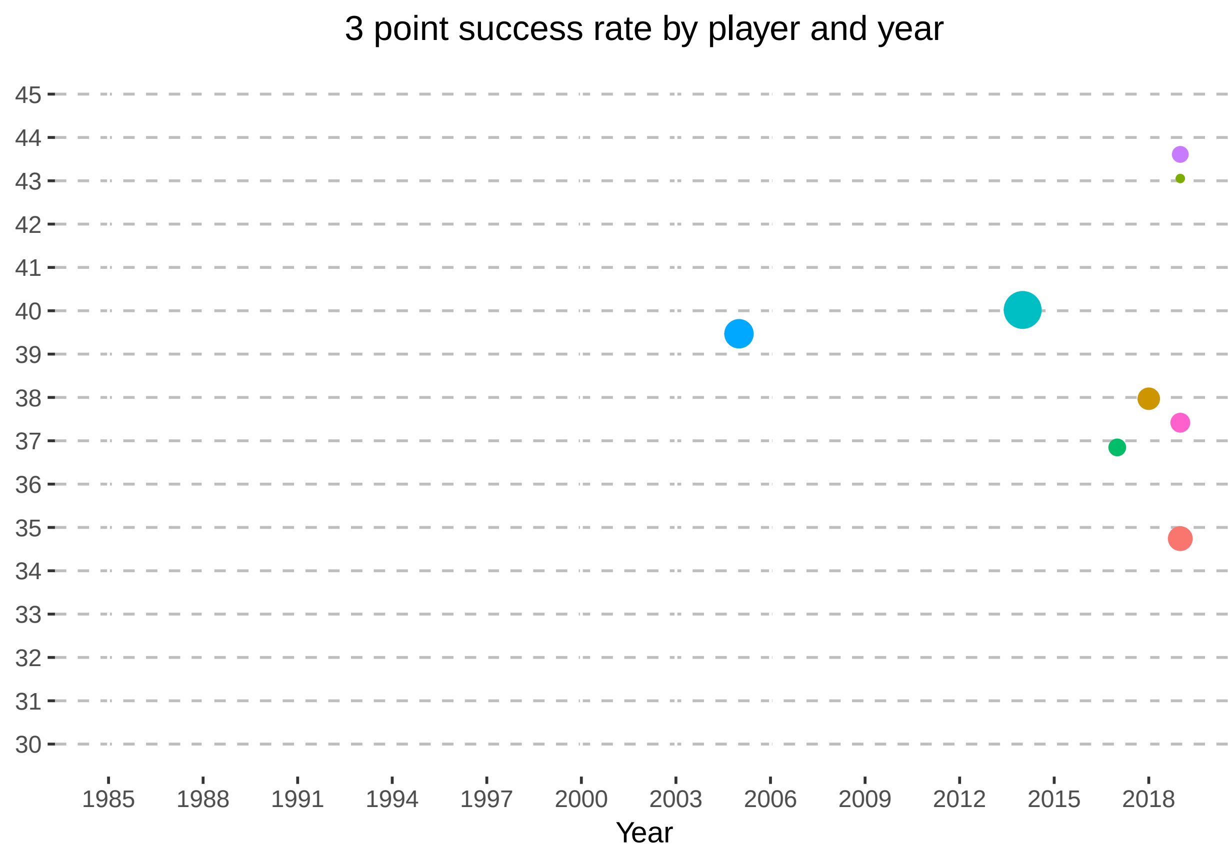

```
xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(30, 45, by=1)

plotPlayer100 <- ggplot() +
  geom_point(data=fgplayer100, aes(x=lastYear, y=pctfg3, size=fg3m+fg3a, colour=Player), show.legend = F
  # geom_point(data=fgplayer100, aes(x=lastYear, y=pctfg3, colour=Player), size=1, show.legend = FALSE)
  # geom_line
  xlab('Year') +
  ylab(NULL) +
  ggtitle('3 point success rate by player and year') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=2
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(30, 45), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
plotPlayer100
```




```
plotPlayer1000 <- ggplot() +
  geom_point(data=fgplayer1000, aes(x=lastYear, y=pctfg3, size=fg3m+fg3a, colour=Player), show.legend =
  # geom_point(data=fgplayer100, aes(x=lastYear, y=pctfg3, colour=Player), size=1, show.legend = FALSE)
  # geom_line
  xlab('Year') +
  ylab(NULL) +
  ggtitle('3 point success rate by player and year') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=2
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(30, 45), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
plotPlayer1000
```




```
plotPlayer2000 <- ggplot() +
  geom_point(data=fgplayer2000, aes(x=lastYear, y=pctfg3, size=fg3m+fg3a, colour=Player), show.legend =
  # geom_point(data=fgplayer100, aes(x=lastYear, y=pctfg3, colour=Player), size=1, show.legend = FALSE)
  # geom_line
  xlab('Year') +
  ylab(NULL) +
  ggtitle('3 point success rate by player and year') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(30, 45), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
plotPlayer2000
```



Above graph shows more players are trying 3 point shots than before. even though the average success rate is similar.

Q4. Players who are good at 3-pointers are also good at 2-pointers or free throws?

By regression.

Players who are good at free throws tend to be good at 3-pointers. However, 2-point field goal success rate is not related with 3-point field goal success rate!!! Why?

```
linearModel <- lm(pctfg3 ~ pctfg2, data=fgplayer100)
summary(linearModel)

Call:
lm(formula = pctfg3 ~ pctfg2, data = fgplayer100)

Residuals:
    Min      1Q  Median      3Q     Max
-13.480  -2.088   0.208   2.228  10.128

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   33.696      1.752   19.23   <2e-16 ***
pctfg2         0.033      0.040    0.82     0.41
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.41 on 747 degrees of freedom
```

```
Multiple R-squared:  0.000907,  Adjusted R-squared:  -0.000431
F-statistic: 0.678 on 1 and 747 DF,  p-value: 0.411

linearModel2 <- lm(fg3m ~ fgm, data=fgplayer100)
summary(linearModel2)

Call:
lm(formula = fg3m ~ fgm, data = fgplayer100)

Residuals:
    Min      1Q  Median      3Q     Max
-1507.3  -152.4   -45.6   153.2  1580.2

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.84e+02   1.96e+01    9.41   <2e-16 ***
fgm         1.43e-01   6.18e-03   23.08   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 338 on 747 degrees of freedom
Multiple R-squared:  0.416, Adjusted R-squared:  0.415
F-statistic:  533 on 1 and 747 DF,  p-value: <2e-16

linearModel3 <- lm(fg3a ~ fga, data=fgplayer100)
summary(linearModel3)

Call:
lm(formula = fg3a ~ fga, data = fgplayer100)

Residuals:
   Min     1Q Median     3Q    Max
 -3928   -345    -84    383   3299

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 4.04e+02   4.80e+01    8.42   <2e-16 ***
fga         1.97e-01   6.87e-03   28.61   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 811 on 747 degrees of freedom
Multiple R-squared:  0.523, Adjusted R-squared:  0.522
F-statistic:  819 on 1 and 747 DF,  p-value: <2e-16

linearModel4 <- lm(fg3a ~ fga + fta, data=fgplayer100)
summary(linearModel4)

Call:
lm(formula = fg3a ~ fga + fta, data = fgplayer100)

Residuals:
   Min     1Q Median     3Q    Max
```

```
 -4161   -287    -47    324   3796

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 275.9537    47.4035    5.82  8.7e-09 ***
fga           0.3470     0.0172   20.23  < 2e-16 ***
fta          -0.4553     0.0481   -9.47  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 767 on 746 degrees of freedom
Multiple R-squared:  0.574, Adjusted R-squared:  0.573
F-statistic:  503 on 2 and 746 DF,  p-value: <2e-16

linearModel5 <- lm(pctfg3 ~ pctft, data=fgplayer100)
summary(linearModel5)

Call:
lm(formula = pctfg3 ~ pctft, data = fgplayer100)

Residuals:
    Min      1Q  Median      3Q     Max
-13.945  -1.946   0.194   2.013   8.631

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  18.2487     1.4223    12.8   <2e-16 ***
pctft         0.2160     0.0181    11.9   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.12 on 747 degrees of freedom
Multiple R-squared:  0.16,  Adjusted R-squared:  0.158
F-statistic:  142 on 1 and 747 DF,  p-value: <2e-16

linearModel6 <- lm(pctfg2 ~ pctft, data=fgplayer100)
summary(linearModel6)

Call:
lm(formula = pctfg2 ~ pctft, data = fgplayer100)

Residuals:
   Min     1Q Median     3Q    Max
-9.428 -2.257 -0.205  1.941 13.732

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  41.9230     1.4155   29.62   <2e-16 ***
pctft         0.0219     0.0180    1.21     0.23
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.11 on 747 degrees of freedom
```

```
Multiple R-squared:  0.00197,   Adjusted R-squared:  0.000631
F-statistic: 1.47 on 1 and 747 DF,  p-value: 0.225

linearModel7 <- lm(pctfg3 ~ pctfg2 + pctft, data=fgplayer100)
summary(linearModel7)

Call:
lm(formula = pctfg3 ~ pctfg2 + pctft, data = fgplayer100)

Residuals:
    Min      1Q  Median      3Q     Max
-13.977  -1.972   0.162   2.021   8.663

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  17.6788     2.0985    8.42   <2e-16 ***
pctfg2        0.0136     0.0368    0.37     0.71
pctft         0.2157     0.0182   11.88   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.13 on 746 degrees of freedom
Multiple R-squared:  0.16,  Adjusted R-squared:  0.157
F-statistic: 70.9 on 2 and 746 DF,  p-value: <2e-16
```

When we look at all the players, 2-pointers and 3-pointers are reverse-related. Maybe because of dunk shots?

```
linearModel7 <- lm(pctfg3 ~ pctfg2 + pctft, data=fgplayer)
summary(linearModel7)

Call:
lm(formula = pctfg3 ~ pctfg2 + pctft, data = fgplayer)

Residuals:
   Min     1Q Median     3Q    Max
-36.58  -8.75   3.58   8.30  84.94

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.6531     2.5237    1.45     0.15
pctfg2       -0.0441     0.0415   -1.06     0.29
pctft         0.3293     0.0237   13.89   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.4 on 2324 degrees of freedom
  (398 observations deleted due to missingness)
Multiple R-squared:  0.0774,   Adjusted R-squared:  0.0766
F-statistic: 97.5 on 2 and 2324 DF,  p-value: <2e-16
```

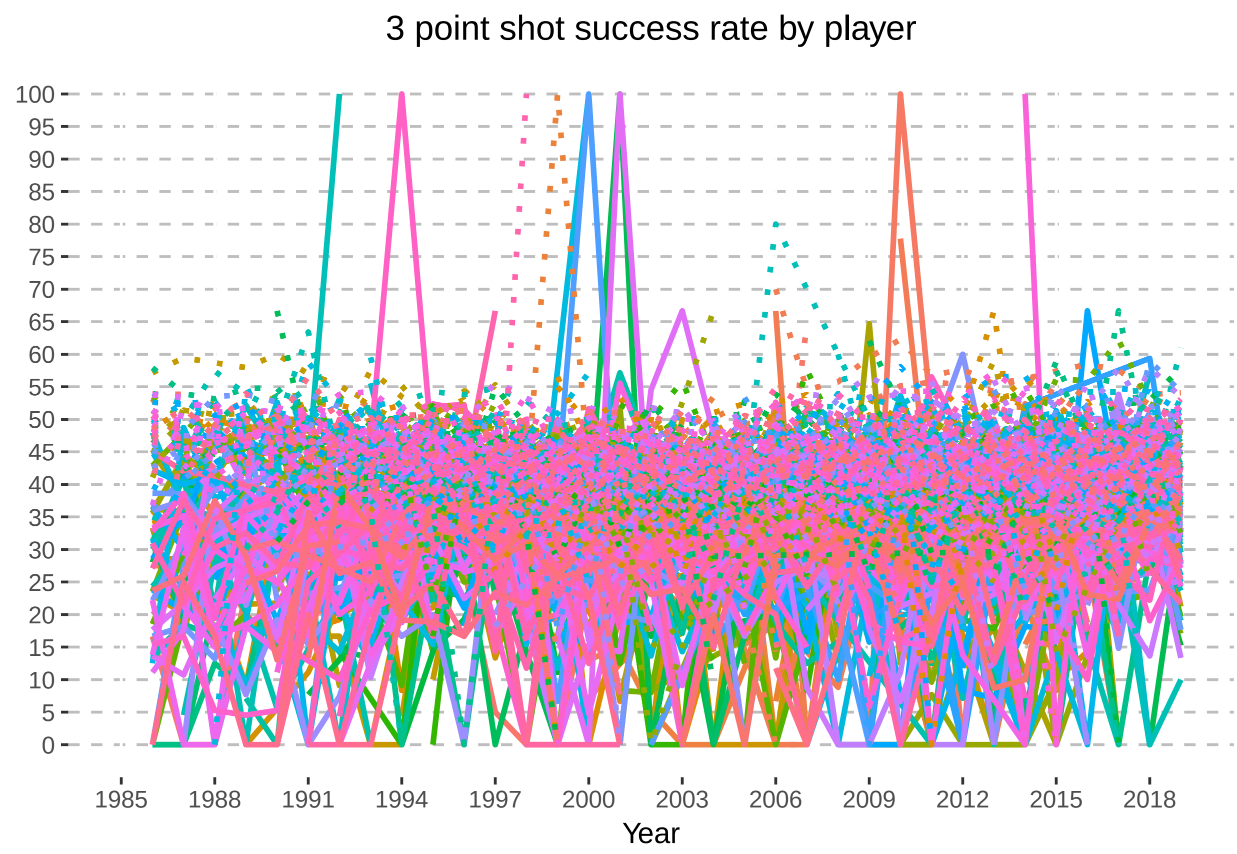
Best players (more than 1,000 career 3-point field goals) are good at 2-pointers as well!!!

```r
linearModel7 <- lm(pctfg3 ~ pctfg2 + pctft, data=fgplayer1000)
summary(linearModel7)

Call:
lm(formula = pctfg3 ~ pctfg2 + pctft, data = fgplayer1000)

Residuals:
   Min     1Q Median     3Q    Max 
-5.191 -1.085  0.106  1.255  4.613 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   3.7572     4.0565    0.93     0.36    
pctfg2        0.3450     0.0843    4.09  8.4e-05 ***
pctft         0.2264     0.0344    6.58  2.0e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.92 on 104 degrees of freedom
Multiple R-squared:  0.426, Adjusted R-squared:  0.415 
F-statistic: 38.6 on 2 and 104 DF,  p-value: 2.91e-13

linearModel8 <- lm(pctfg3 ~ pctfg2 + pctft, data=fgplayer2000)
summary(linearModel8)

Call:
lm(formula = pctfg3 ~ pctfg2 + pctft, data = fgplayer2000)

Residuals:
     1      2      3      4      5      6      7      8 
 0.750  3.634 -0.475 -2.360 -0.449  0.901 -0.579 -1.424 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -21.540     20.147   -1.07     0.33
pctfg2         0.799      0.442    1.81     0.13
pctft          0.290      0.231    1.26     0.26

Residual standard error: 2.14 on 5 degrees of freedom
Multiple R-squared:  0.648, Adjusted R-squared:  0.507 
F-statistic:  4.6 on 2 and 5 DF,  p-value: 0.0737
```

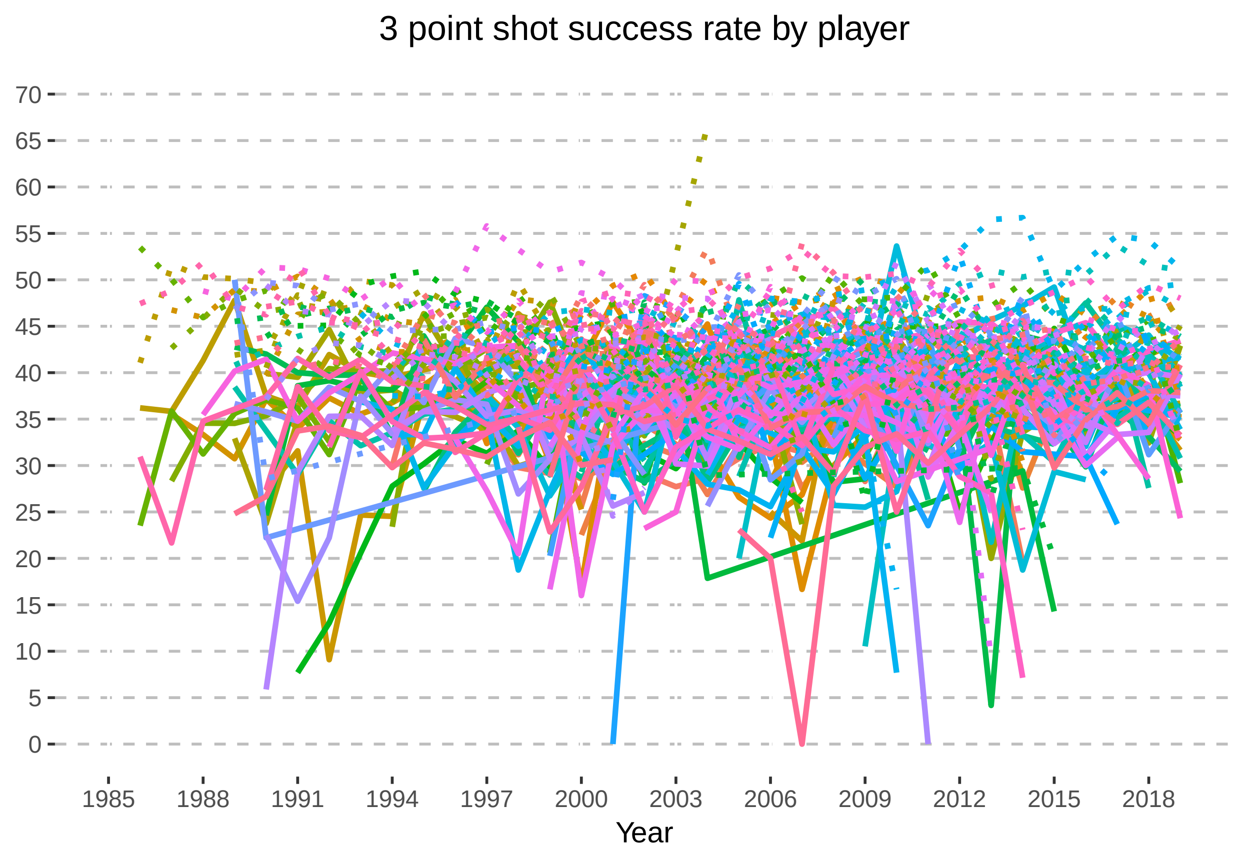
-. Are there any relationship between players' ages and 3-pointers? Both total and average.

```r
fgyearplayer100 <- fgyearplayer %>% filter(Player %in% fgplayer100$Player)
fgyearplayer1000 <- fgyearplayer100 %>% filter(Player %in% fgplayer1000$Player)
fgyearplayer2000 <- fgyearplayer1000 %>% filter(Player %in% fgplayer2000$Player)

xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(0, 100, by=5)

plotYearPlayer100 <- ggplot() +
  geom_line(data=fgyearplayer100, aes(x=Year, y=pctfg3, colour=Player), size=1, show.legend = FALSE) +
```

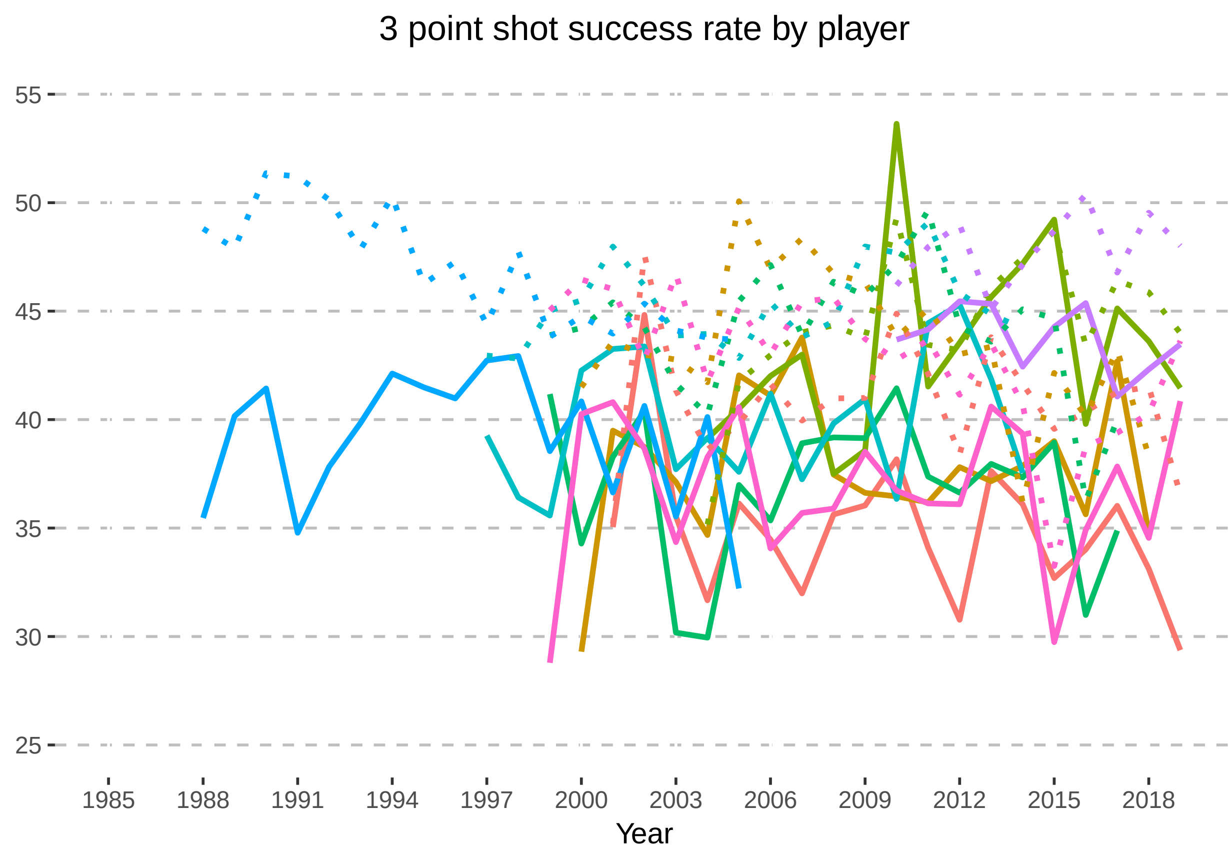
```
  geom_line(data=fgyearplayer100, aes(x=Year, y=pctfg2, colour=Player), size=1, linetype="dotted", show
  xlab('Year') +
  ylab(NULL) +
  ggtitle('3 point shot success rate by player') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(0, 100), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
plotYearPlayer100
```




```
xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(0, 70, by=5)

plotYearPlayer1000 <- ggplot() +
  geom_line(data=fgyearplayer1000, aes(x=Year, y=pctfg3, colour=Player), size=1, show.legend = FALSE) +
  geom_line(data=fgyearplayer1000, aes(x=Year, y=pctfg2, colour=Player), size=1, linetype="dotted", show
  xlab('Year') +
  ylab(NULL) +
  ggtitle('3 point shot success rate by player') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(0, 70), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
plotYearPlayer1000
```




```
xaxisbreaks <- seq(1985, 2019, by=3)
yaxisbreaks <- seq(25, 55, by=5)

plotYearPlayer2000 <- ggplot() +
  geom_line(data=fgyearplayer2000, aes(x=Year, y=pctfg3, colour=Player), size=1, show.legend = FALSE) +
  geom_line(data=fgyearplayer2000, aes(x=Year, y=pctfg2, colour=Player), size=1, linetype="dotted", sho
  xlab('Year') +
  ylab(NULL) +
  ggtitle('3 point shot success rate by player') +
  theme(panel.background=element_rect(fill=NA), panel.grid.major.y=element_line(color="grey", linetype=2
        plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits=c(25, 55), breaks=yaxisbreaks, labels=yaxisbreaks) +
  scale_x_continuous(limits=c(1985,2019), breaks=xaxisbreaks)
plotYearPlayer2000
```



Let's regress.

```
fgyearplayerjoined <- left_join(fgyearplayer, fgplayer, by=c("Player" = "Player"))
fgyearplayerjoined$career = fgyearplayerjoined$Year - fgyearplayerjoined$firstYear + 1

fgyearplayerjoined100 <- fgyearplayerjoined %>% filter(Player %in% fgplayer100$Player)
fgyearplayerjoined1000 <- fgyearplayerjoined100 %>% filter(Player %in% fgplayer1000$Player)
fgyearplayerjoined2000 <- fgyearplayerjoined1000 %>% filter(Player %in% fgplayer2000$Player)

linearModel <- lm(pctfg3.x ~ career, data=fgyearplayerjoined2000)
summary(linearModel)

Call:
lm(formula = pctfg3.x ~ career, data = fgyearplayerjoined2000)

Residuals:
    Min      1Q  Median      3Q     Max
-10.674  -2.637  -0.219   2.729  14.771

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  39.5611     0.7195   54.98   <2e-16 ***
career       -0.0994     0.0656   -1.51     0.13
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 4.21 on 138 degrees of freedom
Multiple R-squared:  0.0163,    Adjusted R-squared:  0.00921
F-statistic: 2.29 on 1 and 138 DF,  p-value: 0.132
linearModel2 <- lm(pctfg3.x ~ career, data=fgyearplayerjoined1000)
summary(linearModel2)

Call:
lm(formula = pctfg3.x ~ career, data = fgyearplayerjoined1000)

Residuals:
   Min     1Q Median     3Q    Max
-36.23  -2.39   0.63   3.45  17.70

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  35.4285     0.2807  126.19   <2e-16 ***
career        0.0730     0.0306    2.38    0.017 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.57 on 1466 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared:  0.00386,   Adjusted R-squared:  0.00318
F-statistic: 5.68 on 1 and 1466 DF,  p-value: 0.0173
linearModel3 <- lm(pctfg3.x ~ career, data=fgyearplayerjoined100)
summary(linearModel3)

Call:
lm(formula = pctfg3.x ~ career, data = fgyearplayerjoined100)

Residuals:
   Min     1Q Median     3Q    Max
-35.61  -3.08   1.55   5.39  68.11

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   31.702      0.207  152.77  < 2e-16 ***
career         0.186      0.028    6.63  3.6e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.43 on 6858 degrees of freedom
  (52 observations deleted due to missingness)
Multiple R-squared:  0.00637,   Adjusted R-squared:  0.00622
F-statistic: 43.9 on 1 and 6858 DF,  p-value: 3.63e-11
linearModel4 <- lm(pctfg3.x ~ career, data=fgyearplayerjoined)
summary(linearModel4)

Call:
lm(formula = pctfg3.x ~ career, data = fgyearplayerjoined)

Residuals:
   Min     1Q Median     3Q    Max
```

```
-34.83 -10.29   4.47  10.36  75.53

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  24.0595     0.2520    95.5   <2e-16 ***
career        0.4144     0.0378    11.0   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 16.8 on 12412 degrees of freedom
  (2300 observations deleted due to missingness)
Multiple R-squared:  0.00959,   Adjusted R-squared:  0.00951
F-statistic:  120 on 1 and 12412 DF,  p-value: <2e-16
```

Really good players are not related with ages/career. Average players' success rate is increased by 0.4% in one year. Not bad…?

- Players with high salaries are good at 3-pointers?

2018-2019 season data only

```
nbaInsiderSalaries <- nba_insider_salaries(assume_player_opt_out = T, assume_team_doesnt_exercise = T, r
You got salary data for the Atlanta Hawks
You got salary data for the Boston Celtics
You got salary data for the Brooklyn Nets
You got salary data for the Charlotte Hornets
You got salary data for the Chicago Bulls
You got salary data for the Cleveland Cavaliers
You got salary data for the Dallas Mavericks
You got salary data for the Denver Nuggets
You got salary data for the Detroit Pistons
You got salary data for the Golden State Warriors
You got salary data for the Houston Rockets
You got salary data for the Indiana Pacers
You got salary data for the Los Angeles Clippers
You got salary data for the Los Angeles Lakers
You got salary data for the Memphis Grizzlies
You got salary data for the Miami Heat
You got salary data for the Milwaukee Bucks
You got salary data for the Minnesota Timberwolves
You got salary data for the New Orleans Pelicans
You got salary data for the New York Knicks
You got salary data for the Oklahoma City Thunder
You got salary data for the Orlando Magic
You got salary data for the Philadelphia 76ers
You got salary data for the Phoenix Suns
You got salary data for the Portland Trail Blazers
You got salary data for the Sacramento Kings
You got salary data for the San Antonio Spurs
You got salary data for the Toronto Raptors
You got salary data for the Utah Jazz
You got salary data for the Washington Wizards
```

```r
fgplayersalary <- left_join(fgplayer, nbaInsiderSalaries, by=c("Player"="namePlayer"))

fgplayersalary2 <- na.omit(fgplayersalary)
fgplayersalary2$salaryinK = fgplayersalary2$value / 1000
fgplayersalary2$salaryinM = fgplayersalary2$value / 1000000

linearModel <- lm(pctfg3 ~ salaryinM, data=fgplayersalary2)
summary(linearModel)

Call:
lm(formula = pctfg3 ~ salaryinM, data = fgplayersalary2)

Residuals:
   Min     1Q Median     3Q    Max
-32.26  -0.87   3.23   5.80  21.83

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  29.6999     0.4504   65.94   <2e-16 ***
salaryinM     0.0931     0.0343    2.72   0.0067 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.9 on 1069 degrees of freedom
Multiple R-squared:  0.00685,	Adjusted R-squared:  0.00592
F-statistic: 7.38 on 1 and 1069 DF,  p-value: 0.00671
linearModel2 <- lm(fg3m ~ salaryinM, data=fgplayersalary2)
summary(linearModel2)

Call:
lm(formula = fg3m ~ salaryinM, data = fgplayersalary2)

Residuals:
   Min     1Q Median     3Q    Max
-747.6 -140.7  -90.4   75.4 2072.6

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    94.48      14.73    6.42  2.1e-10 ***
salaryinM      23.06       1.12   20.59  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 355 on 1069 degrees of freedom
Multiple R-squared:  0.284,	Adjusted R-squared:  0.283
F-statistic:  424 on 1 and 1069 DF,  p-value: <2e-16
```

When the salary increases by a million dollar, career success rate of 3-point shots increases by 0.09% only. It's difficult to say that 3-pointer success rate is the most important factor for one's salary.

- We would like to explore the importance of three point shooters in a given team by measuring the share of the team's total salary over time.

- We want to analyze whether players can drastically improve their three point shooting skills over time or the skill is rather something people are borned with.

There is no dramatic increase in 3-pointer success rate. Maybe if we can check the players' data from NCAA or high school league, there might be different insight. However, based on NBA data, no big changes.

- Show the 3-pointer statistics geographically based on players' hometowns. Maybe this help illustrates the different basketball playing style across different regions, both domestic and international.

```r
playerHometown <- read_csv("PlayerHometown.csv")

fgplayerhometown <- left_join(fgplayer, playerHometown, by=c("Player"="Player"))
fgplayerhometown <- fgplayerhometown %>% filter(not(is.na(State)))
fgplayerhometown <- na.omit(fgplayerhometown)

fgplayerhometownState <- aggregate(fgplayerhometown[, 2:7], list(fgplayerhometown$State), sum)
colnames(fgplayerhometownState)[1] <- "State"
fgplayerhometownState$pctfg3 <- fgplayerhometownState$fg3m / fgplayerhometownState$fg3a * 100
fgplayerhometownState$pctfg2 <- fgplayerhometownState$fgm / fgplayerhometownState$fga * 100
fgplayerhometownState$pctft <- fgplayerhometownState$ftm / fgplayerhometownState$fta * 100

plotState <- ggplot() +
  geom_point(data=fgplayerhometownState, aes(x=State, y=pctfg3, colour=State)) +
  xlab(NULL) +
  ylab(NULL)
plotState
```