

warmup04-Dui-Lee

Dui Lee

2018년 9월 23일

1) Import the data in R

#import using read_csv, and col_types are specified. Column types not specified below had integer values, hence are left untouched.

```
library("readr")
```

```
data = read_csv("nba2018.csv", col_types = cols(player = col_character(),  
                                                team = col_character(),  
                                                birth_date = col_character(),  
                                                country = col_character(),  
                                                experience = col_character(),  
                                                college = col_character(),  
                                                position = col_factor(c("C", "PF", "PG", "SF", "SG")),  
                                                salary = col_double(),  
                                                field_goals_perc = col_double(),  
                                                points3_perc = col_double(),  
                                                points2_perc = col_double(),  
                                                points1_perc = col_double(),  
                                                effective_field_goal_perc = col_double(),  
                                                salary = col_integer()  
                                                ))
```

```
str(data)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame': 477 obs. of 38 variables:
## $ player : chr "Al Horford" "Amir Johnson" "Avery Bradley" "Demetrius Jackson" ...
## $ number : chr "42" "90" "0" "9" ...
## $ team : chr "BOS" "BOS" "BOS" "BOS" ...
## $ position : Factor w/ 5 levels "C","PF","PG",...: 1 2 5 3 4 3 4 5 4 2 ...
## $ height : chr "6-10" "6-9" "6-2" "6-1" ...
## $ weight : int 245 240 180 201 205 185 235 215 225 231 ...
## $ birth_date : chr "June 3, 1986" "May 1, 1987" "November 26, 1990" "September 7, 1994" ...
## $ country : chr "do" "us" "us" "us" ...
## $ experience : chr "9" "11" "6" "R" ...
## $ college : chr "University of Florida" NA "University of Texas at Austin" "University of Notre Dame" ...
## $ salary : num 26540100 12000000 8269663 1450000 1410598 ...
## $ rank : int 4 6 5 15 11 1 3 13 8 10 ...
## $ age : int 30 29 26 22 31 27 26 21 20 29 ...
## $ games : int 68 80 55 5 47 76 72 29 78 78 ...
## $ games_started : int 68 77 55 0 0 76 72 0 20 6 ...
## $ minutes : int 2193 1608 1835 17 538 2569 2335 220 1341 1232 ...
## $ field_goals : int 379 213 359 3 95 682 333 25 192 114 ...
## $ field_goals_atts : int 801 370 775 4 232 1473 720 58 423 262 ...
## $ field_goals_perc : num 0.473 0.576 0.463 0.75 0.409 0.463 0.463 0.431 0.454 0.435 ...
## $ points3 : int 86 27 108 1 39 245 157 12 46 45 ...
## $ points3_atts : int 242 66 277 1 111 646 394 35 135 130 ...
## $ points3_perc : num 0.355 0.409 0.39 1 0.351 0.379 0.398 0.343 0.341 0.346 ...
## $ points2 : int 293 186 251 2 56 437 176 13 146 69 ...
## $ points2_atts : int 559 304 498 3 121 827 326 23 288 132 ...
## $ points2_perc : num 0.524 0.612 0.504 0.667 0.463 0.528 0.54 0.565 0.507 0.523 ...
## $ effective_field_goal_perc : num 0.527 0.612 0.533 0.875 0.494 0.546 0.572 0.534 0.508 0.521 ...
## $ points1 : int 108 67 68 3 33 590 176 6 85 26 ...
## $ points1_atts : int 135 100 93 6 41 649 217 9 124 37 ...
## $ points1_perc : num 0.8 0.67 0.731 0.5 0.805 0.909 0.811 0.667 0.685 0.703 ...
## $ off_rebounds : int 95 118 65 2 16 43 48 6 45 59 ...
## $ def_rebounds : int 370 248 269 2 68 162 367 20 175 213 ...
## $ total_rebounds : int 465 366 334 4 84 205 415 26 220 272 ...
## $ assists : int 337 140 122 3 33 448 155 4 64 71 ...
## $ steals : int 52 51 68 0 9 70 73 10 35 25 ...
## $ blocks : int 86 62 11 0 7 13 23 2 18 17 ...
## $ turnovers : int 115 77 88 0 25 210 80 4 68 39 ...
## $ fouls : int 138 211 141 0 48 167 161 15 142 122 ...
## $ points : int 952 520 894 10 262 2199 999 68 515 299 ...
## - attr(*, "spec")=List of 2
## ..$ cols :List of 38
## .. ..$ player : list()
## .. ..$- attr(*, "class")= chr "collector_character" "collector"
## .. ..$ number : list()
## .. ..$- attr(*, "class")= chr "collector_character" "collector"
## .. ..$ team : list()
## .. ..$- attr(*, "class")= chr "collector_character" "collector"
```

```

## .. ..$ position           :List of 3
## .. ..$ levels             : chr  "C" "PF" "PG" "SF" ...
## .. ..$ ordered            : logi FALSE
## .. ..$ include_na: logi FALSE
## .. ..$- attr(*, "class")= chr  "collector_factor" "collector"
## .. ..$ height              : list()
## .. ..$- attr(*, "class")= chr  "collector_character" "collector"
## .. ..$ weight              : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ birth_date          : list()
## .. ..$- attr(*, "class")= chr  "collector_character" "collector"
## .. ..$ country             : list()
## .. ..$- attr(*, "class")= chr  "collector_character" "collector"
## .. ..$ experience          : list()
## .. ..$- attr(*, "class")= chr  "collector_character" "collector"
## .. ..$ college             : list()
## .. ..$- attr(*, "class")= chr  "collector_character" "collector"
## .. ..$ salary              : list()
## .. ..$- attr(*, "class")= chr  "collector_double" "collector"
## .. ..$ rank                : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ age                 : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ games               : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ games_started       : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ minutes             : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ field_goals         : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ field_goals_atts    : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ field_goals_perc    : list()
## .. ..$- attr(*, "class")= chr  "collector_double" "collector"
## .. ..$ points3             : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ points3_atts        : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ points3_perc        : list()
## .. ..$- attr(*, "class")= chr  "collector_double" "collector"
## .. ..$ points2             : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ points2_atts        : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ points2_perc        : list()
## .. ..$- attr(*, "class")= chr  "collector_double" "collector"
## .. ..$ effective_field_goal_perc: list()
## .. ..$- attr(*, "class")= chr  "collector_double" "collector"
## .. ..$ points1             : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ points1_atts        : list()
## .. ..$- attr(*, "class")= chr  "collector_integer" "collector"
## .. ..$ points1_perc        : list()

```

```
## .. ..- attr(*, "class")= chr "collector_double" "collector"
## .. ..$ off_rebounds : list()
## .. ..- attr(*, "class")= chr "collector_integer" "collector"
## .. ..$ def_rebounds : list()
## .. ..- attr(*, "class")= chr "collector_integer" "collector"
## .. ..$ total_rebounds : list()
## .. ..- attr(*, "class")= chr "collector_integer" "collector"
## .. ..$ assists : list()
## .. ..- attr(*, "class")= chr "collector_integer" "collector"
## .. ..$ steals : list()
## .. ..- attr(*, "class")= chr "collector_integer" "collector"
## .. ..$ blocks : list()
## .. ..- attr(*, "class")= chr "collector_integer" "collector"
## .. ..$ turnovers : list()
## .. ..- attr(*, "class")= chr "collector_integer" "collector"
## .. ..$ fouls : list()
## .. ..- attr(*, "class")= chr "collector_integer" "collector"
## .. ..$ points : list()
## .. ..- attr(*, "class")= chr "collector_integer" "collector"
## ..$ default: list()
## .. ..- attr(*, "class")= chr "collector_guess" "collector"
## ..- attr(*, "class")= chr "col_spec"
```

2)Right after importing the data

```
#experience "R" converted to 0 and column type converted into integers.
data$experience[data$experience == "R"] = 0
data$experience = as.integer(data$experience)
summary(data$experience)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.000   1.000   4.000   4.662   7.000  18.000
```

```
#Salary converted into millions and display summary
data$salary = data$salary / 1000000
summary(data$salary)
```

```
##      Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.005145 1.050961 3.000000 5.804697 8.269663 30.963450
```

```
#Relabel positions factors and display frequencies of the factors
levels(data$position) <- c("center", "power_fwd", "point_guard", "small_fwd", "shoot_guard")
table(data$position)
```

```
##
##      center  power_fwd point_guard  small_fwd shoot_guard
##          97          98          96          84         102
```

3) A bit of subscripting(i.e. indexing, slicing, subsetting)

```
#How many players went to UCLA ("University of California, Los Angeles")?  
length(data$college[data$college == "University of California, Los Angeles" & is.na(data$college) == FALSE])
```

```
## [1] 14
```

```
#How many players went to Cal ("University of California, Berkeley")?  
length(data$college[data$college == "University of California, Berkeley" & is.na(data$college) == FALSE])
```

```
## [1] 0
```

```
#What's the largest weight value?  
max(data$weight)
```

```
## [1] 290
```

```
#Who are the players with the largest weight value?  
data$player[data$weight == max(data$weight)]
```

```
## [1] "Boban Marjanovic"
```

```
#What's the overall average weight?  
mean(data$weight)
```

```
## [1] 219.9119
```

```
#What is the median salary of all players?  
median(data$salary)
```

```
## [1] 3
```

```
#What is the median salary of the players with 10 years of experience or more?  
median(data$salary[data$experience >= 10])
```

```
## [1] 4.8375
```

```
#What is the median salary of Shooting Guards (SG) and Point Guards (PG)?
median(data$salary[data$position == "shoot_guard" | data$position == "point_guard"])
```

```
## [1] 2.789697
```

```
#What is the median salary of Power Forwards (PF), 30 years or older, weighing 240 pounds or more?
```

```
median(data$salary[data$position == "power_fwd" & data$age >= 30 & data$weight >= 240])
```

```
## [1] 8
```

```
#Create a data frame gsw with the player name, position, height, weight, and age of Golden State Warriors (GSW). Display this data frame.
```

```
gsw = data[c(data$team == "GSW") , c(1,4,5,6,13)]
```

```
gsw
```

```
## # A tibble: 16 x 5
##   player           position height weight  age
##   <chr>           <fct>    <chr>   <int> <int>
## 1 Anderson Varejao center    6-10     273   34
## 2 Andre Iguodala  small_fwd 6-6      215   33
## 3 Damian Jones    center    7-0      245   21
## 4 David West      center    6-9      250   36
## 5 Draymond Green  power_fwd 6-7      230   26
## 6 Ian Clark       shoot_guard 6-3      175   25
## 7 James Michael McAdoo power_fwd 6-9      230   24
## 8 JaVale McGee    center    7-0      270   29
## 9 Kevin Durant    power_fwd 6-9      240   28
## 10 Kevon Looney    center    6-9      220   20
## 11 Klay Thompson  shoot_guard 6-7      215   26
## 12 Matt Barnes    small_fwd 6-7      226   36
## 13 Patrick McCaw  shoot_guard 6-7      185   21
## 14 Shaun Livingston point_guard 6-7      192   31
## 15 Stephen Curry  point_guard 6-3      190   28
## 16 Zaza Pachulia  center    6-11     270   32
```

4) Performance of players

```
#Add the following variables
##missed_field_goals (missed field goals)
data["missed_field_goals"] <- NA
data$missed_field_goals <- data$field_goals_atts - data$field_goals
##missed_free_throws (missed free throws)
data["missed_free_throws"] <- NA
data$missed_free_throws <- data$points1_atts - data$points1
##rebounds (total rebounds: offensive and defensive)
data["rebounds"] <- NA
data$rebounds <- data$off_rebounds + data$def_rebounds
##mins_game (minutes per game; NOT to be used when calculating EFF)
data["mins_game"] <- NA
data$mins_game <- data$minutes / data$games
```

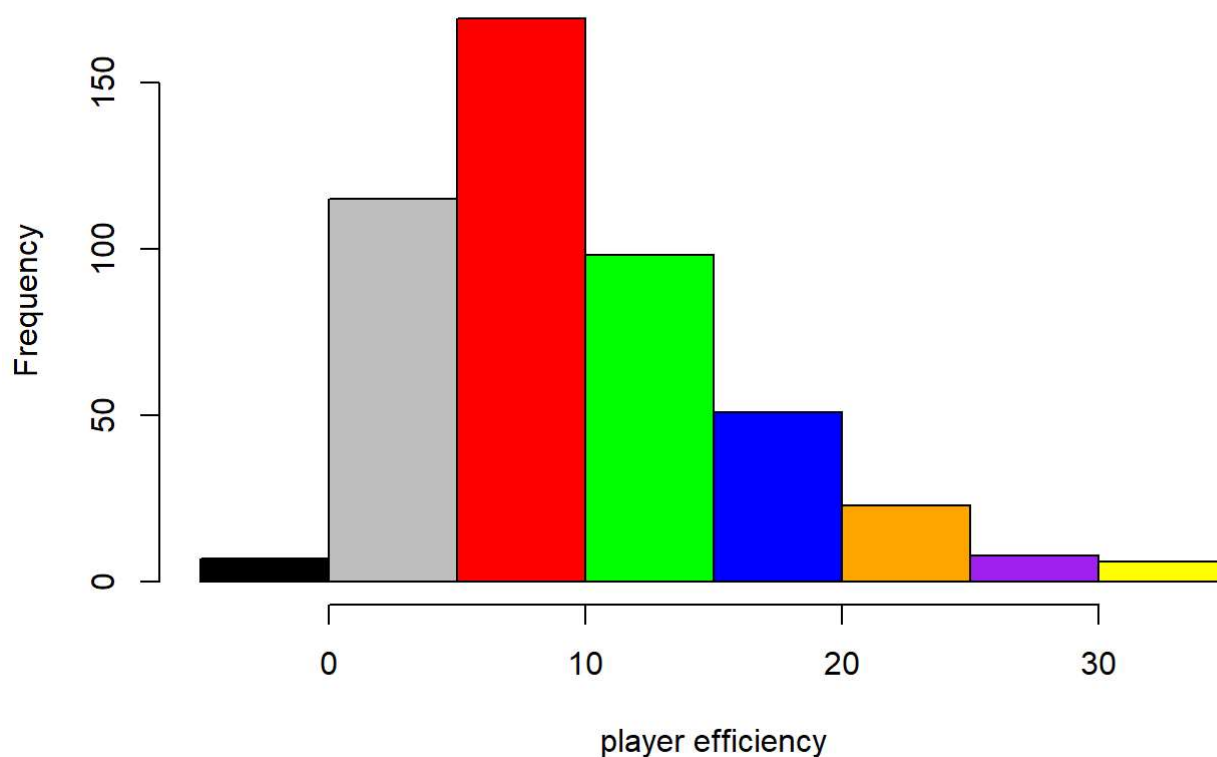
```
#add column efficiency
eff = (data$points + data$total_rebounds + data$assists + data$steals + data$blocks - data$missed_field_goals - data$missed_free_throws - data$turnovers) / data$games
data["efficiency"] <- NA
data$efficiency <- eff
```

```
#summary statistics for efficiency
summary(data$efficiency)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -0.6667  5.0000   8.3472   9.5788 12.6066  33.8272
```

```
#histrogram for efficiency along with colors, title, and axis-label
hist(data$efficiency, main="Distribution of Player Efficiency", col = c("black", "gray", "red", "green", "blue", "orange", "purple", "yellow"), xlab="player efficiency")
```

Distribution of Player Efficiency



```
#Player name, team, salary, and efficiency value of the top-10 players by EFF in decreasing order
data[order(data$efficiency, decreasing = TRUE),][c(1:10),][c("player", "team", "salary", "efficiency")]
```

```
## # A tibble: 10 x 4
##   player          team salary efficiency
##   <chr>          <chr>   <dbl>     <dbl>
## 1 Russell Westbrook OKC    26.5      33.8
## 2 James Harden     HOU    26.5      32.4
## 3 Anthony Davis    NOP    22.1      31.1
## 4 LeBron James     CLE    31.0      31.0
## 5 Karl-Anthony Towns MIN     5.96      30.3
## 6 Kevin Durant     GSW    26.5      30.2
## 7 Giannis Antetokounmpo MIL     3.00      28.4
## 8 DeMarcus Cousins  NOP    17.0      27.9
## 9 Jimmy Butler     CHI    17.6      25.6
## 10 Hassan Whiteside MIA    22.1      25.4
```

```
#players with a negative EFF
data[data$efficiency < 0,][, c("player")]
```

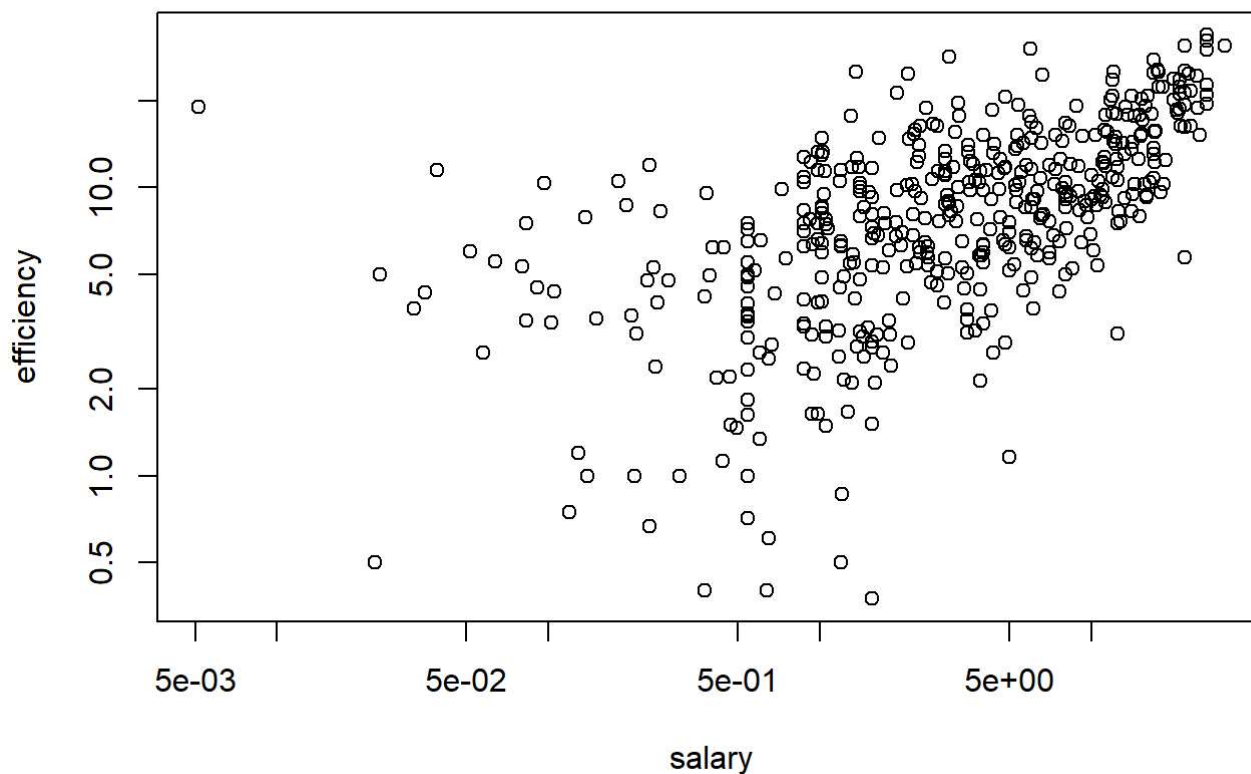


```
## # A tibble: 4 x 1
##   player
##   <chr>
## 1 Gary Neal
## 2 Axel Toupane
## 3 Patricio Garino
## 4 Ben Bentil
```

5) Further Exploration

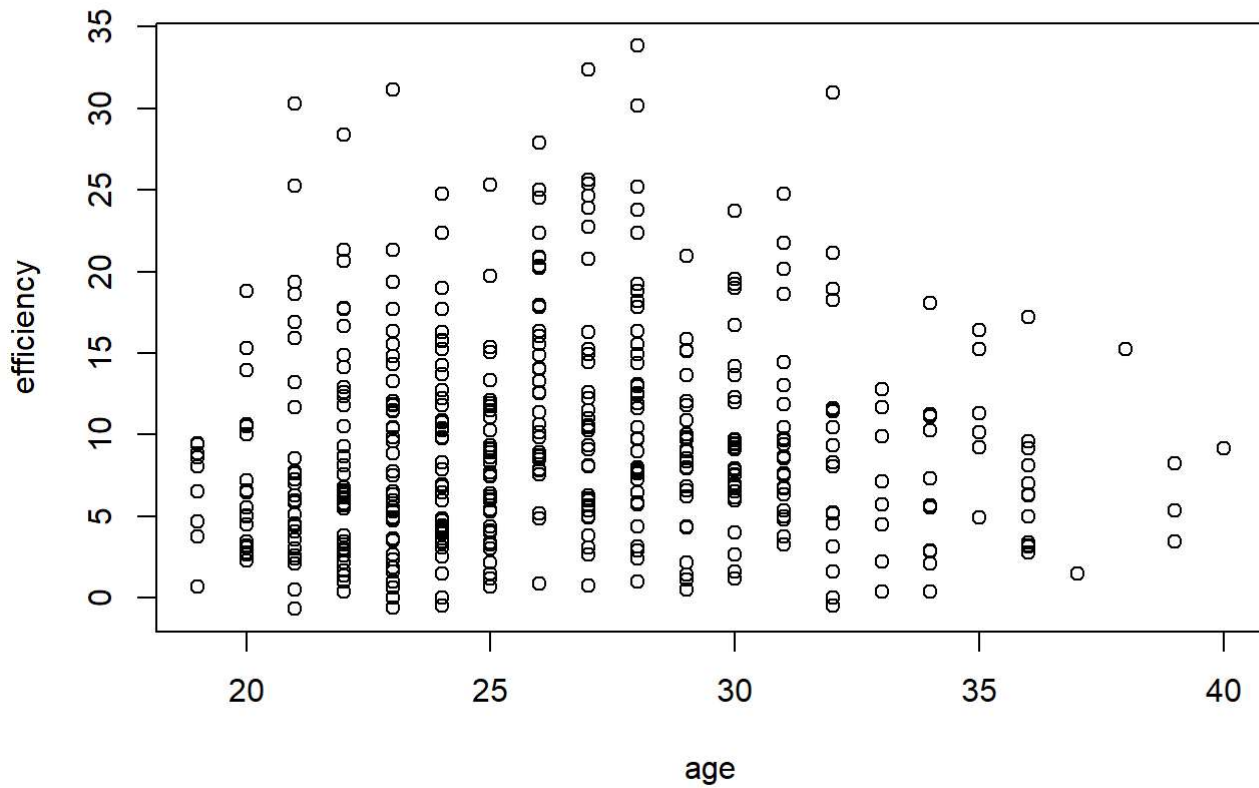
```
#The more efficient a player is, the higher his salary?
plot(efficiency~salary, data = data ,log= "xy")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 7 y values <= 0 omitted
## from logarithmic plot
```



*Looks like there is a correlation between salary and efficiency and it is a positive relation. Hence I can say that higher the salary, higher the efficiency

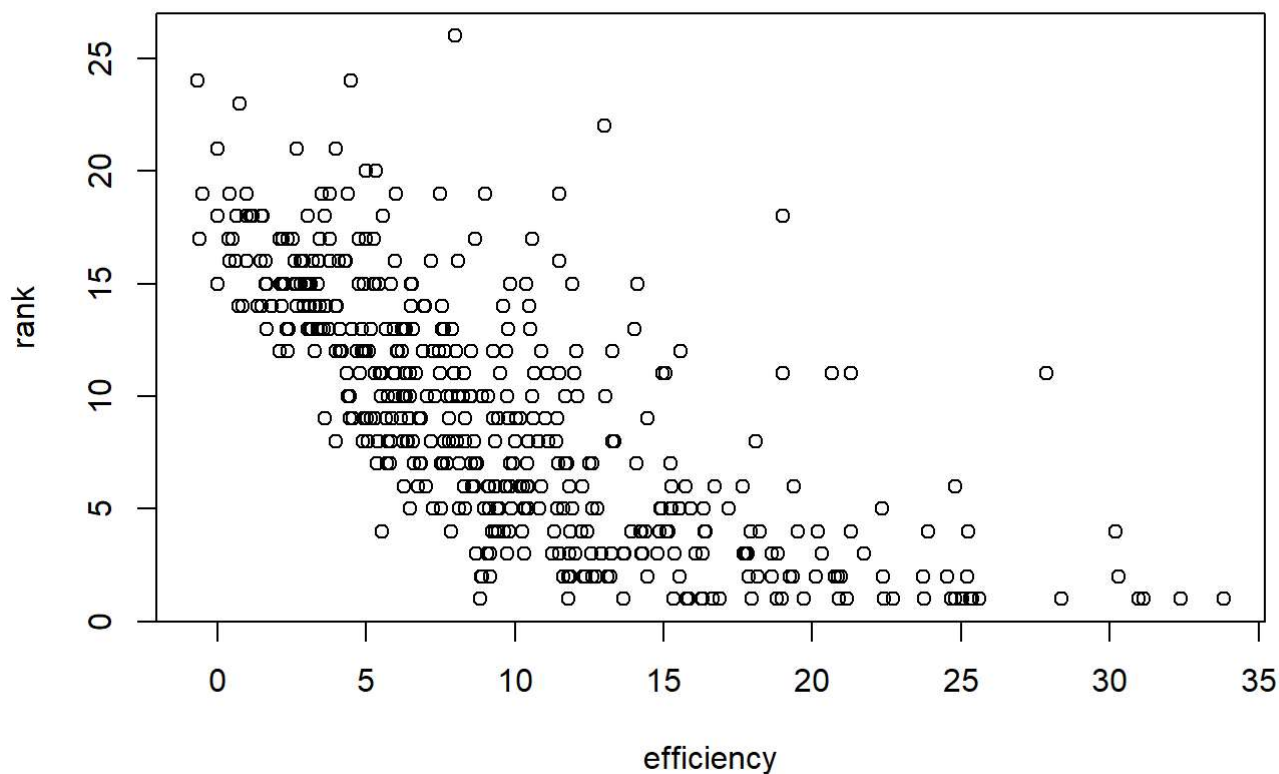
```
#As players get older, do they tend to become more efficient?
plot(efficiency~age, data = data)
```



*There seems to be no correlation between age and efficiency. Hence it is not true that as players get older, they tend to become more efficient

#Does the rank of a player seem to be associated with his efficiency (i.e. the more important the rank, the more efficient)?

```
plot(rank~efficiency, data = data)
```



*It looks like as ranks get closer to 1, the efficiency is more concentrated on the higher side. Hence we can say that the more important the rank, the more efficient

6) Comments and Reflections

*It took me about 4~5 hours to complete this. I think modifying the data frame was the hard part. Plotting the graphs to find evidence for the last part of the lab was relatively on the easier side. To complete this assignment, I did have to consult some external websites to find specific commands. The most consuming part was setting up the data, assigning the data types and changing labels. I didn't really have something that I did not understand, but figuring out how to modify data frames felt a little frustrating at first.