shot_dist_asess

2022-10-01

read csv and load tidyverse

```
shots_data <- read.csv("shots_data.csv")
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Peek at data frame structure

```
head(shots_data)
```

```
##     team     x    y fgmade
## 1 Team A  -5.0 26.4      0
## 2 Team A  -0.8  1.2      1
## 3 Team A -13.9  9.5      1
## 4 Team A  -5.4 26.2      0
## 5 Team A  -4.9 14.5      1
## 6 Team A -10.9 23.9      1
```

```
str(shots_data)
```

```
## 'data.frame':    504 obs. of  4 variables:
##  $ team  : chr  "Team A" "Team A" "Team A" "Team A" ...
##  $ x     : num  -5 -0.8 -13.9 -5.4 -4.9 -10.9 -12 -7.3 -2.8 1.4 ...
##  $ y     : num  26.4 1.2 9.5 26.2 14.5 23.9 23.6 5.7 11 1.6 ...
##  $ fgmade: int  0 1 1 0 1 1 1 0 1 0 ...
```

504 obs, 4 vars. chr, num, num, int.

add new variable to data for distance from basket

```
shots_data <- shots_data%>%mutate(distance = sqrt(x**2 + y**2))
```

intoduce new zone variable

```
shots_w_zones<- shots_data %>%
  mutate(zone = case_when(distance < 23.75 & (x <= 22 | x >= -22) ~ 'two-pointer',
```

```
                            (x >= 22 | x <= -22) & y <= 7.8 ~ 'Corner 3',
                            distance >= 23.75 & y > 7.8 ~ 'NonCorner 3'))
```

```
head(shots_w_zones)
```

```
##     team     x    y fgmade  distance        zone
## 1 Team A  -5.0 26.4      0 26.869313 NonCorner 3
## 2 Team A  -0.8  1.2      1  1.442221 two-pointer
## 3 Team A -13.9  9.5      1 16.836270 two-pointer
## 4 Team A  -5.4 26.2      0 26.750701 NonCorner 3
## 5 Team A  -4.9 14.5      1 15.305555 two-pointer
## 6 Team A -10.9 23.9      1 26.268232 NonCorner 3
```

group by team and zone and agg by count

```
shots_w_zones%>%group_by(team, zone)%>%summarise(attempts = n())
```

```
## `summarise()` has grouped output by 'team'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 3
## # Groups:   team [2]
##   team   zone        attempts
##   <chr>  <chr>          <int>
## 1 Team A Corner 3           3
## 2 Team A NonCorner 3       68
## 3 Team A two-pointer      209
## 4 Team B Corner 3           1
## 5 Team B NonCorner 3       62
## 6 Team B two-pointer      161
```

shot distribution for team A

```
perc_A_shots_twos <- 209 / (209 + 68 + 3)
perc_A_shots_corner <- 3 / (209 + 68 + 3)
perc_A_shots_noncorner <- 68 / (209 + 68 + 3)
```

```
team_a_shot_dist <- data.frame(perc_A_shots_twos, perc_A_shots_corner, perc_A_shots_noncorner)
```

```
names(team_a_shot_dist) <- c('A twos', 'A corner', 'A nonCorner')
```

```
team_a_shot_dist
```

```
##      A twos   A corner A nonCorner
## 1 0.7464286 0.01071429   0.2428571
```

shot distribution for team B

```
perc_B_shots_twos <- 161 / (161+1+62)
perc_B_shots_corner <- 1 / (161+1+62)
perc_B_shots_noncorner <- 62 / (161+1+62)
```

```
team_b_shot_dist <- data.frame(perc_B_shots_twos, perc_B_shots_corner, perc_B_shots_noncorner)
```

```
names(team_b_shot_dist) <- c('B twos', 'B corner', 'B nonCorner')
```

```
team_b_shot_dist
```

```
##    B twos    B corner B nonCorner
## 1 0.71875 0.004464286   0.2767857
```

efg calculation depending on zone, df grouped by team and zone

```
efg_gpby_teamandzone<-shots_w_zones%>%group_by(team, zone)%>%
  summarise(eFG = case_when(zone == 'corner 3' |
                              zone == 'NonCorner 3' ~ (sum(fgmade) + (.5*sum(fgmade)))/length(fgmade),
                            zone == 'two-pointer' ~ sum(fgmade) / length(fgmade)))
```

```
## `summarise()` has grouped output by 'team', 'zone'. You can override using the
## `.groups` argument.
```

tmAefg dist.

```
tmA_efgs <- efg_gpby_teamandzone%>%filter(team == 'Team A')%>%group_by(zone)%>%summarise(efg = mean(eFG)

tmB_efgs <- efg_gpby_teamandzone%>%filter(team == 'Team B')%>%group_by(zone)%>%summarise(efg = mean(eFG)
```

```
tmA_efgs
```

```
## # A tibble: 3 x 2
##   zone           efg
##   <chr>        <dbl>
## 1 Corner 3    NA
## 2 NonCorner 3  0.463
## 3 two-pointer  0.488
```

tmB efg dist.

```
tmB_efgs
```

```
## # A tibble: 3 x 2
##   zone           efg
##   <chr>        <dbl>
## 1 Corner 3    NA
## 2 NonCorner 3  0.508
## 3 two-pointer  0.441
```