EPL Analysis

NAMES

16 September 2021

Intoduction

1.About EPL /Football 2 Why analysing -i.e Betting 3. About the data 4. Variable definitions

Loading the required library

```
library(tidyverse)
```

Loading the dataset

```
url <- "https://raw.githubusercontent.com/Neethithevan/EPL-Data-Analysis/main/Dataset/"

csv_names = c("2000-01.csv","2001-02.csv","2002-03.csv","2003-04.csv","2004-05.csv",
              "2005-06.csv","2006-07.csv","2007-08.csv","2008-09.csv","2009-10.csv",
              "2010-11.csv","2011-12.csv","2012-13.csv","2013-14.csv","2014-15.csv",
              "2015-16.csv","2016-17.csv","2017-18.csv","2018-19.csv","2019-20.csv")

url_csv <- str_c(url , csv_names)

data_all <- url_csv %>%
              lapply(read_csv) %>% bind_rows
```

```
data <- read_csv(str_c(url,csv_names[1]))
```

```
## Rows: 380 Columns: 28

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): Div, Date, HomeTeam, AwayTeam, FTR, HTR, Referee
## dbl (21): FTHG, FTAG, HTHG, HTAG, Attendance, HS, AS, HST, AST, HHW, AHW, HC...

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
df <- data_all %>% select("Date","HomeTeam","AwayTeam","FTHG","FTAG","FTR",
                          "HTHG","HTAG","HTR","HST","AST","HF","AF", "HS", "AS", "Attendance")
head(as_tibble(df))
```

```
## # A tibble: 6 x 16
##   Date   HomeTeam AwayTeam  FTHG  FTAG FTR    HTHG  HTAG HTR     HST   AST    HF
##   <chr>  <chr>    <chr>    <dbl> <dbl> <chr> <dbl> <dbl> <chr> <dbl> <dbl> <dbl>
## 1 19/08~ Charlton Man City     4     0 H         2     0 H        14     4    13
## 2 19/08~ Chelsea  West Ham     4     2 H         1     0 H        10     5    19
## 3 19/08~ Coventry Middles~     1     3 A         1     1 D         3     9    15
## 4 19/08~ Derby    Southam~     2     2 D         1     2 A         4     6    11
## 5 19/08~ Leeds    Everton      2     0 H         2     0 H         8     6    21
## 6 19/08~ Leicest~ Aston V~     0     0 D         0     0 D         4     3    12
## # ... with 4 more variables: AF <dbl>, HS <dbl>, AS <dbl>, Attendance <dbl>
```

Exploring the data set

```
sprintf("The number of rows : %d",dim(df)[1])
```

```
## [1] "The number of rows : 7260"
```

```
sprintf("The number of columns : %d", dim(df)[2])
```

```
## [1] "The number of columns : 16"
```

```
team_names <- df %>% distinct(HomeTeam)
sprintf("Total Number of Teams : %d", dim(team_names)[1])
```

```
## [1] "Total Number of Teams : 44"
```

Analysis 1

Goals Average/ Game through the 20 seasons

```
season_vec <- c("2000-01","2001-02","2002-03","2003-04","2004-05","2005-06",
                "2006-07","2007-08","2008-09","2009-10","2010-11","2011-12",
                "2012-13","2013-14","2014-15","2015-16","2016-17","2017-18",
                "2018-19","2019-20")
season_col <- c()

for(i in 1:20){
  if(i<19){season_col <-  c(season_col,rep(season_vec[i],380))}
  else if( i == 19){season_col <-  c(season_col,rep(season_vec[i],160))}
  else if(i==20){season_col <-  c(season_col,rep(season_vec[i],260))}
}

df$Season <- season_col # Adding it to the date frame df
```

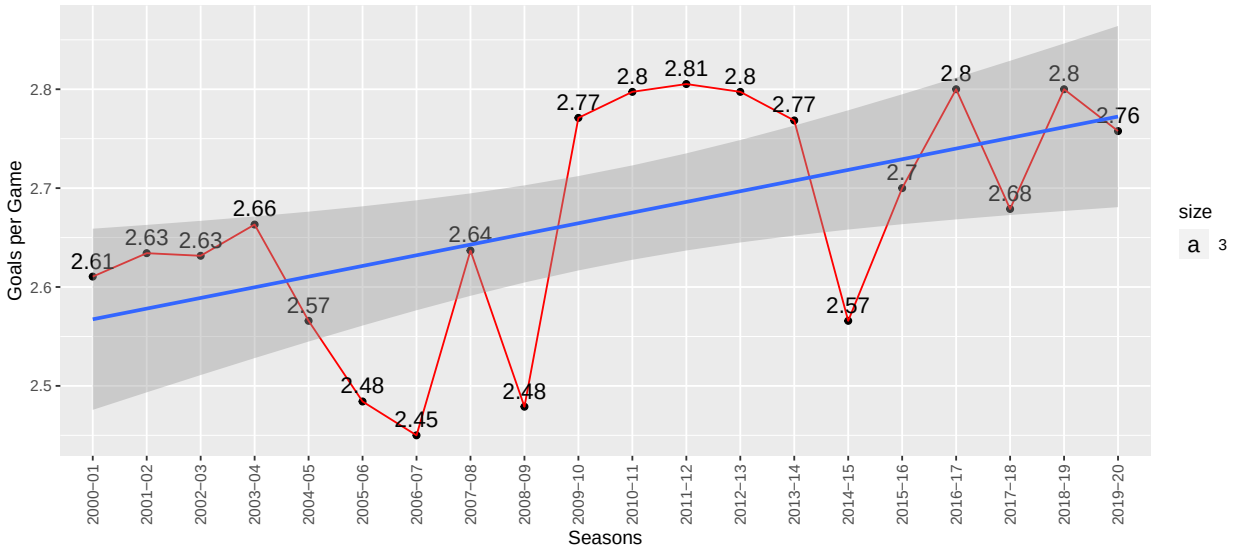
```
# Finding the goal average per match/game and summarizing the result in season wise
df_goal_avg_season <- df %>%mutate(TG = FTHG + FTAG) %>%
  group_by(Season) %>%
  summarise(goal_avg = sum(TG)/n())
head(df_goal_avg_season)
```

```
## # A tibble: 6 x 2
##   Season  goal_avg
##   <chr>      <dbl>
## 1 2000-01     2.61
## 2 2001-02     2.63
## 3 2002-03     2.63
## 4 2003-04     2.66
## 5 2004-05     2.57
## 6 2005-06     2.48
```

```
# Ploting the results in graph
ggplot(df_goal_avg_season, aes(Season ,goal_avg ,group = 2)) +
  geom_point()+
  geom_line(col = "red" ,bg = "blue")+
  geom_text(aes(label=signif(goal_avg ,3), size= 3 , vjust = -0.5)) +
  geom_smooth(method = "lm")+
  ylab("Goals per Game")+
  xlab("Seasons")+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

```
## Warning: Ignoring unknown parameters: fill
```

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



Analysis 1 inference i.e We can see the increase in trend

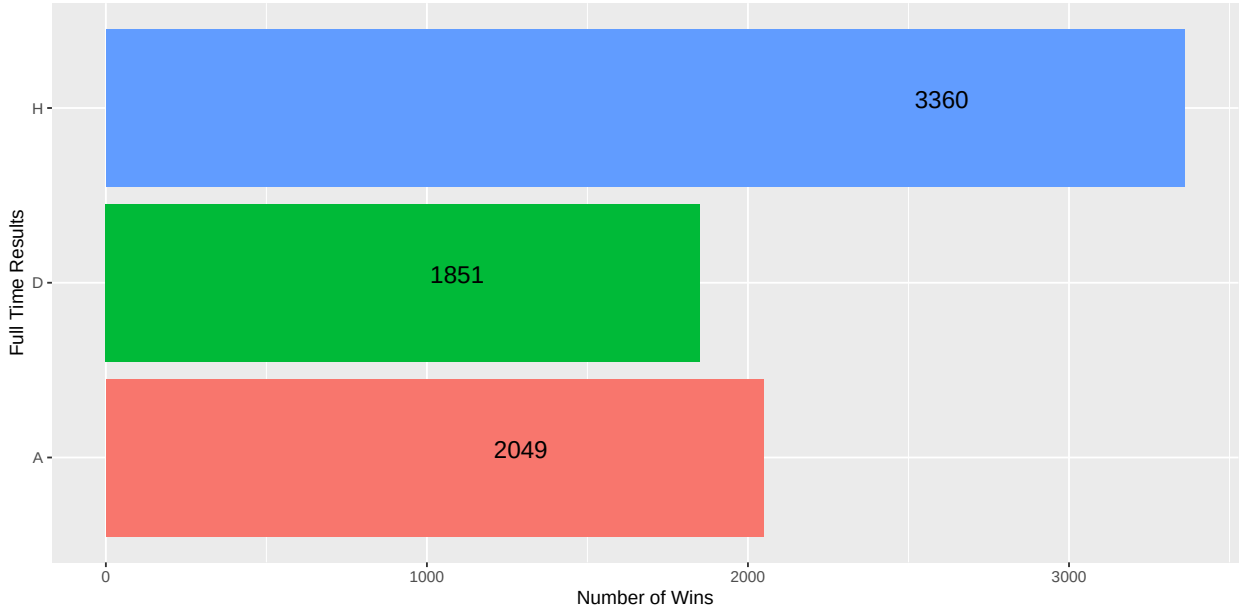
Analysis 2

Home Ground Advantage

```
df %>% group_by(FTR) %>% summarize(number_of_wins = n()) %>% mutate(percent = number_of_wins/sum(number
```

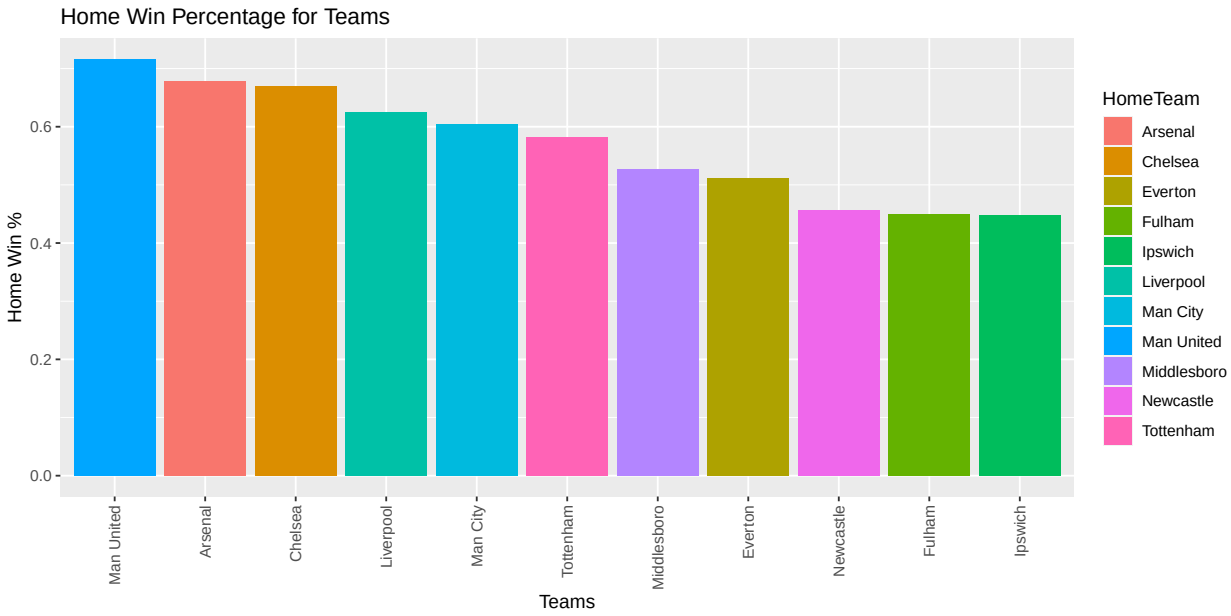
```
## # A tibble: 3 x 3
##   FTR   number_of_wins percent
##   <chr>          <int>   <dbl>
## 1 A               2049   0.282
## 2 D               1851   0.255
## 3 H               3360   0.463
```

```
ggplot(df %>% group_by(FTR) %>% summarize( number_of_wins = n()) ,
       aes(x = FTR , y = number_of_wins,fill = FTR))+
  geom_bar(stat ="identity") + coord_flip() +theme_grey() +
  geom_text(aes(label=number_of_wins), vjust=0 , hjust = 5 ,size = 5)+
  ylab("Number of Wins")+
  xlab("Full Time Results")+
  theme(legend.position = "none")
```

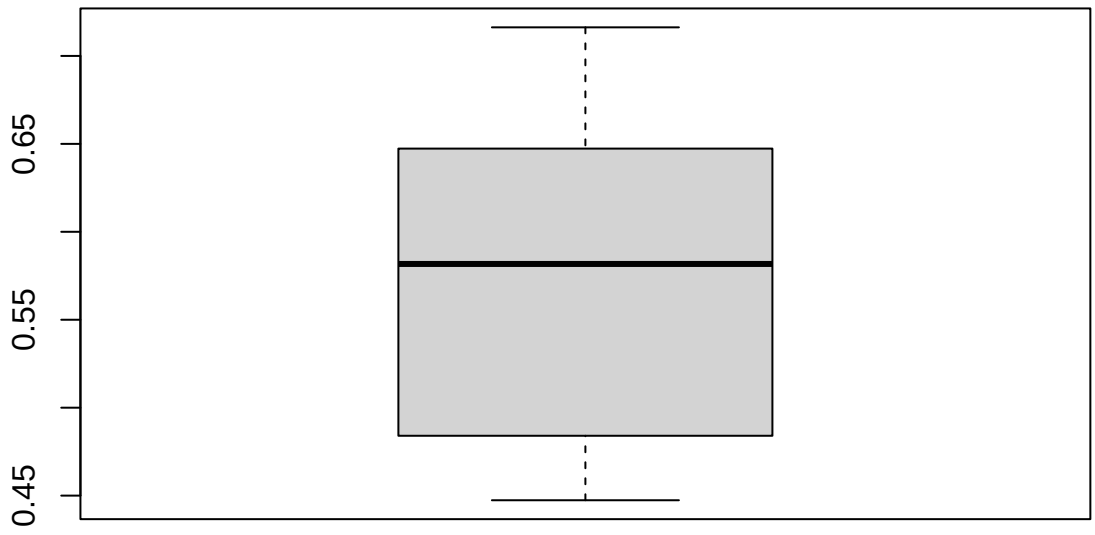



```
## Analysing Home Wins by teams

df_home_ground<-df %>% group_by(HomeTeam,FTR) %>% summarise(wins= n()) %>%
  ungroup() %>% group_by(HomeTeam) %>%  mutate(total_wins = sum(wins)) %>%
  ungroup() %>% mutate(win_percentage= wins/total_wins) %>% filter(FTR == "H") %>%
  arrange(desc(win_percentage)) %>% head(11)

ggplot(df_home_ground , aes(x = reorder(HomeTeam ,-win_percentage), y = win_percentage ,fill = HomeTeam
  geom_bar(stat = "identity") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
```

```
  xlab("Teams")+
  ylab("Home Win %")+
  ggtitle( "Home Win Percentage for Teams ")
```




```
boxplot(df_home_ground$win_percentage)
```



Analysis- 2 Inference

i.e Explain how man United when played in their home ground have a 0.71 possiblity of winning

Hypothesis / Problem statement - 1

```
## Hypothesis : Result at the end of first half will might well be the final result

df_htr <- cbind(df  %>%filter(HTR == FTR) %>% summarise(HTR_is_same_as_FTR =n()),
                df %>% filter(HTR != FTR) %>% summarize(HTR_is_not_same_as_FTR = n()))
df_htr <- df_htr %>%
  mutate(HTR_is_same_as_FTR_per = HTR_is_same_as_FTR/(HTR_is_same_as_FTR+HTR_is_not_same_as_FTR))
df_htr
```

```
##   HTR_is_same_as_FTR HTR_is_not_same_as_FTR HTR_is_same_as_FTR_per
## 1               4371                   2889              0.6020661
```

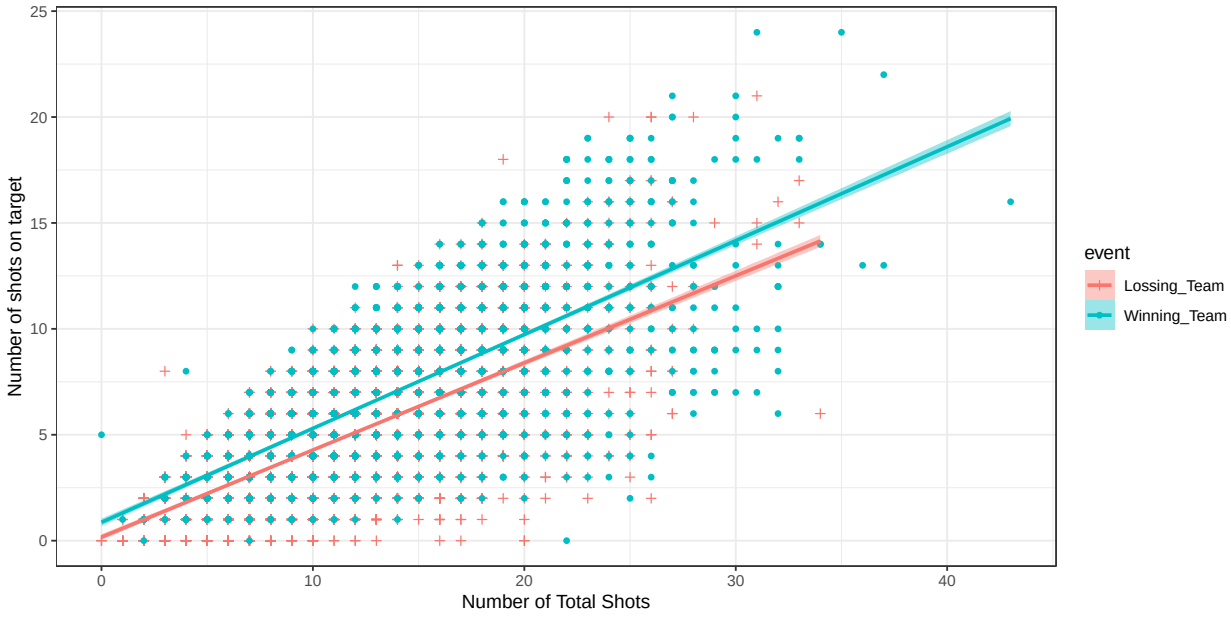
Inference

i.e About 60 % of the time the results at the end of first half well as become the result at the end time . It does not support our hypothesis effectiently (i.e 60% is just 10 % more than the equal possibility)

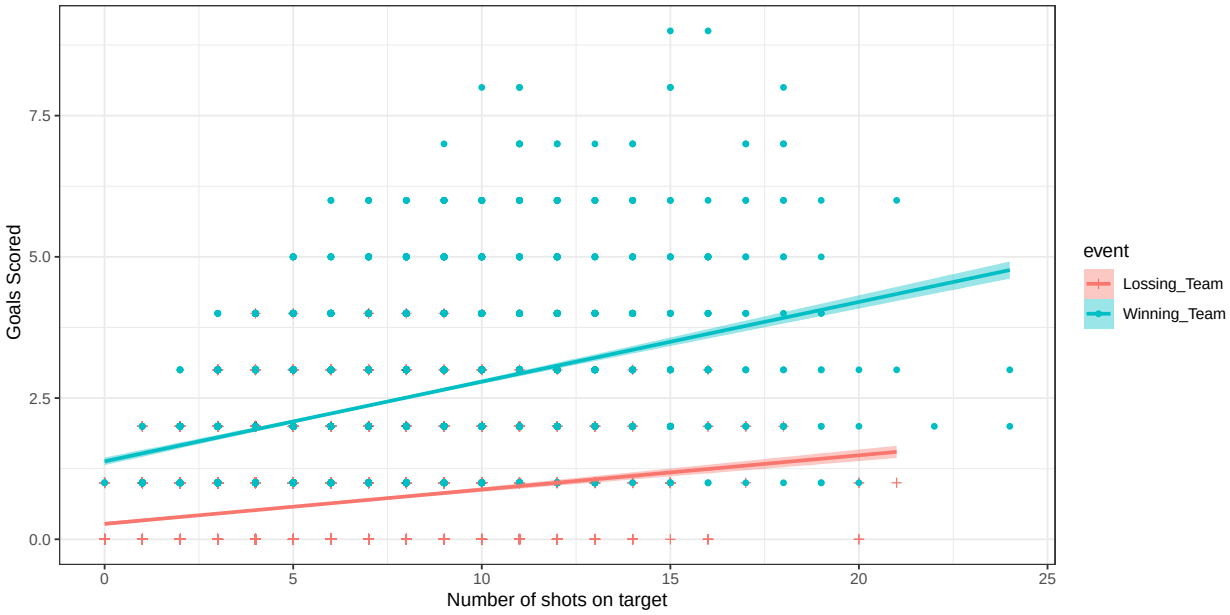
Hypothesis / PS - 2

```
# Does Shot on target have any relation with goals ?
#Just first find it for winning teams and compare it with losing team
# For Wining Team
df_st_h <- df %>% filter(FTR =='H' ) %>% select(FTHG,HST, HS)
colnames(df_st_h)<- c('GoalsScored','ST', 'S')

df_st_a <- df %>% filter(FTR =='A' ) %>% select(FTAG,AST, AS)
colnames(df_st_a)<- c('GoalsScored','ST', 'S')

df_st_w <- rbind(df_st_a ,df_st_h)
df_st_w$event <- "Winning_Team"

#For Lossing Team

df_st_h <- df %>% filter(FTR =='H' ) %>% select(FTAG,AST, AS)
colnames(df_st_h)<- c('GoalsScored','ST', 'S')

df_st_a <- df %>% filter(FTR =='A' ) %>% select(FTHG,HST, HS)
colnames(df_st_a)<- c('GoalsScored','ST', 'S')

df_st_l <- rbind(df_st_a ,df_st_h)

df_st_l$event <- "Lossing_Team"

df_st <- rbind(df_st_l , df_st_w)
```

```
#Building a linear model
ggplot(df_st , aes(x= S ,y = ST,colour = event , fill = event)) +
  geom_point(aes(shape=event)) +
  scale_shape_manual(values=c(3,16))+
  geom_smooth(method = "lm")+
  theme_bw() +
  xlab("Number of Total Shots") +
  ylab("Number of shots on target")
```




```
#Building a linear model
ggplot(df_st , aes(x= ST ,y = GoalsScored,colour = event , fill = event)) +
  geom_point(aes(shape=event)) +
  scale_shape_manual(values=c(3,16))+
  geom_smooth(method = "lm")+
  theme_bw() +
  xlab("Number of shots on target") +
  ylab(" Goals Scored")
```

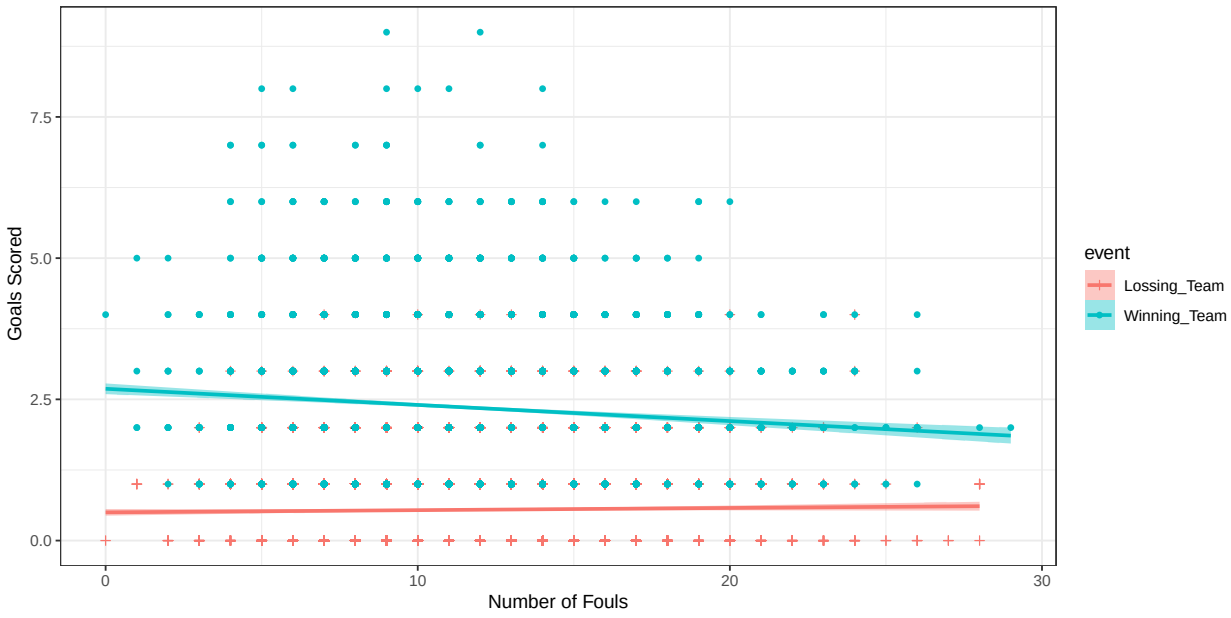


Hypothesis / PS -3

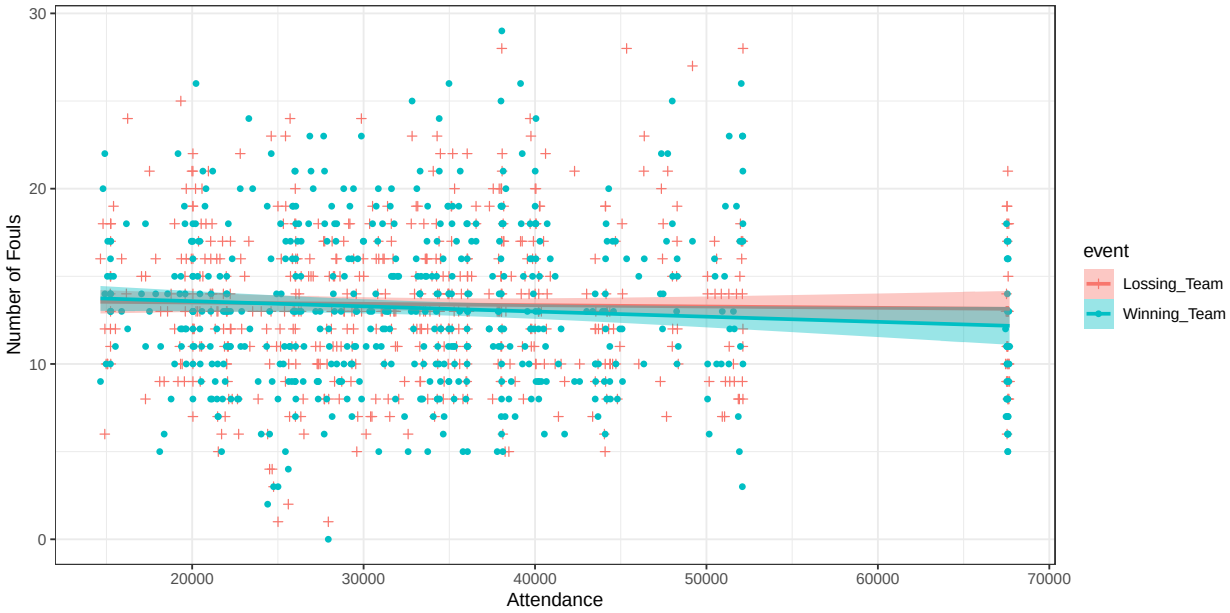
```
# Commiting more fouls will eventualy result in less goals .
# Comparing it for wining and lossing side

df_fouls_h <- df %>% filter(FTR =='H' ) %>% select(FTHG,HF,Attendance)
colnames(df_fouls_h)<- c('GoalsScored','Fouls', 'Attendance')

df_fouls_a <- df %>% filter(FTR =='A' ) %>% select(FTAG,AF, Attendance)
colnames(df_fouls_a)<- c('GoalsScored','Fouls', 'Attendance')

df_fouls_w <- rbind(df_fouls_a ,df_fouls_h)
df_fouls_w$event <- "Winning_Team"

#For Lossing Team

df_fouls_h <- df %>% filter(FTR =='H' ) %>% select(FTAG,AF, Attendance)
colnames(df_fouls_h)<- c('GoalsScored','Fouls', 'Attendance')

df_fouls_a <- df %>% filter(FTR =='A' ) %>% select(FTHG,HF, Attendance)
colnames(df_fouls_a)<- c('GoalsScored','Fouls', 'Attendance')

df_fouls_l <- rbind(df_fouls_a ,df_fouls_h)

df_fouls_l$event <- "Lossing_Team"

df_fouls <- rbind(df_fouls_l , df_fouls_w)
```

```
ggplot(df_fouls , aes(x= Fouls ,y = GoalsScored,colour = event , fill = event)) +
  geom_point(aes(shape=event)) +
```

```
  scale_shape_manual(values=c(3,16))+
  geom_smooth(method = "lm")+
  theme_bw() +
  xlab("Number of Fouls") +
  ylab(" Goals Scored")
```




```
ggplot(df_fouls , aes(x= Attendance ,y = Fouls,colour = event , fill = event)) +
  geom_point(aes(shape=event)) +
  scale_shape_manual(values=c(3,16))+
  geom_smooth(method = "lm")+
  theme_bw() +
  xlab("Attendance") +
  ylab("Number of Fouls")
```


Number of Fouls
30
20
10
0
20000
30000
40000
50000
60000
70000
Attendance
event
Lossing_Team
Winning_Team