Boyu Chen 678 Midterm Project

Boyu Chen

12/4/2021

```
df<-read.csv("stats.csv",header = TRUE)
goals<-df$goals
offside<-df$total_offside
ontarget<-df$ontarget_scoring_att
interception<-df$interception
pass<-df$total_pass
season<-df$season
tackle<-df$total_tackle
# Calculate the total credit
win<-df$wins
losse<-df$losses
tie<-38-win-losse
totalpoints<-win*3+tie*1+losse*0
```

Introduction: The era of data is silently influencing and transforming human life in unimaginable ways, and in sports, the collection and analysis of data is making the sport harmonious and controllable. With the development of technology, soccer has also become surrounded by data and has a wealth of tools to interpret this data. While data analysis may not change the outcome of a game, it can have a huge impact on the outcome of a game. For soccer, the one of the important way to make team more competitive is to use data analytics; and for data analytics to be used in soccer is a testament to its importance.

One example is the penalty that Pogba conceded in the match between Manchester United and Wolves. After collecting and analyzing data on Pogba's penalty taking habits, Wolves' data analysis team found that Pogba would kick the penalty more to the left when his team was in a tie or trailing. Because they told Wolves goalkeeper Patricio in advance, then Patricio was guarding the penalty was moving to the left in advance, and indeed pounced on the penalty. In combination, data analysis can really touch the habits and subconscious performance of players, something that is difficult for players to change and adjust, after all, in many cases it is already set in stone.

First, here is the variable I used in this project.

```
colnames<-c("Goals","Offside","Ontarget","Interception","Pass","Tackle","Total Points","Season")
explaination<-c("Number of goal for each team","Number of offsides in single season","Number of shots on
table <- cbind(colnames, explaination)
colnames(table) <- c("Variables", "Explanation")
knitr::kable(table, "pipe")
```

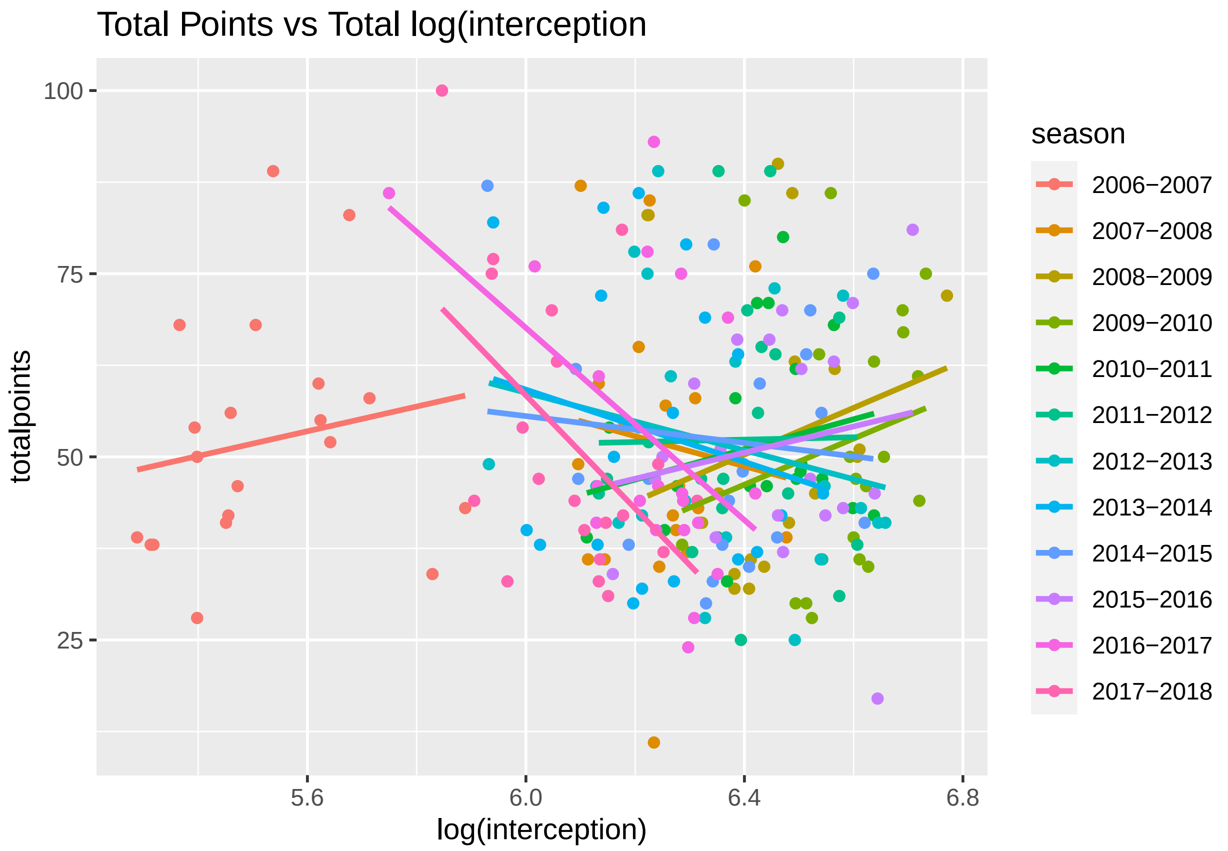
Variables	Explanation
Goals	Number of goal for each team
Offside	Number of offsides in single season

Variables	Explanation
Ontarget	Number of shots on target in single season
Interception	Number of interceptions in single season
Pass	Total number of passing ball in single season
Tackle	Total tackle in single season
Total Points	Total points in single season
Season	Season

EDA Part

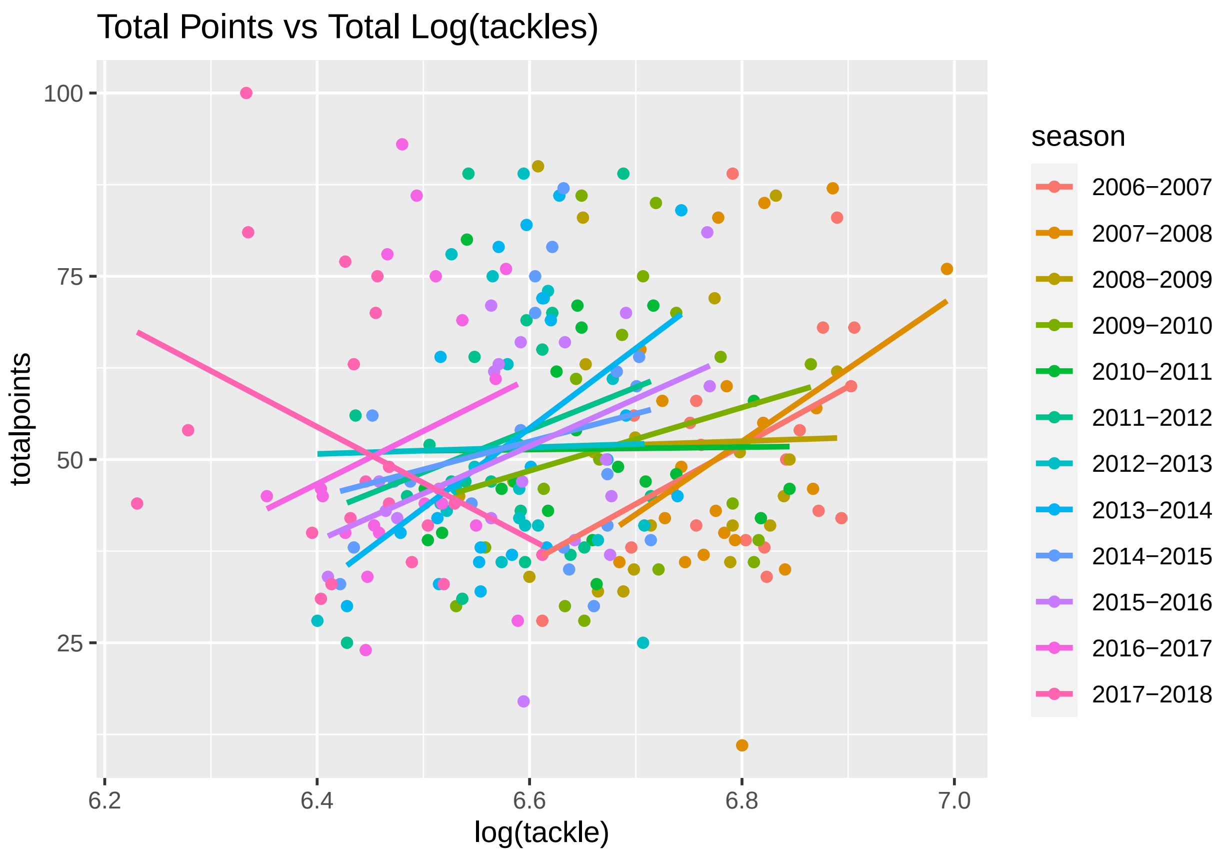
```
par(mfrow = c(3,3))
ggplot(data=df, mapping=aes(x=log(interception), y=totalpoints, group=season)) +
 # geom_line(aes(linetype=season,color=season))+
  geom_point(aes(color=season))+
  geom_smooth(se = F,aes(color = season), method = "lm")+
  labs(title = "Total Points vs Total log(interception")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



```
ggplot(data=df, mapping=aes(x=log(tackle), y=totalpoints, group=season)) +
 # geom_line(aes(linetype=season,color=season))+
  geom_point(aes(color=season))+
  geom_smooth(se = F,aes(color = season), method = "lm")+
  labs(title = "Total Points vs Total Log(tackles)")
```

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



```
ggplot(data=df, mapping=aes(x=goals, y=totalpoints, group=season)) +
 # geom_line(aes(linetype=season,color=season))+
  geom_point(aes(color=season))+
  geom_smooth(se = F,aes(color = season), method = "lm")+
  labs(title = "Total Points vs Total Goals")
```

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

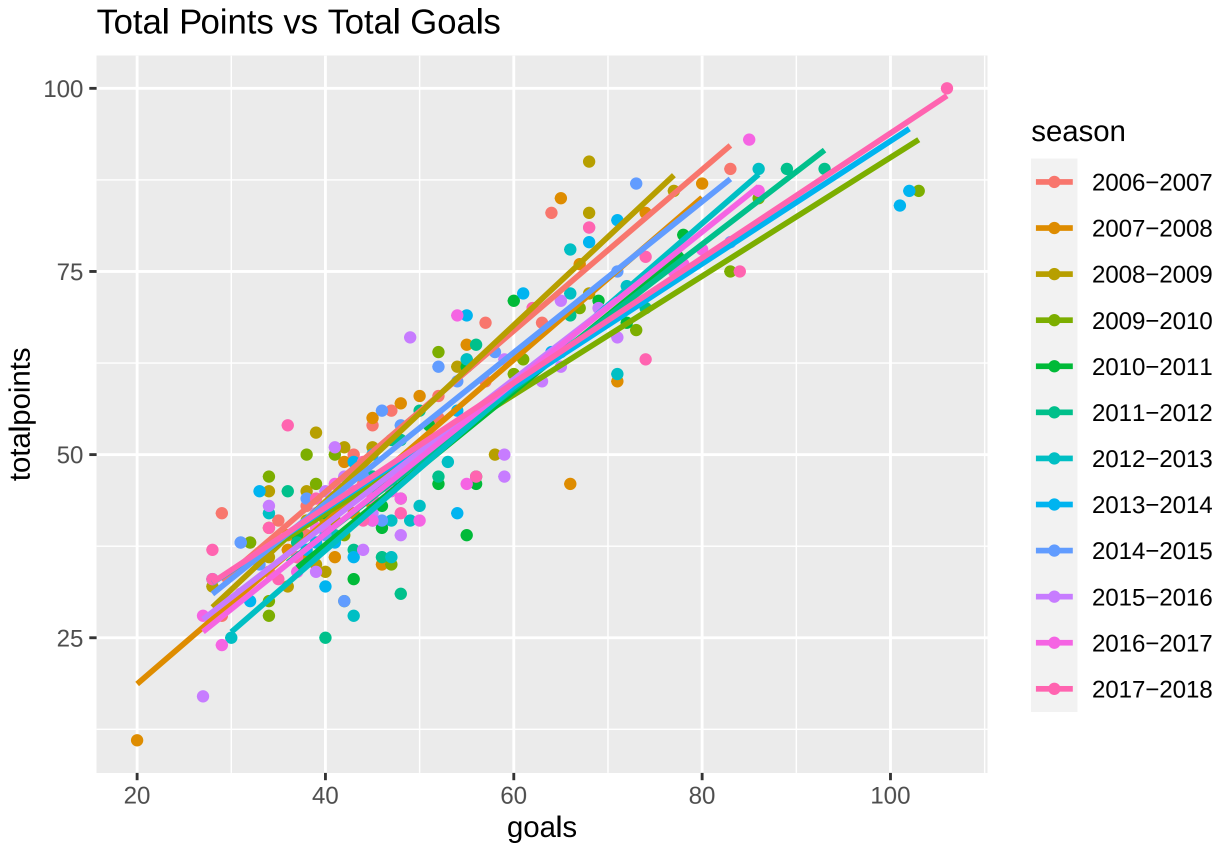


Figure 1 is the relationship between the number of goals and total points for each season. From the plot we can find that for all the season, the more goals on the court lead higher total points. The largest slope for the line is season 2008-2009. Because there are an obvious relationship between goals and total points. So I add this indicator to my model.

```
## Make model and model fitted
fit1<-lmer(totalpoints~goals+log(interception)+offside+log(tackle)+ontarget+log(pass)+(1+log(pass)|seas
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
## boundary (singular) fit: see ?isSingular
```

```
summary(fit1)
```

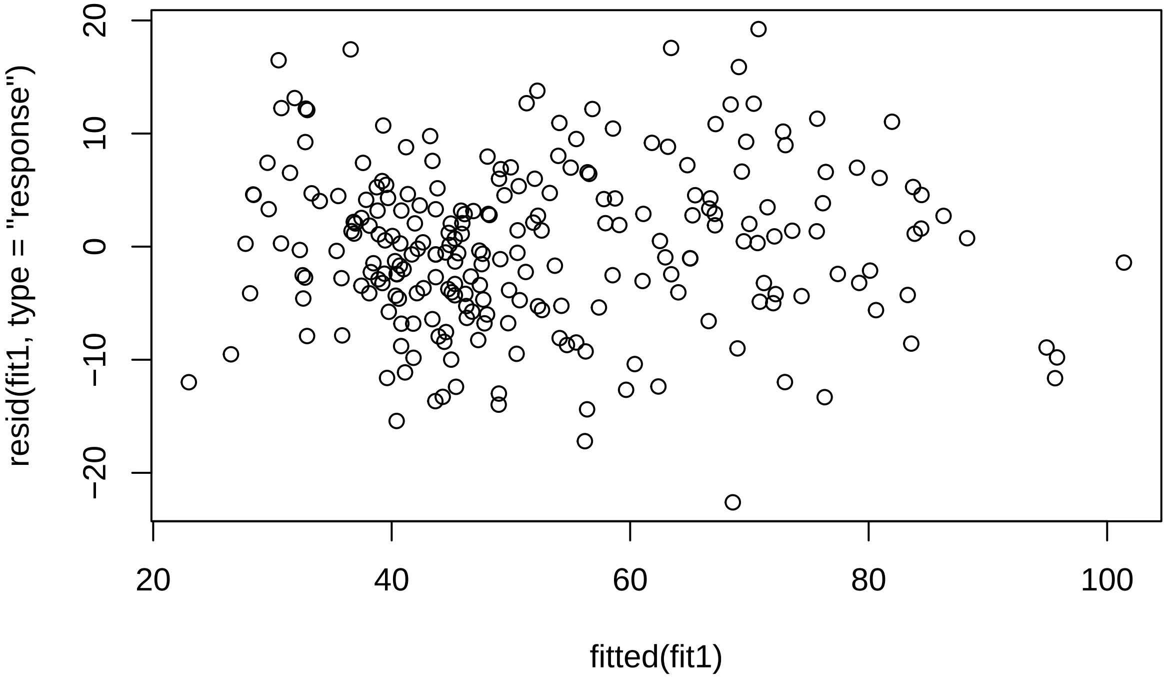
```
## Linear mixed model fit by REML ['lmerMod']
## Formula: totalpoints ~ goals + log(interception) + offside + log(tackle) +
##     ontarget + log(pass) + (1 + log(pass) | season) + goals *      ontarget
##    Data: df
##
## REML criterion at convergence: 1643.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.12003 -0.60016  0.03676  0.62660  2.65445
```

```
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  season   (Intercept) 58.4483  7.6452
##           log(pass)    0.7924  0.8902   -1.00
##  Residual             52.5126  7.2466
## Number of obs: 240, groups:  season, 12
##
## Fixed effects:
##                    Estimate Std. Error t value
## (Intercept)       -1.270e+02  5.374e+01  -2.363
## goals              9.942e-01  1.549e-01   6.417
## log(interception) -3.185e+00  1.783e+00  -1.787
## offside            3.889e-02  2.612e-02   1.489
## log(tackle)        5.971e+00  4.352e+00   1.372
## ontarget           7.371e-02  5.041e-02   1.462
## log(pass)          1.055e+01  4.022e+00   2.622
## goals:ontarget    -9.414e-04  7.221e-04  -1.304
##
## Correlation of Fixed Effects:
##             (Intr) goals  lg(nt) offsid lg(tc) ontrgt lg(ps)
## goals        0.031
## lg(ntrcptn) -0.151 -0.111
## offside     -0.032 -0.092  0.238
## log(tackle) -0.754 -0.066  0.003 -0.169
## ontarget     0.309  0.526 -0.134 -0.251 -0.312
## log(pass)   -0.849 -0.121 -0.069  0.089  0.369 -0.334
## goals:ntrgt -0.033 -0.908  0.149  0.161  0.129 -0.791  0.085
## fit warnings:
## Some predictor variables are on very different scales: consider rescaling
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

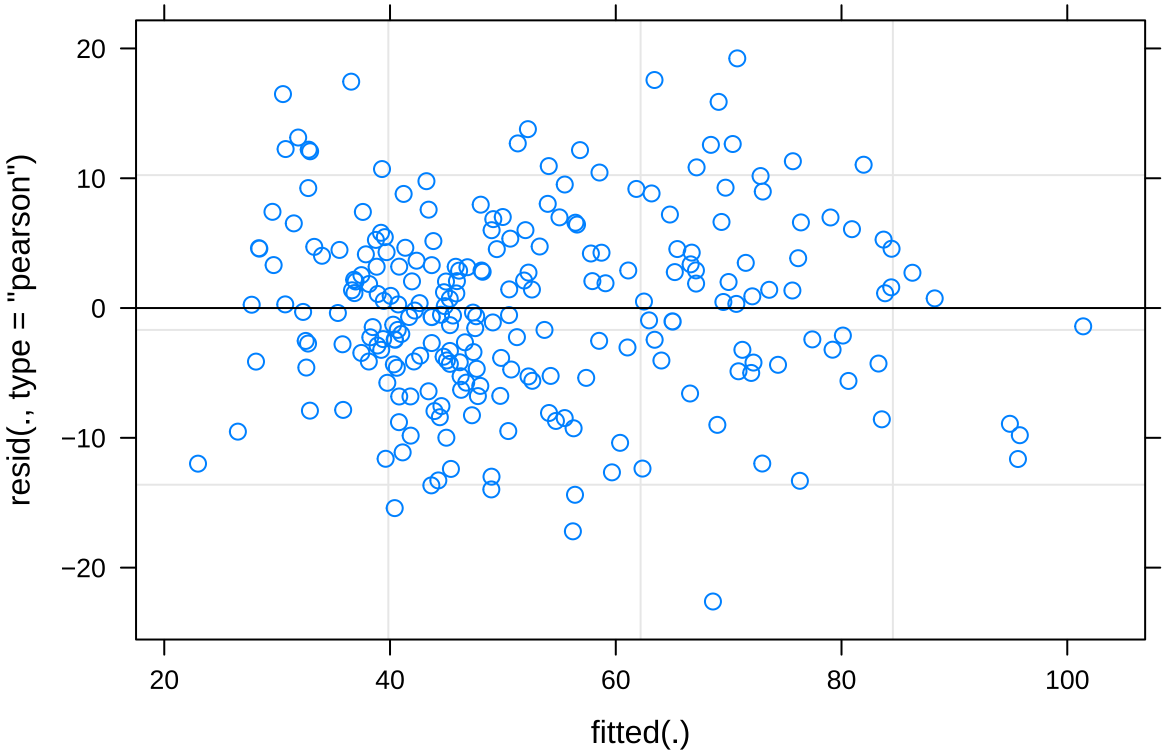
```
# fit2<-lm(totalpoints~goals+interception+offside+ontarget+log(pass) ,data = df)
fit1
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: totalpoints ~ goals + log(interception) + offside + log(tackle) +
##     ontarget + log(pass) + (1 + log(pass) | season) + goals *      ontarget
##    Data: df
## REML criterion at convergence: 1643.418
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  season   (Intercept) 7.6452
##           log(pass)   0.8902   -1.00
##  Residual             7.2466
## Number of obs: 240, groups:  season, 12
## Fixed Effects:
##       (Intercept)              goals  log(interception)            offside
##        -1.270e+02          9.942e-01         -3.185e+00          3.889e-02
##       log(tackle)           ontarget          log(pass)     goals:ontarget
##         5.971e+00          7.371e-02          1.055e+01         -9.414e-04
## fit warnings:
## Some predictor variables are on very different scales: consider rescaling
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
```

```
plot(fitted(fit1), resid(fit1,type = "response"))
```



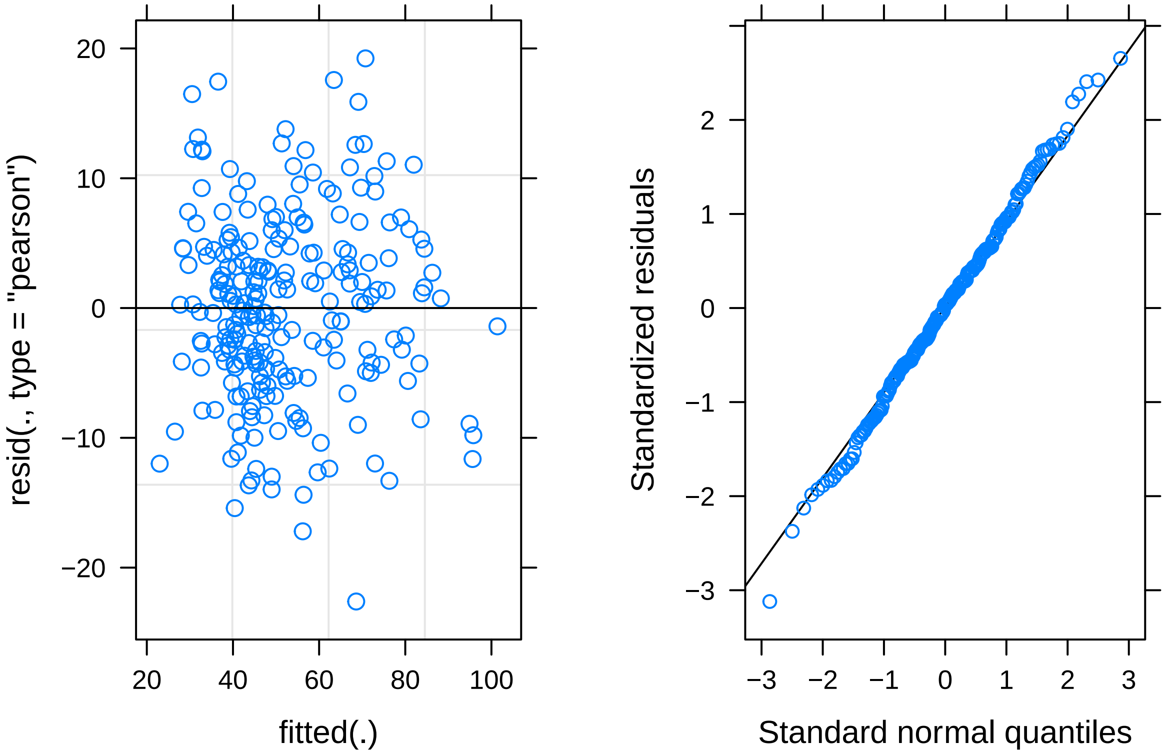
```
plot(fit1)
```



```
AIC(fit1)
```

```
## [1] 1667.418
```

```
re <- plot(fit1)
qq <- lattice::qqmath(fit1)
grid.arrange(re,qq,nrow=1)
```

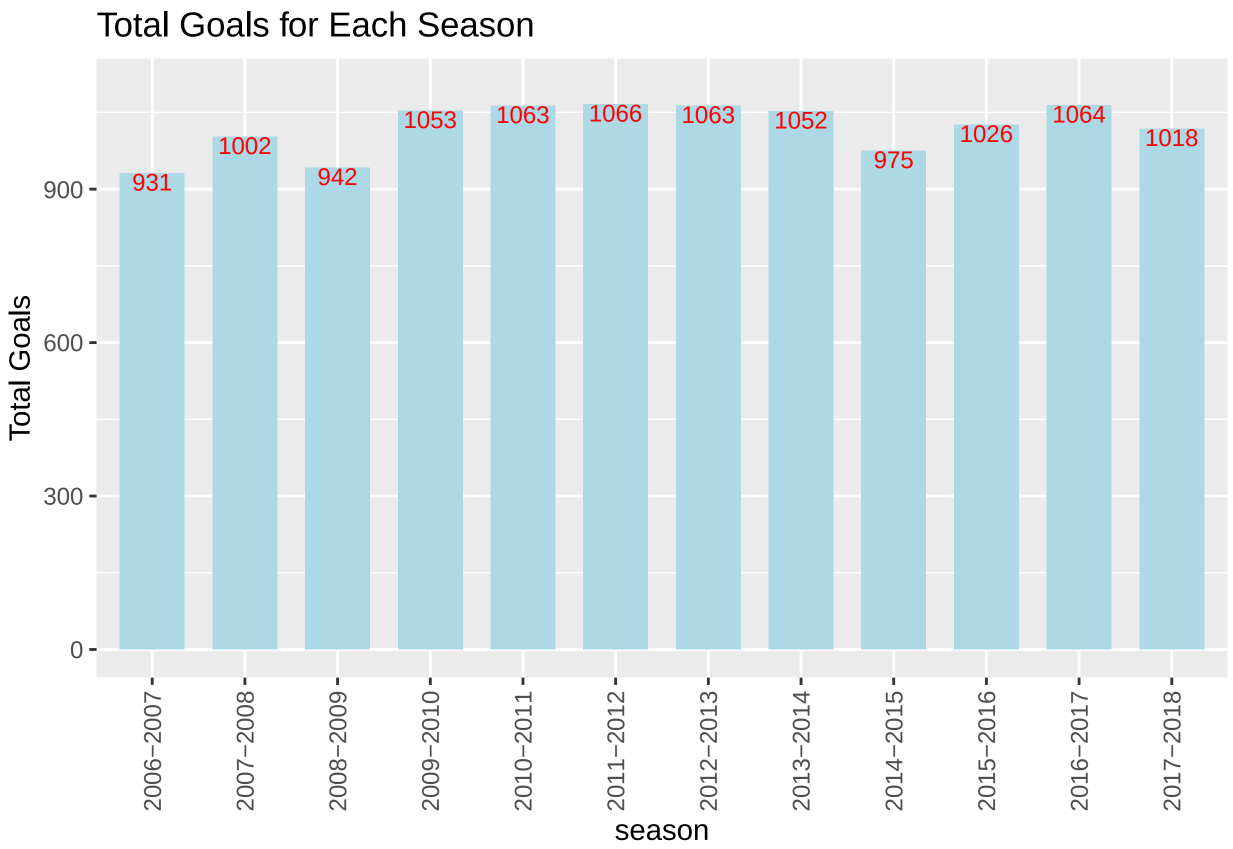


```
coefficients(fit1)
```

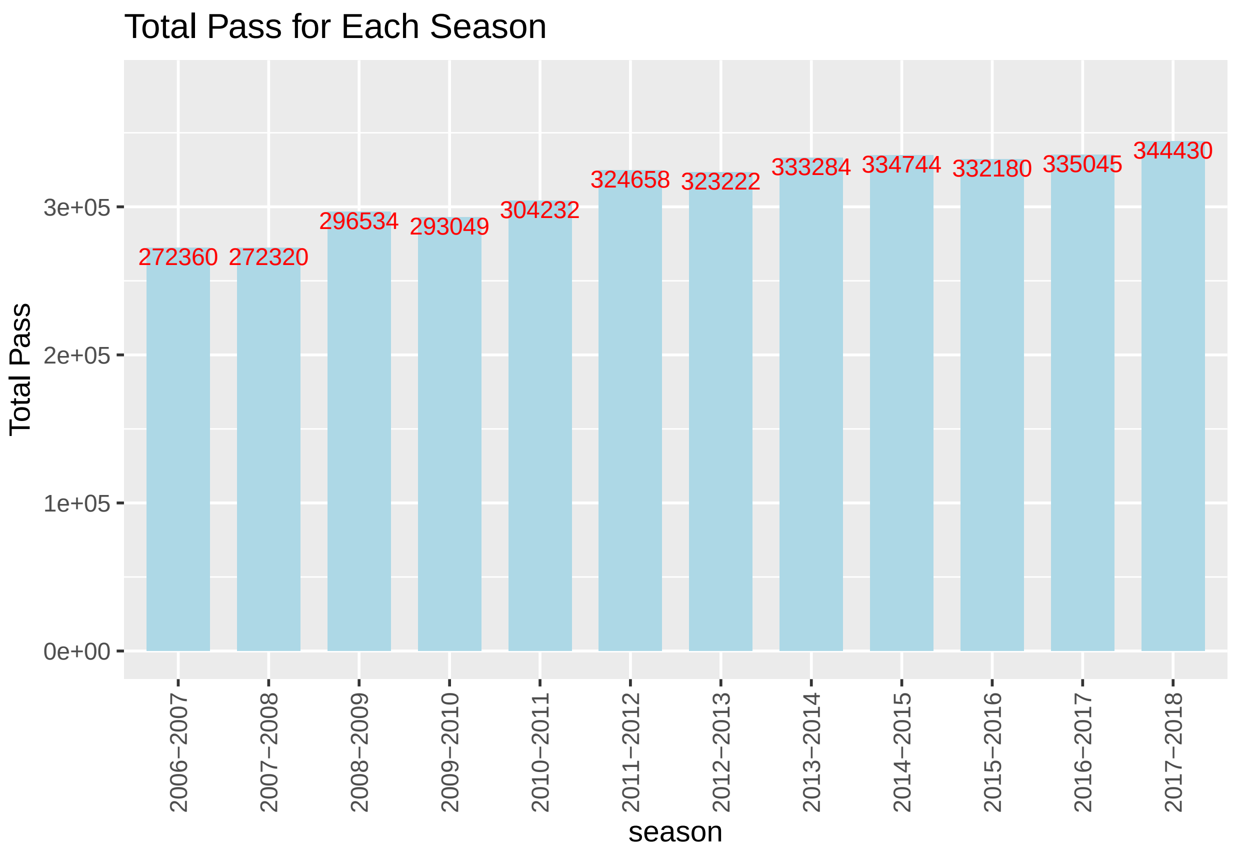
```
## $season
##            (Intercept)     goals log(interception)    offside log(tackle)
## 2006-2007   -128.8898 0.9941952         -3.185185 0.03889435    5.970778
## 2007-2008   -128.7829 0.9941952         -3.185185 0.03889435    5.970778
## 2008-2009   -134.1595 0.9941952         -3.185185 0.03889435    5.970778
## 2009-2010   -125.9573 0.9941952         -3.185185 0.03889435    5.970778
## 2010-2011   -122.2811 0.9941952         -3.185185 0.03889435    5.970778
## 2011-2012   -124.0344 0.9941952         -3.185185 0.03889435    5.970778
## 2012-2013   -122.1208 0.9941952         -3.185185 0.03889435    5.970778
## 2013-2014   -125.4761 0.9941952         -3.185185 0.03889435    5.970778
## 2014-2015   -132.5074 0.9941952         -3.185185 0.03889435    5.970778
## 2015-2016   -126.1362 0.9941952         -3.185185 0.03889435    5.970778
## 2016-2017   -126.3855 0.9941952         -3.185185 0.03889435    5.970778
## 2017-2018   -127.2715 0.9941952         -3.185185 0.03889435    5.970778
##              ontarget log(pass) goals:ontarget
## 2006-2007 0.07370852 10.766445  -0.0009413933
## 2007-2008 0.07370852 10.753997  -0.0009413933
## 2008-2009 0.07370852 11.380020  -0.0009413933
## 2009-2010 0.07370852 10.425002  -0.0009413933
## 2010-2011 0.07370852  9.996960  -0.0009413933
## 2011-2012 0.07370852 10.201105  -0.0009413933
## 2012-2013 0.07370852  9.978303  -0.0009413933
## 2013-2014 0.07370852 10.368970  -0.0009413933
## 2014-2015 0.07370852 11.187659  -0.0009413933
## 2015-2016 0.07370852 10.445832  -0.0009413933
## 2016-2017 0.07370852 10.474850  -0.0009413933
## 2017-2018 0.07370852 10.578018  -0.0009413933
##
## attr(,"class")
## [1] "coef.mer"
```

Appendix

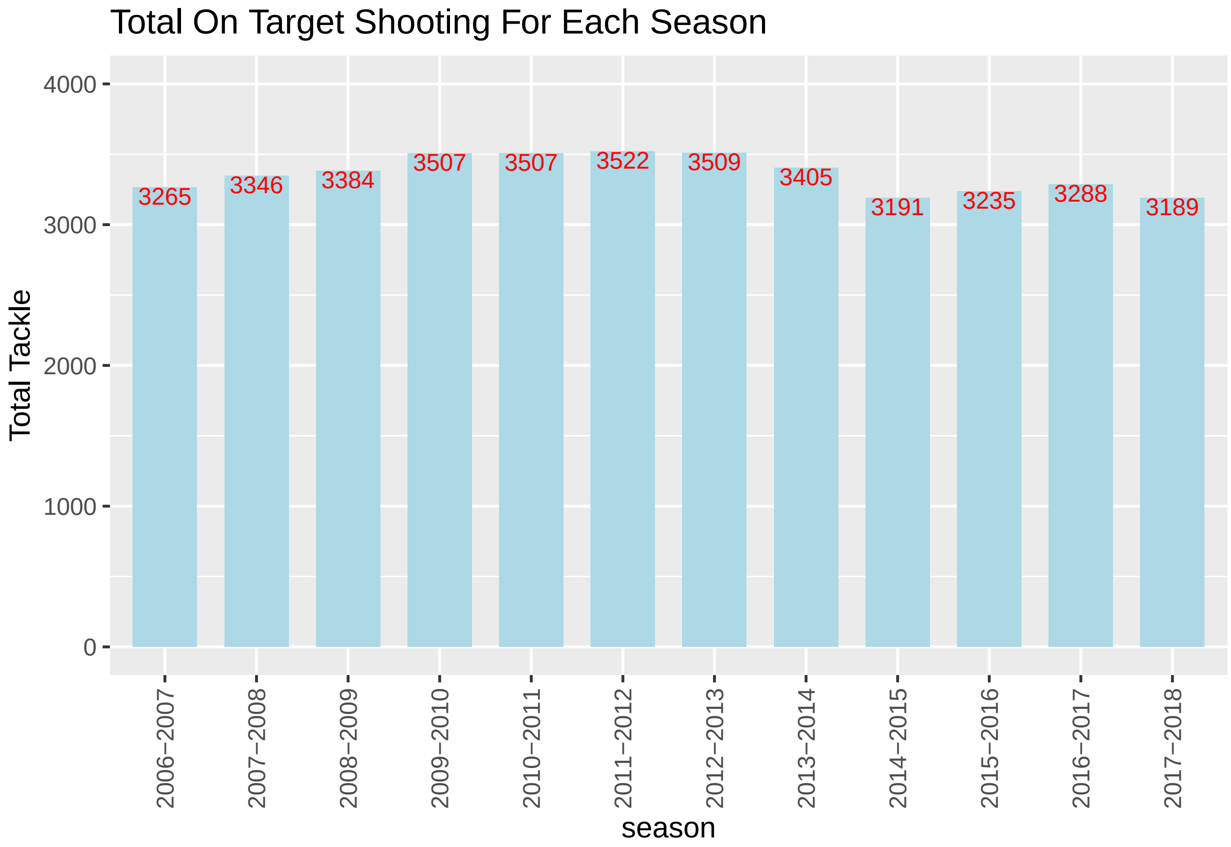
```
totalgoals<-aggregate(goals, by=list(season=df$season), FUN=sum)
ggplot(data = totalgoals, mapping = aes(x = season, y = x,label = season))+
  geom_bar(stat = 'identity',fill = 'lightblue',width = 0.7)+
  geom_text(mapping = aes(label = x), size = 3, colour = 'red', vjust = 1, hjust = .5,
          position = position_dodge(0.9))+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(x = "season",y = "Total Goals",title = "Total Goals for Each Season")+ylim(0,1100)
```



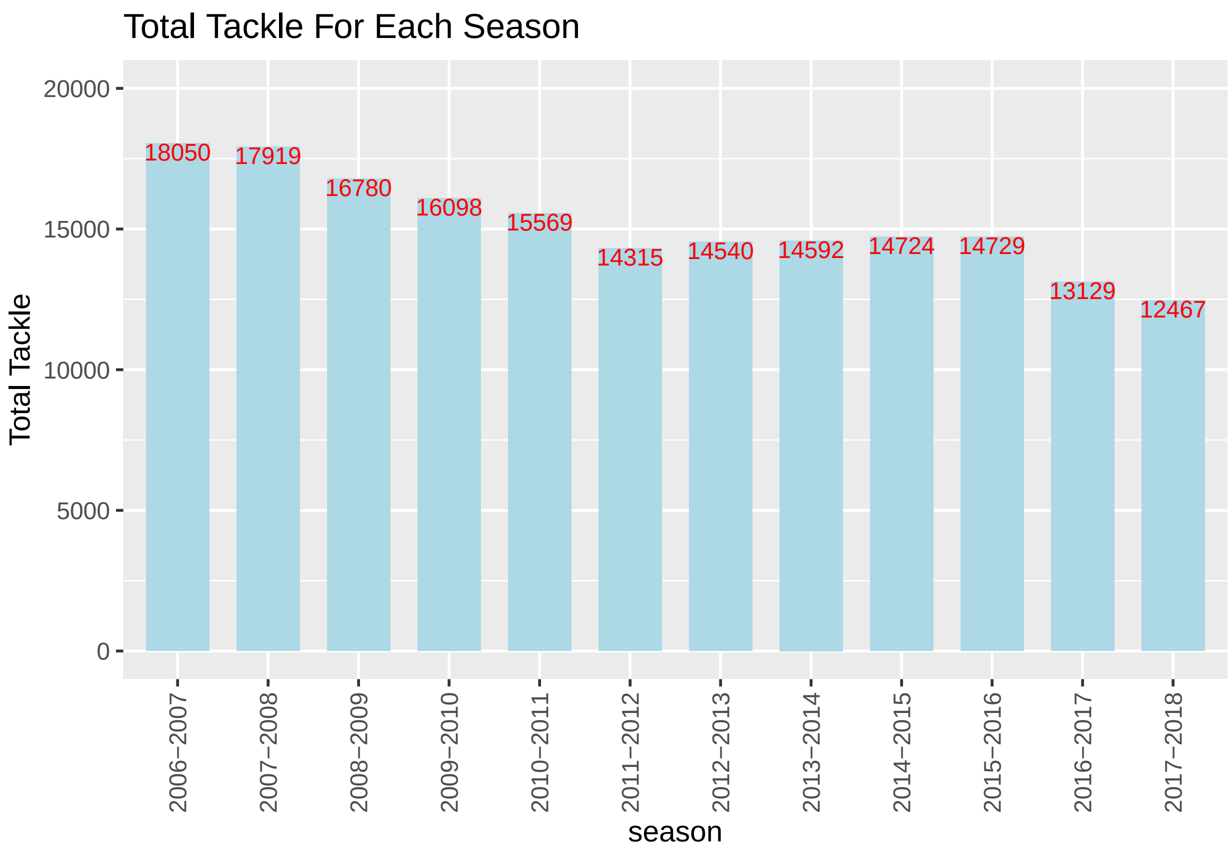
```
totalpass<-aggregate(pass, by=list(season=df$season), FUN=sum)
ggplot(data = totalpass, mapping = aes(x = season, y = x,label = season))+
  geom_bar(stat = 'identity',fill = 'lightblue',width = 0.7)+
  geom_text(mapping = aes(label = x), size = 3, colour = 'red', vjust = 1, hjust = .5,
          position = position_dodge(0.9))+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(x = "season",y = "Total Pass",title = "Total Pass for Each Season")+ylim(0,380000)
```



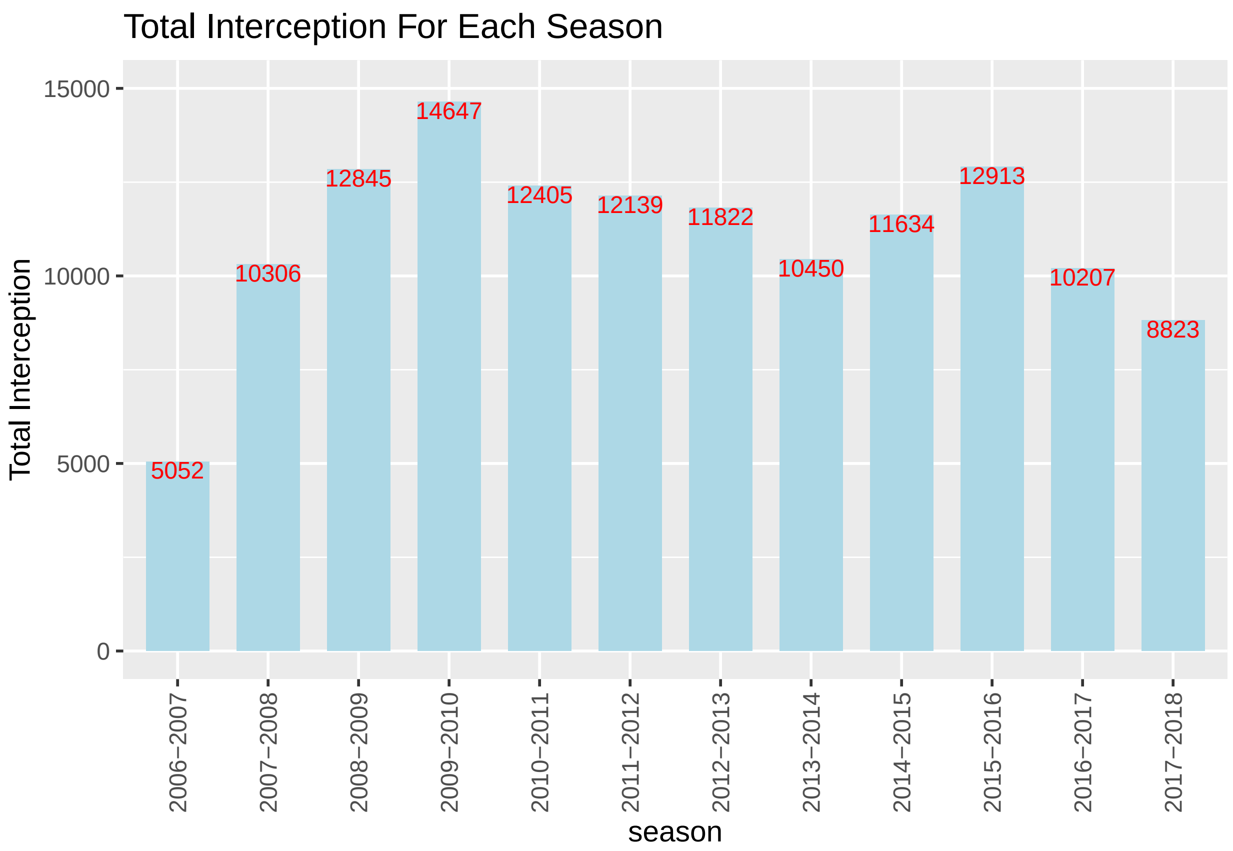
```
totalontarget<-aggregate(ontarget, by=list(season=df$season), FUN=sum)
ggplot(data = totalontarget, mapping = aes(x = season, y = x,label = season))+
  geom_bar(stat = 'identity',fill = 'lightblue',width = 0.7)+
  geom_text(mapping = aes(label = x), size = 3, colour = 'red', vjust = 1, hjust = .5,
            position = position_dodge(0.9))+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(x = "season",y = "Total Tackle",title = "Total On Target Shooting For Each Season")+ylim(0,4000)
```



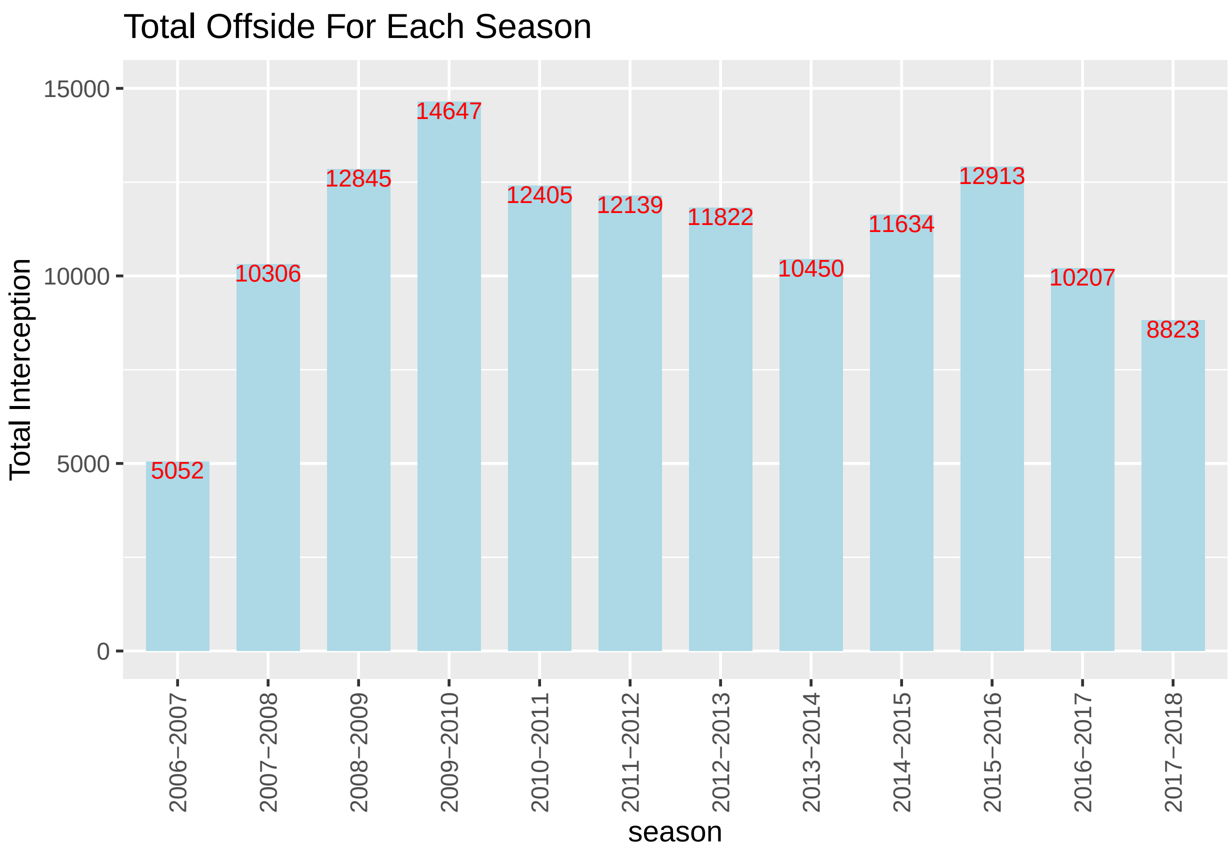
```
totaltackle<-aggregate(tackle, by=list(season=df$season), FUN=sum)
ggplot(data = totaltackle, mapping = aes(x = season, y = x,label = season))+
  geom_bar(stat = 'identity',fill = 'lightblue',width = 0.7)+
  geom_text(mapping = aes(label = x), size = 3, colour = 'red', vjust = 1, hjust = .5,
            position = position_dodge(0.9))+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(x = "season",y = "Total Tackle",title = "Total Tackle For Each Season")+ylim(0,20000)
```



```
totalinterception<-aggregate(interception, by=list(season=df$season), FUN=sum)
ggplot(data = totalinterception, mapping = aes(x = season, y = x,label = season))+
  geom_bar(stat = 'identity',fill = 'lightblue',width = 0.7)+
  geom_text(mapping = aes(label = x), size = 3, colour = 'red', vjust = 1, hjust = .5,
            position = position_dodge(0.9))+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(x = "season",y = "Total Interception",title = "Total Interception For Each Season")+ylim(0,15000
```

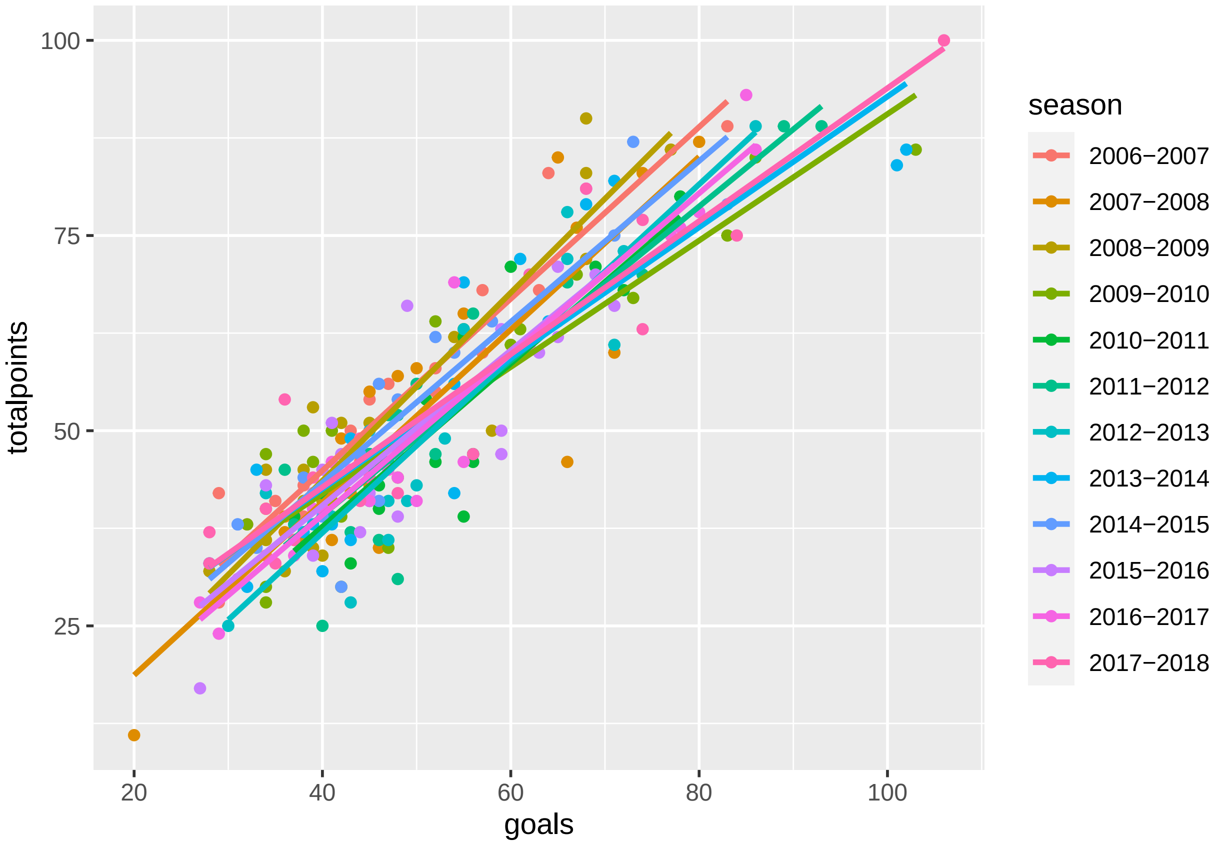


```
totaloffside<-aggregate(offside, by=list(season=df$season), FUN=sum)
ggplot(data = totalinterception, mapping = aes(x = season, y = x,label = season))+
  geom_bar(stat = 'identity',fill = 'lightblue',width = 0.7)+
  geom_text(mapping = aes(label = x), size = 3, colour = 'red', vjust = 1, hjust = .5,
            position = position_dodge(0.9))+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(x = "season",y = "Total Interception",title = "Total Offside For Each Season")+ylim(0,15000)
```



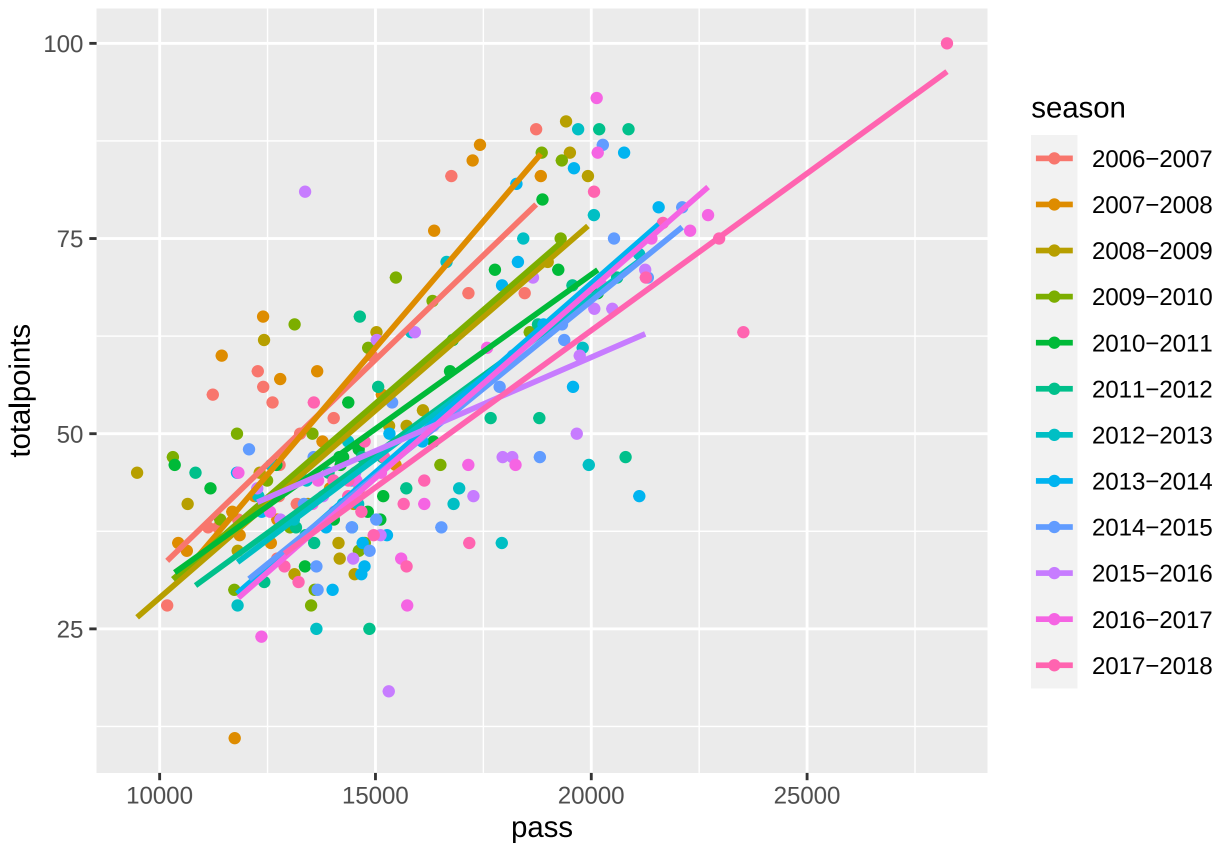
```
ggplot(data=df, mapping=aes(x=goals, y=totalpoints, group=season)) +
 # geom_line(aes(linetype=season,color=season))+
  geom_point(aes(color=season))+
  geom_smooth(se = F,aes(color = season), method = "lm")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



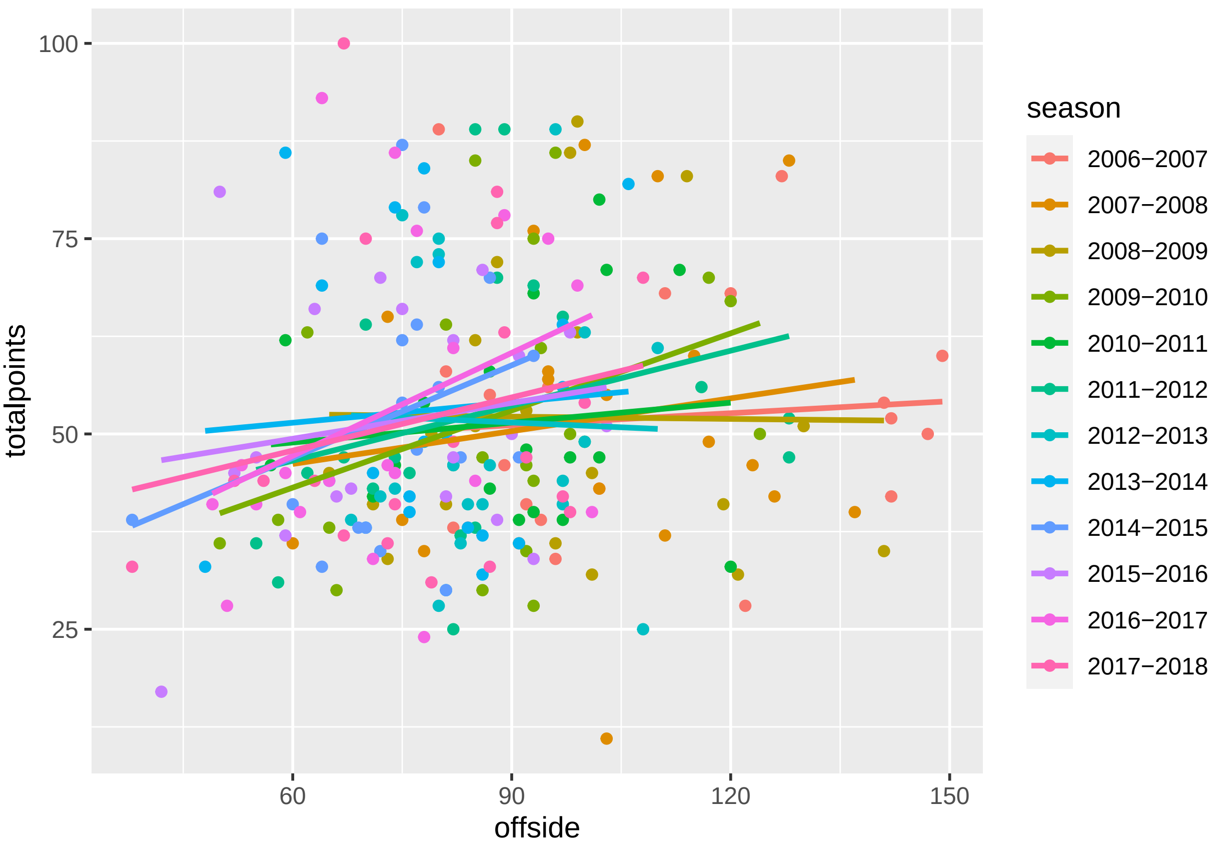
```
ggplot(data=df, mapping=aes(x=pass, y=totalpoints, group=season)) +
 # geom_line(aes(linetype=season,color=season))+
  geom_point(aes(color=season))+
  geom_smooth(se = F,aes(color = season), method = "lm")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



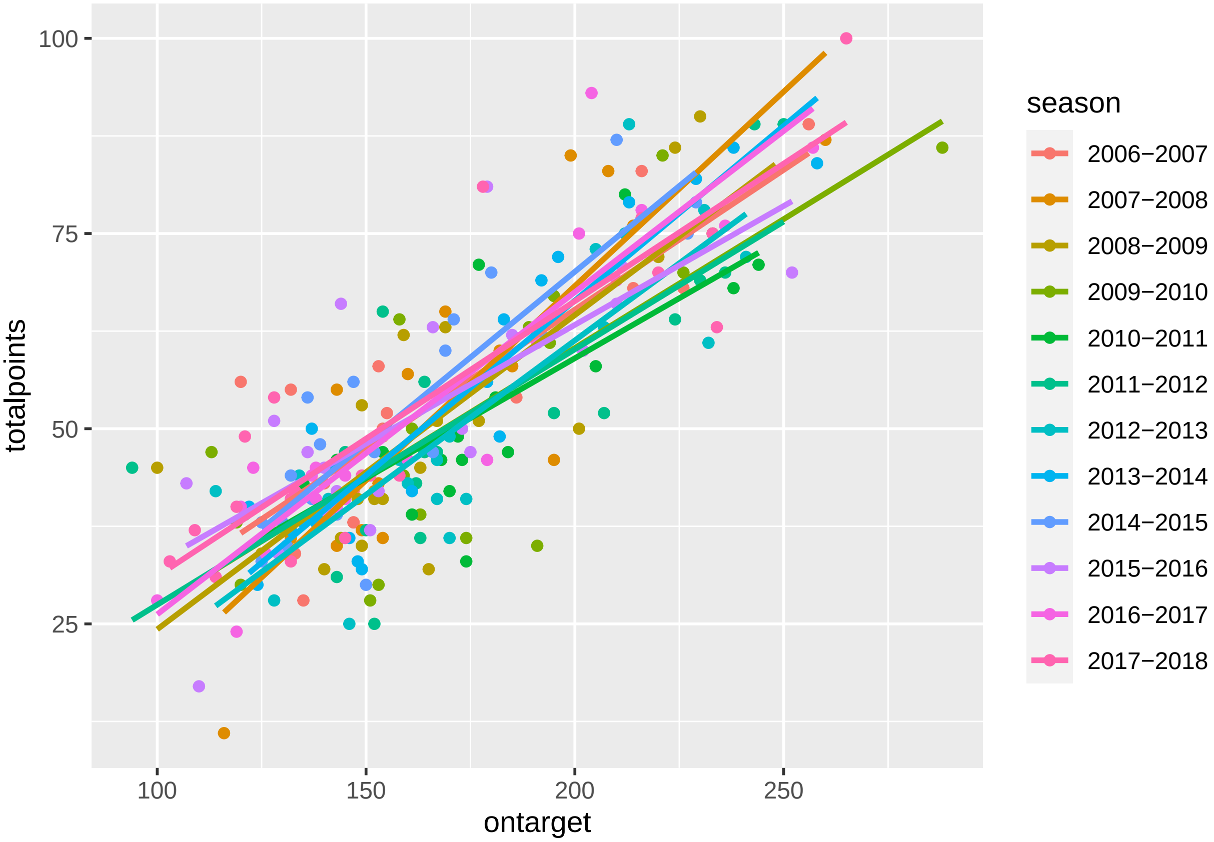
```
ggplot(data=df, mapping=aes(x=offside, y=totalpoints, group=season)) +
 # geom_line(aes(linetype=season,color=season))+
  geom_point(aes(color=season))+
  geom_smooth(se = F,aes(color = season), method = "lm")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```



```
ggplot(data=df, mapping=aes(x=ontarget, y=totalpoints, group=season)) +
 # geom_line(aes(linetype=season,color=season))+
  geom_point(aes(color=season))+
  geom_smooth(se = F,aes(color = season), method = "lm")
```

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


100
75
50
25
totalpoints
100
150
200
250
ontarget
season
2006−2007
2007−2008
2008−2009
2009−2010
2010−2011
2011−2012
2012−2013
2013−2014
2014−2015
2015−2016
2016−2017
2017−2018