Final Project Report

Kush Lalwani

2024-04-28

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(rvest)
```

```
##
## Attaching package: 'rvest'
##
## The following object is masked from 'package:readr':
##
##     guess_encoding
```

```
library(chromote)
library(ggpubr)
```

Research Question

I will be analyzing the relationship between a player's wages and their in game performance. The question is: Does a player's wages affect their performance in game? This question is important because it is important to understand if players are playing better or worse depending on their salary. This could help the teams determine if they should give their best players more money to keep their performances at a high level. Or it could also be used to decide the future of struggling players; whether they should get new contracts or sold.

Data Access and Wrangling

Primary Data

```r
wage_link<-"https://fbref.com/en/comps/9/wages/Premier-League-Wages"
wage_stats <- wage_link %>%
  read_html() %>%
  html_elements(css ="table")%>%
  html_table()

wage_stats <- wage_stats[[2]] #the player dataset is the second one on the page
head(wage_stats,3)
```

```
## # A tibble: 3 x 9
##      Rk Player      Nation Pos   Squad   Age `Weekly Wages` `Annual Wages` Notes
##   <int> <chr>       <chr>  <chr> <chr> <int> <chr>          <chr>          <chr>
## 1     1 Kevin De B~ be BEL MF    Manc~    32 £ 400,000 (€ ~ £ 20,800,000 ~ ""
## 2     2 Erling Haa~ no NOR FW    Manc~    23 £ 375,000 (€ ~ £ 19,500,000 ~ ""
## 3     3 Casemiro    br BRA MF    Manc~    31 £ 350,000 (€ ~ £ 18,200,000 ~ ""
```

Here we have the raw data directly from the webpage. We see that is not in tidy data form. The rank variable is not needed and neither is nation or notes, so they can be selected out. Also the wages are a character string in three different currencies, we want it in only dollars.

```r
# This function will be used to remove the dollar amount from the wages variables
extract_dollar <- function(value) {
  dollar_amount <- gsub(".*\\$([0-9,]+).*", "\\1", value)
  dollar_amount <- as.numeric(gsub(",", "", dollar_amount))
  return(dollar_amount)
}
```

```r
player_wages <-
  wage_stats %>%
  select(!c(Rk,Notes,Age,`Weekly Wages`,Nation)) %>%
  rename(name = Player, position = Pos, team = Squad, pay_year = `Annual Wages`) %>%
  mutate(pay_year = extract_dollar(pay_year))
head(player_wages,3)
```

```
## # A tibble: 3 x 4
##   name            position team            pay_year
##   <chr>           <chr>    <chr>              <dbl>
## 1 Kevin De Bruyne MF       Manchester City 25767947
## 2 Erling Haaland  FW       Manchester City 24157450
## 3 Casemiro        MF       Manchester Utd  22546954
```

Now the data is in tidy data form, and has been converted to US dollars ### Seconday Data Sources

```r
#This function is used to extract all the secondary data I want from the respective websites
get_table <- function(link){
  page <- link %>%
    read_html_live() %>%
    html_elements(css = "table") %>%
    html_table()
  out <- page[[3]]
  return(out)
}
```

I had trouble reading the table I needed from the website. I found out that the table was loaded using javascript and not directly in the HTML. So to solve this issue, I used a new function in rvest called read_html_live(). This function uses the chromote package, which uses google chrome to load the entire website and read it directly. Unfortunately, this function doesn't work when trying to knit into a pdf. So I will use csv files that were copy and pasted instead. Now the data needs to be put into tidy data form and I will only select the stats that I am interested in.

Goalkeepers

For goalkeepers, performance will be rated on their save percentage, goals allowed, and penalty saves

```
gk_link<-"https://fbref.com/en/comps/9/keepers/Premier-League-Stats"
#gk_stats <- get_table(gk_link)

gk_stats <- read.csv("goalkeeper stats.csv")

#colnames(gk_stats) <- as.character(gk_stats[1,])
colnames(gk_stats)[26] <- "PKSave%"
head(gk_stats,3)
```

```
##   Rk          Player  Nation Pos     Squad    Age Born MP Starts   Min X90s GA
## 1  1         Alisson  br BRA  GK Liverpool 31-197 1992 22     22 1,980   22 20
## 2  2 Alphonse Areola  fr FRA  GK  West Ham 31-049 1993 27     27 2,339   26 42
## 3  3  Daniel Bentley eng ENG  GK    Wolves 30-278 1993  4      2   293  3.3  4
##   GA90 SoTA Saves Save.  W D L CS  CS. PKatt PKA PKsv PKm PKSave% Matches
## 1 0.91   77    58  75.3 13 6 3  7 31.8     1   1    0   0       0 Matches
## 2 1.62  153   111  75.8 10 8 9  4 14.8     7   5    2   0    28.6 Matches
## 3 1.23   12     8  66.7  1 0 1  1   50     0   0    0   0         Matches
```

```
goalkeeper_stat <-
  gk_stats %>%
  select(c(Player,Pos,Squad,`X90s`,GA90,`Save.`,`PKSave%`)) %>%
  rename(name = Player, position = Pos, team = Squad, games = `X90s`, goal_against_per90 = GA90, save_p
  filter(team != "Squad") %>%
  mutate(games = as.numeric(games),goal_against_per90 = as.numeric(goal_against_per90),save_percentage =
  filter(games >= 5)

head(goalkeeper_stat,3)
```

```
##              name position          team games goal_against_per90
## 1         Alisson       GK     Liverpool  22.0               0.91
## 2 Alphonse Areola       GK      West Ham  26.0               1.62
## 3 Martin Dúbravka       GK Newcastle Utd  17.1               1.99
##   save_percentage penalty_save_percentage
## 1            75.3                     0.0
## 2            75.8                    28.6
## 3            70.6                    20.0
```

```
#Now we need to join this plot with the wage plot
goalkeeper_stat <-
  goalkeeper_stat %>%
  left_join(player_wages, by = c("name","position","team"))
```

```r
head(goalkeeper_stat,3)
```

```
##              name position          team games goal_against_per90
## 1         Alisson       GK     Liverpool  22.0               0.91
## 2 Alphonse Areola       GK      West Ham  26.0               1.62
## 3 Martin Dúbravka       GK Newcastle Utd  17.1               1.99
##   save_percentage penalty_save_percentage pay_year
## 1            75.3                     0.0  9662980
## 2            75.8                    28.6  7730384
## 3            70.6                    20.0  2576795
```

Defenders

For defenders, performance will be rated on their tackle success percentage, blocks, interceptions, and errors leading to goals.

```r
df_link<-"https://fbref.com/en/comps/9/defense/Premier-League-Stats"
#df_stats <- get_table(df_link)

df_stats <- read.csv("defender stats.csv")

colnames(df_stats) <- as.character(df_stats[1,])
head(df_stats,3)
```

```
##   Rk             Player  Nation   Pos         Squad    Age Born  90s Tkl TklW
## 1 Rk             Player  Nation   Pos         Squad    Age Born  90s Tkl TklW
## 2  1         Max Aarons eng ENG    DF   Bournemouth 24-104 2000 12.3  28   18
## 3  2 Bénie Adama Traore  ci CIV FW,MF Sheffield Utd 21-139 2002  4.3   4    2
##   Def 3rd Mid 3rd Att 3rd Tkl Att Tkl% Lost Blocks Sh Pass Int Tkl+Int Clr Err
## 1 Def 3rd Mid 3rd Att 3rd Tkl Att Tkl% Lost Blocks Sh Pass Int Tkl+Int Clr Err
## 2      20       6       2  19  30 63.3   11      9  5    4   6      34  23   0
## 3       1       2       1   0   4    0    4      4  1    3   1       5   1   0
##   Matches
## 1 Matches
## 2 Matches
## 3 Matches
```

```r
defender_stats <-
  df_stats %>%
  select(c(Player,Pos,Squad,`90s`,`Tkl%`,Blocks,Int,Err)) %>%
  rename(name=Player,position=Pos,team=Squad,games = `90s`,
         tackle_percent=`Tkl%`,block=Blocks,interceptions=Int,errors=Err) %>%
  filter(team != "Squad") %>%
  filter(grepl("DF|LB|RB|FB|CB",position)) %>% # These are all different abbreviations for defender pos
  mutate(tackle_percent=as.numeric(tackle_percent),block=as.numeric(block), games = as.numeric(games),
         interceptions=as.numeric(interceptions),errors=as.numeric(errors)) %>%
  filter(games >= 5)

head(defender_stats,3)
```

```
##              name position        team games tackle_percent block
```

```
## 1       Max Aarons       DF Bournemouth  12.3           63.3     9
## 2 Tosin Adarabioyo       DF      Fulham  17.0           61.1    15
## 3     Nayef Aguerd       DF    West Ham  20.6           72.7    33
##   interceptions errors
## 1             6      0
## 2            23      0
## 3            17      2
```

```
defender_stats <-
  defender_stats %>%
  inner_join(player_wages, by = c("name","position","team"))

head(defender_stats,3)
```

```
##               name position        team games tackle_percent block
## 1       Max Aarons       DF Bournemouth  12.3           63.3     9
## 2 Tosin Adarabioyo       DF      Fulham  17.0           61.1    15
## 3     Nayef Aguerd       DF    West Ham  20.6           72.7    33
##   interceptions errors pay_year
## 1             6      0  2254695
## 2            23      0  2576795
## 3            17      2  3220993
```

Midfielders

For midfielders, performance will be based on their pass completion percentage, assists (actual and expected), and progressive passes.

```
md_link<-"https://fbref.com/en/comps/9/passing/Premier-League-Stats"
#md_stats <- get_table(md_link)

md_stats <- read.csv("midfield stats.csv")

colnames(md_stats) <- as.character(md_stats[1,])
#all these variables have the same name, so they need to be renamed
colnames(md_stats)[16] <- "short_cmp%"
colnames(md_stats)[19] <- "med_cmp%"
colnames(md_stats)[22] <- "long_cmp%"
head(md_stats,3)
```

```
##   Rk             Player  Nation   Pos         Squad    Age Born  90s Cmp Att
## 1 Rk             Player  Nation   Pos         Squad    Age Born  90s Cmp Att
## 2  1         Max Aarons eng ENG    DF   Bournemouth 24-104 2000 12.3 394 516
## 3  2 Bénie Adama Traore  ci CIV FW,MF Sheffield Utd 21-139 2002  4.3  55  71
##   Cmp% TotDist PrgDist Cmp Att short_cmp% Cmp Att med_cmp% Cmp Att long_cmp%
## 1 Cmp% TotDist PrgDist Cmp Att       Cmp% Cmp Att     Cmp% Cmp Att      Cmp%
## 2 76.4    6518    2525 193 220       87.7 163 209       78  32  57      56.1
## 3 77.5     775     185  34  38       89.5  19  23     82.6   1   1       100
##   Ast xAG  xA A-xAG KP 3-Jan PPA CrsPA PrgP Matches
## 1 Ast xAG  xA A-xAG KP 3-Jan PPA CrsPA PrgP Matches
## 2   1 0.8 0.9   0.2  7    22  13     2   41 Matches
## 3   0 0.5 0.5  -0.5  4     2   7     1    9 Matches
```

```
midfield_stats <- md_stats %>%
  select(c(Player, Pos, Squad, `90s`, `Cmp%`, Ast, xA, PrgP)) %>%
  rename(name = Player, position = Pos, team = Squad, games = `90s`, cmp_perc = `Cmp%`, assists = Ast,
         xAssists = xA, prog_pass = PrgP) %>%
  filter(name != "Player") %>%
  filter(grepl("MF|AM|DM|LM|WM|RM", position)) %>%
  mutate(games = as.numeric(games), cmp_perc = as.numeric(cmp_perc), assists = as.numeric(assists),
         xAssists = as.numeric(xAssists), prog_pass = as.numeric(prog_pass)) %>%
  filter(games >= 5)

head(midfield_stats,3)
```

```
##              name position            team games cmp_perc assists xAssists
## 1 Rayan Aït-Nouri    DF,MF          Wolves  21.2     84.8       1      1.5
## 2   Manuel Akanji    DF,MF Manchester City  22.1     93.2       0      1.4
## 3   Edson Álvarez       MF        West Ham  23.9     85.6       1      0.7
##   prog_pass
## 1        90
## 2       124
## 3        79
```

```
midfield_stats <-
  midfield_stats %>%
  inner_join(player_wages, by = c("name","position","team"))

head(midfield_stats,3)
```

```
##              name position            team games cmp_perc assists xAssists
## 1 Rayan Aït-Nouri    DF,MF          Wolves  21.2     84.8       1      1.5
## 2   Manuel Akanji    DF,MF Manchester City  22.1     93.2       0      1.4
## 3   Edson Álvarez       MF        West Ham  23.9     85.6       1      0.7
##   prog_pass pay_year
## 1        90   644199
## 2       124 11595576
## 3        79  6441987
```

Forwards

For forwards, performance will be based on their goals (actual and expected) and shot on target percentage.

```
fw_link<-"https://fbref.com/en/comps/9/shooting/Premier-League-Stats"
#fw_stats <- get_table(fw_link)

fw_stats <- read.csv("attacker stats.csv")

colnames(fw_stats) <- as.character(fw_stats[1,])

head(fw_stats,3)
```

```
##   Rk             Player  Nation   Pos         Squad    Age Born  90s Gls Sh SoT
## 1 Rk             Player  Nation   Pos         Squad    Age Born  90s Gls Sh SoT
```

```
## 2  1         Max Aarons eng ENG    DF   Bournemouth 24-104 2000 12.3   0  2   0
## 3  2 Bénie Adama Traore  ci CIV FW,MF Sheffield Utd 21-139 2002  4.3   0  1   1
##   SoT% Sh/90 SoT/90 G/Sh G/SoT Dist FK PK PKatt  xG npxG npxG/Sh G-xG np:G-xG
## 1 SoT% Sh/90 SoT/90 G/Sh G/SoT Dist FK PK PKatt  xG npxG npxG/Sh G-xG np:G-xG
## 2    0  0.16      0    0       23.9  0  0     0   0    0    0.02    0       0
## 3  100  0.23   0.23    0     0 15.3  0  0     0 0.3  0.3    0.27 -0.3    -0.3
##   Matches
## 1 Matches
## 2 Matches
## 3 Matches
```

```
attack_stats <- fw_stats %>%
  select(c(Player, Pos, Squad, `90s`,Gls,`SoT%`,xG)) %>%
  rename(name = Player, position = Pos, team = Squad, games = `90s`,goals = Gls,
         shot_target_perc = `SoT%`,xGoal = xG) %>%
  filter(name != "Player") %>%
  filter(grepl("FW|CF|LW|RW", position)) %>%
  mutate(games = as.numeric(games),goals = as.numeric(goals),
         shot_target_perc=as.numeric(shot_target_perc),xGoal = as.numeric(xGoal)) %>%
  filter(games >= 5)

head(attack_stats,3)
```

```
##             name position          team games goals shot_target_perc xGoal
## 1 Elijah Adebayo       FW    Luton Town  12.9     9             42.9   5.6
## 2  Simon Adingra       FW      Brighton  19.6     6             45.0   3.5
## 3 Miguel Almirón       FW Newcastle Utd  20.5     3             27.5   4.5
```

```
attack_stats <-
  attack_stats %>%
  inner_join(player_wages, by = c("name","position","team"))

head(attack_stats,3)
```

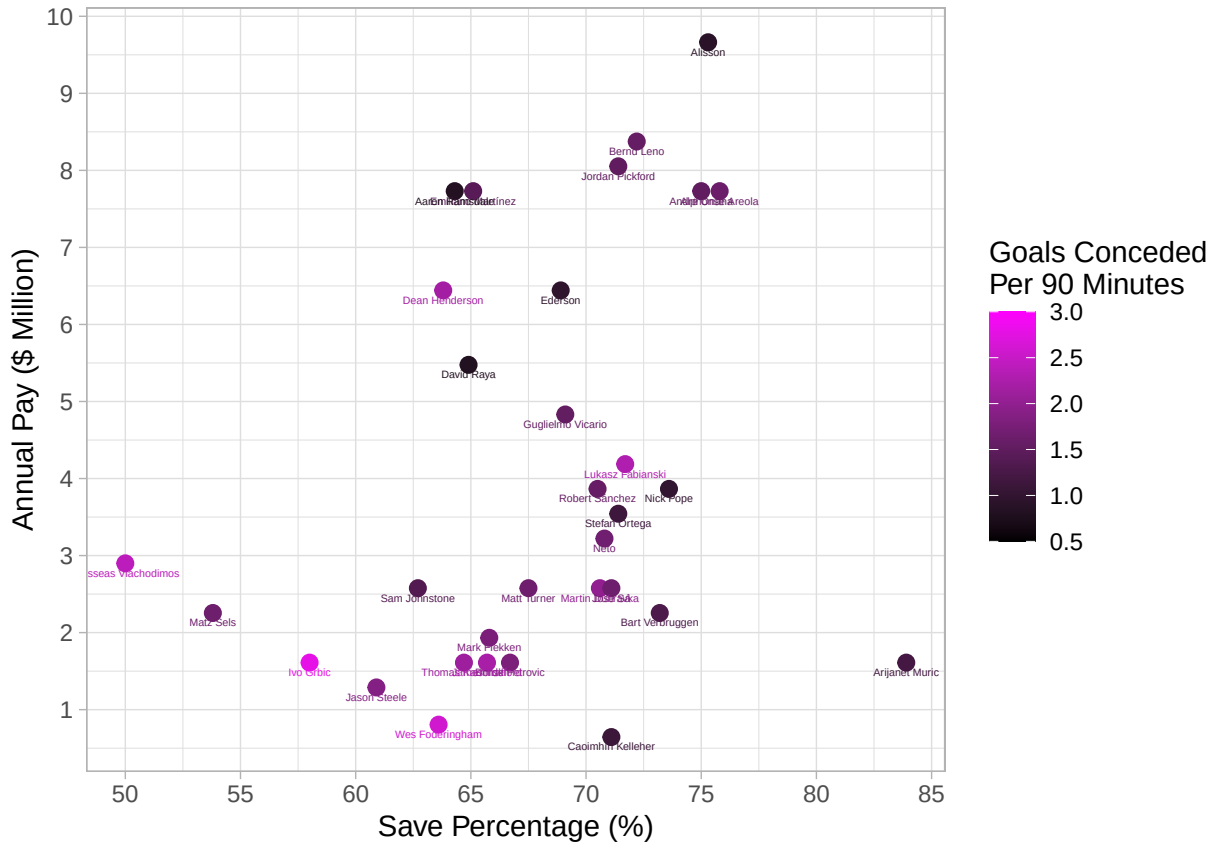
```
##             name position          team games goals shot_target_perc xGoal
## 1 Elijah Adebayo       FW    Luton Town  12.9     9             42.9   5.6
## 2  Simon Adingra       FW      Brighton  19.6     6             45.0   3.5
## 3 Miguel Almirón       FW Newcastle Utd  20.5     3             27.5   4.5
##   pay_year
## 1   805248
## 2   805248
## 3  3865192
```

Data Visualization

GK plot

```
ggplot(goalkeeper_stat) +
  aes(x = save_percentage,y = pay_year/1000000,colour = goal_against_per90,label = name) +
  geom_point(shape = "circle", size = 2.5) +
```

```
  scale_color_gradient(limits= c(0.5,3),low = "black",high = "magenta",name = "Goals Conceded \nPer 90 M
  scale_y_continuous(breaks = seq(0, 10, 1)) +
  scale_x_continuous(breaks = seq(50,85,5)) +
  geom_text(size = 1.5, vjust = 1.75) +
  ylab("Annual Pay ($ Million)") +
  xlab("Save Percentage (%)") +
  theme_light()
```



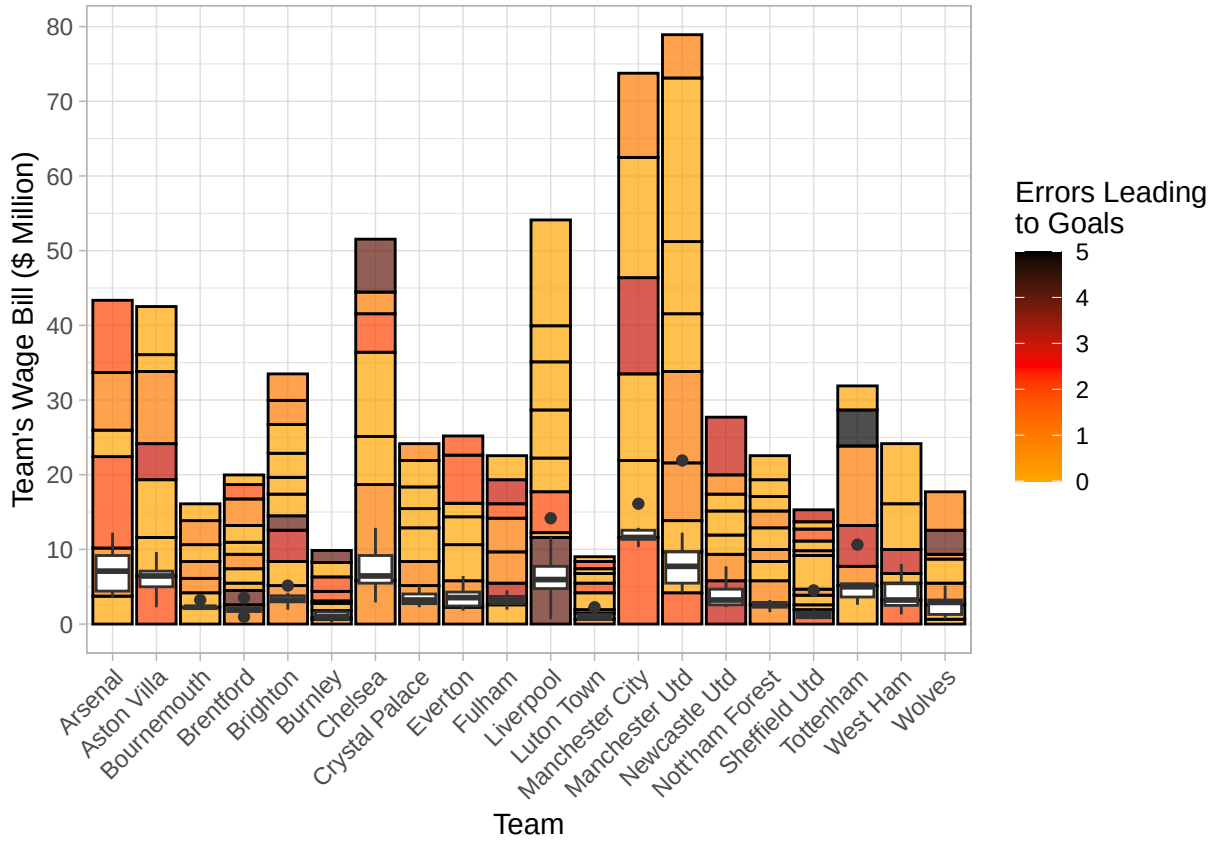
Through this plot we can see the relationship between a goalkeepers save percentage and their annual pay in millions of dollars. We also have the dots colored by the amount of goals each goalkeeper concedes per 90 minutes. We can see a slight positive linear trend in this data meaning that typically the higher paid players will have a higher save percentage. We also see that there are more pink colored points in the bottom left quarter of the graph. This means that players that are paid less and have a lower save percentage might concede more goals. This means shows that there is a relationship between the goalkeepers performance(save percentage/goals conceded) and their pay.

DF Plot

```
ggplot(defender_stats) +
  aes(x = team, y = pay_year/1000000, fill = errors) +
  geom_col(color = "black", alpha = 0.7) +
  scale_fill_gradientn(name = "Errors Leading \nto Goals",colors = c("orange", "red", "black")) +
  scale_y_continuous(breaks = seq(0,80,10)) +
  ylab("Team's Wage Bill ($ Million)") +
```

```
  xlab("Team") +
  geom_boxplot(aes(x=team,y=pay_year/1000000)) +
  theme_light() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

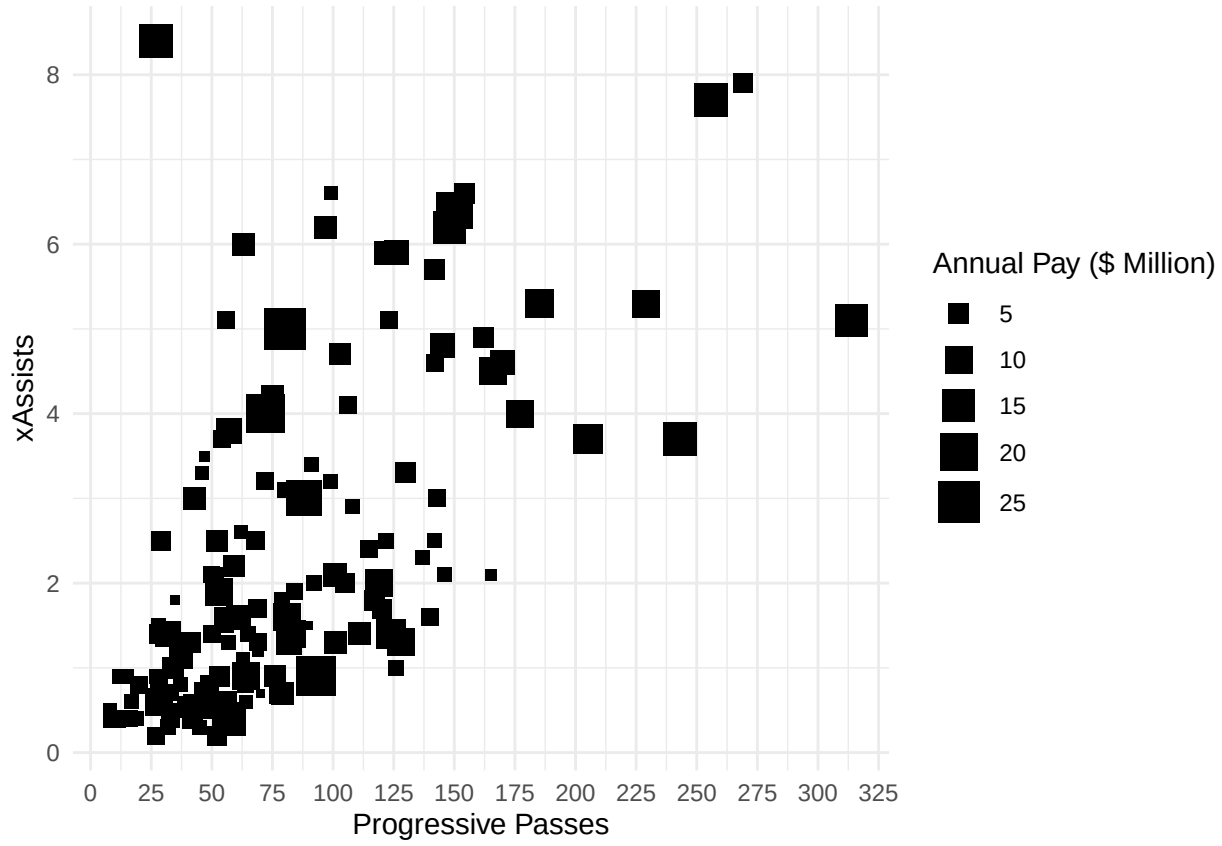
```
## Warning: The following aesthetics were dropped during statistical transformation: fill
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a 'group' aesthetic or to convert a numerical
##   variable into a factor?
```



This plot is a bar chart with the teams on the x-axis and a team's wage bill for defenders plotted on the y-axis. Each separate box on the bars is a player and how much of the team's wages that they make up. We can see that the color of each individual section represents the errors leading to goals that they have this season. There is also a boxplot that overlays each bar which is used to show the variation in salary between players of the same team. We can see that teams like Crystal Palace and Bournemouth have the least errors this season. If we look at the boxplot for these teams we can see that the plot is very small, meaning there is little variation between each players salary. On the other end, there is Liverpool. We see that the box plot is rather spread out, meaning that players are making different salaries. With this information, we can look at the barchart to see that one of their most expensive players has contributed to 4 errors. In the end, we can see that pay does make an impact on a teams ability to defend, but not in the expected manner. Since defending is a team activity, we need everyone to be happy with their salary. My findings have shown that teams have less errors if they pay their players close to the same amount. This shows that salary can affect the performance of players in terms of defending.

MF Plot

```
ggplot(midfield_stats) +
  aes(x = prog_pass,y = xAssists) +
  geom_point(shape = "square",aes(size = pay_year/1000000)) +
  scale_size_continuous(range = c(0.5, 7), name = "Annual Pay ($ Million)") +
  scale_y_continuous(breaks = seq(0,10,2)) +
  scale_x_continuous(breaks=seq(0,325,25))+
  xlab("Progressive Passes") +
  theme_minimal()
```

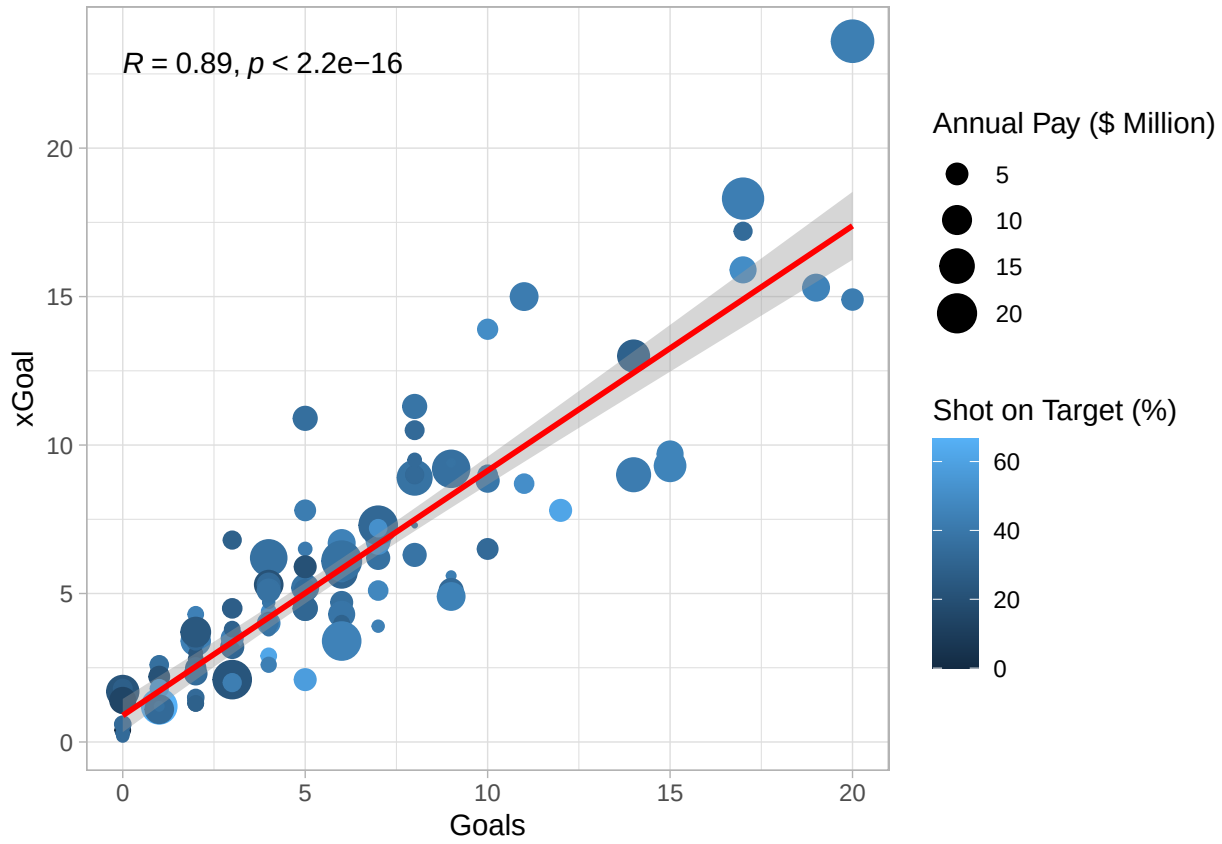


In this plot we can see the relationship between progressive passes and expected assists. We can see that there is a slight linear relation between these two variables. Progressive passes are when a player passes the ball in a forward direction. Every pass that a player makes is given an expected assist value based on how likely that pass is to turn into a goal. As we can see, there is a clear relation between the two; this means that the more forward passes a player makes, the more assists are expected to come from this player. The size of the points is also the pay of the player. We can see that a lot of the players that have a large size, typically have more progressive passes. Through the relation I determined between progressive passes and expected assists, these higher paid players can also be seen to have a high expected assists. However, there are some large points located towards the bottom left of the graph. There is an explanation for this. Some players have multiple positions listed on their profiles. So a player that has the "MF,DF" or "DF,MF" position, plays both midfield and defence. The amount of progressive passes and expected assists a player has can be limited because they might be playing as a defender at certain times. So, accounting for multiple positions, midfielders most certainly have their performance affected by how well they are paid.

FW Plot

```
ggplot(attack_stats) +
  aes(x = goals, y = xGoal, colour = shot_target_perc) +
  geom_point(shape = "circle", aes(size = pay_year/1000000)) +
  scale_size_continuous(range = c(0.5, 7), name = "Annual Pay ($ Million)") +
  scale_color_gradient(limits = c(0, max(attack_stats$shot_target_perc)), name = "Shot on Target (%)") +
  stat_cor(method = "spearman") +
  scale_x_continuous(breaks = seq(0, 20, 5)) +
  scale_y_continuous(breaks = seq(0, 25, 5)) +
  xlab("Goals") +
  stat_smooth(method = "lm", formula = y ~ x, geom = "smooth", color = "red") +
  theme_light()
```

```
## Warning: The following aesthetics were dropped during statistical transformation: colour
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a 'group' aesthetic or to convert a numerical
##   variable into a factor?
```



This graph plots goals vs expected goals. Obviously, there is a linear relationship between goals and expected goals. Obviously there is a clear relationship between goals and expected goals, but the relationship does not exactly follow the line y=x, it is slightly less steep which means that players are scoring less goals than they are expected to on average. The correlation of this graph is 0.89 which is a strong positive linear relation and the p-value is extremely small which means that this correlation is statistically significant. For

the color of the graph, we see that the darker colors are more concentrated in the bottom left. This means that players that put less of their shots on target, typically have a lower amount of goals and expected goals. We can also see that there are some very large points in the top right, these players are highly paid because of the size of the points. They also are significantly above the correlation line, which means that they are expected to score more goals than they do. This shows that higher paid players are more likely to shoot the ball accurately and create more scoring opportunities.

Conclusion

Overall, I found this research to be very insightful. I found that the pay of players does affect the performance on the field, but not in the way that I originally thought. I found that for some positions on the pitch, individual brilliance is more important, but in other positions it matters more to be a team. I found that for goalkeepers, midfielders, and attackers, that their pay directly correlates to their performances. This is obvious for goalkeepers as there is only one of them per team. But for attackers and midfielders, it is also dependent on individuals because things like passing or shooting are largely matters of individual talent. So in simple terms, the more talented a player is, the more likely they are to get paid and perform well. But I found there to be a different narrative when it came to defenders. I found that defences tend to perform better when all members of a team are paid roughly the same amount. This might be because defending in soccer is not an individual thing, the entire team is required to defend well. I also faced a significant challenge in my research. As I stated in the data wrangling section, I had trouble reading my table from the website because the table was not directly in the HTML code. I solved this issue by using the read_html_live() function and also copy and pasting the data into a csv file. The html live function is hit or miss because of the timeout being set to 10 seconds by default. If you want to try to run it using this function, you can comment out all the read.csv lines and uncomment any lines of code throughout.