Final Assignment
Football dataset analysis

Universitat Autònoma de Barcelona

Genís Ruiz Menárguez

1633623

Index

1.Introduction

The goal of the assignment is to create a model for predicting the football players value based on different inputs such as the number of goals, assists and minutes played, their team or age. Then, analysis techniques will be used to check the goodness of the fit. Moreover, the goal is also to explore other possibilities and comparatives with other parameters such as the number of goals and compare the impact of the parameters in the Top5 football Leagues in Europe, especially in *La Liga* and *Premier League.*

2.Description of the dataset

The original dataset consists of 400 columns of parameters and stats of all the players from the Top5 football leagues[1] in the season 2019-2020. The original dataset was extracted from Kaggle[2].

Nevertheless, to simplify the analysis of the dataset, some columns were removed. The main criteria followed was to remove all the parameters which contained general team information, such as matches won, lost, etc., which were a lot. Then, all the data which came from other basic parameters was also removed. For instance, the goals per minute or goals in the first half are derivatives from the parameter goal or minutes, so they were eliminated. Moreover, there remained many columns for too few rows, so the correlation between the response variable called *value* and the other columns was studied. The ones which had a high correlation remained, the others were dismissed.

The number of rows of the modified dataset is the same as the original one, just 2644. Each row is a football player for a certain Top5 league team.

Therefore, the modified dataset, from now on called *dataset*, contains 15 columns:

- **player**: It contains the name and surname of the player.

- **nationality**: It contains the name of the national team which the player plays for.

- **position**: It contains the position of the player (GK, DF, MF or FW)[3]. Some values refer to two positions, ordered by player's preference.

- **squad**: Name of the team they play for.

- **age**: Integer number naming the age.

- **value**: Player's market value in millions of euros.

- **height**: Player's height in centimeters.

- **league**: Name of one of the 5 leagues.

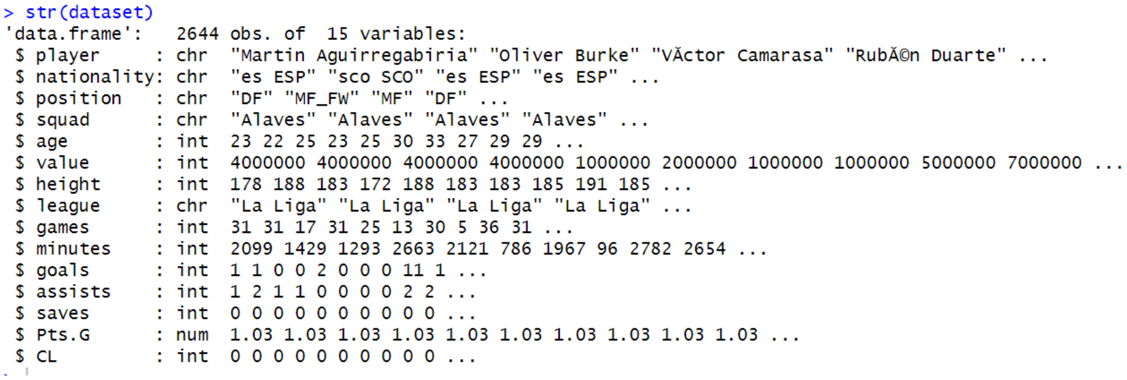
- **games**: Number of games played.

-**minutes**: Number of minutes played.

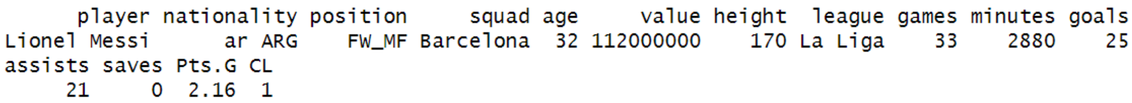
[1] *The Football Top5 Leagues are: La Liga (Spain), Premier League (England + Wales), Serie A (Italy), Bundesliga (Germany) and Ligue 1 (France)*
[2] *Find the original dataset in the bibliography*
[3] *Goalkeeper, Defender, Midfielder and Forward.*

- **Goals**: Number of goals scored.
- **Assists**: Number of assists done.
- **Saves**: Number of saves done (only for goalkeepers.)
- **Pts/G**: Average of squad points per game played.
- **CL**: 0 or 1 depending if the player has competed in Champions League this season.

Some examples of the different values:



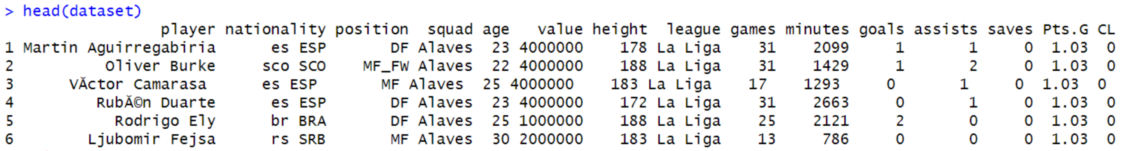
```
> str(dataset)
'data.frame':   2644 obs. of  15 variables:
 $ player     : chr  "Martin Aguirregabiria" "Oliver Burke" "VÃctor Camarasa" "RubÃ©n Duarte" ...
 $ nationality: chr  "es ESP" "sco SCO" "es ESP" "es ESP" ...
 $ position   : chr  "DF" "MF_FW" "MF" "DF" ...
 $ squad      : chr  "Alaves" "Alaves" "Alaves" "Alaves" ...
 $ age        : int  23 22 25 23 25 30 33 27 29 29 ...
 $ value      : int  4000000 4000000 4000000 4000000 1000000 2000000 1000000 1000000 5000000 7000000 ...
 $ height     : int  178 188 183 172 188 183 183 185 191 185 ...
 $ league     : chr  "La Liga" "La Liga" "La Liga" "La Liga" ...
 $ games      : int  31 31 17 31 25 13 30 5 36 31 ...
 $ minutes    : int  2099 1429 1293 2663 2121 786 1967 96 2782 2654 ...
 $ goals      : int  1 1 0 0 2 0 0 0 11 1 ...
 $ assists    : int  1 2 1 1 0 0 0 0 2 2 ...
 $ saves      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Pts.G      : num  1.03 1.03 1.03 1.03 1.03 1.03 1.03 1.03 1.03 1.03 ...
 $ CL         : int  0 0 0 0 0 0 0 0 0 0 ...
```



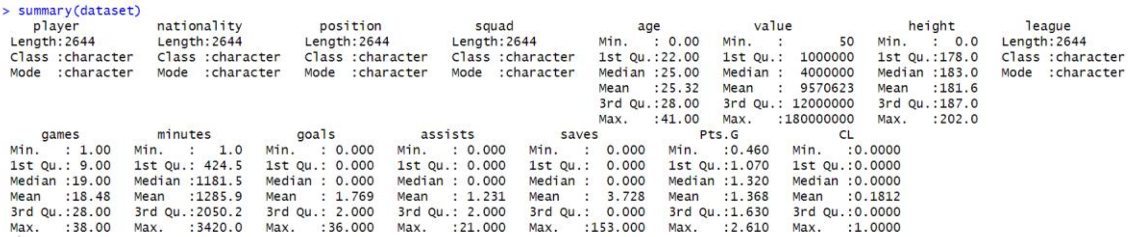
```
       player nationality position     squad age     value height  league games minutes goals
Lionel Messi       ar ARG    FW_MF Barcelona  32 112000000    170 La Liga    33    2880    25
assists saves Pts.G CL
     21     0  2.16  1
```

Lionel Messi stats in the season 2019-2020.

To understand widely how the dataset works, some basic data representation will be provided.



```
> head(dataset)
                 player nationality position  squad age   value height  league games minutes goals assists saves Pts.G CL
1 Martin Aguirregabiria      es ESP       DF Alaves  23 4000000    178 La Liga    31    2099     1       1     0  1.03  0
2           Oliver Burke    sco SCO    MF_FW Alaves  22 4000000    188 La Liga    31    1429     1       2     0  1.03  0
3        VÃctor Camarasa      es ESP       MF Alaves  25 4000000    183 La Liga    17    1293     0       1     0  1.03  0
4          RubÃ©n Duarte      es ESP       DF Alaves  23 4000000    172 La Liga    31    2663     0       1     0  1.03  0
5           Rodrigo Ely      br BRA       DF Alaves  25 1000000    188 La Liga    25    2121     2       0     0  1.03  0
6       Ljubomir Fejsa      rs SRB       MF Alaves  30 2000000    183 La Liga    13     786     0       0     0  1.03  0
```



```
> summary(dataset)
    player           nationality          position            squad                age            value               height          league         
 Length:2644        Length:2644        Length:2644        Length:2644        Min.   : 0.00   Min.   :       50   Min.   :  0.0   Length:2644       
 Class :character   Class :character   Class :character   Class :character   1st Qu.:22.00   1st Qu.:  1000000   1st Qu.:178.0   Class :character  
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Median :25.00   Median :  4000000   Median :183.0   Mode  :character  
                                                                              Mean   :25.32   Mean   :  9570623   Mean   :181.6                     
                                                                              3rd Qu.:28.00   3rd Qu.: 12000000   3rd Qu.:187.0                     
                                                                              Max.   :41.00   Max.   :180000000   Max.   :202.0                     
     games          minutes           goals            assists            saves              Pts.G             CL        
 Min.   : 1.00   Min.   :   1.0   Min.   : 0.000   Min.   : 0.000   Min.   :  0.000   Min.   :0.460   Min.   :0.0000  
 1st Qu.: 9.00   1st Qu.: 424.5   1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.:  0.000   1st Qu.:1.070   1st Qu.:0.0000  
 Median :19.00   Median :1181.5   Median : 0.000   Median : 0.000   Median :  0.000   Median :1.320   Median :0.0000  
 Mean   :18.48   Mean   :1285.9   Mean   : 1.769   Mean   : 1.231   Mean   :  3.728   Mean   :1.368   Mean   :0.1812  
 3rd Qu.:28.00   3rd Qu.:2050.2   3rd Qu.: 2.000   3rd Qu.: 2.000   3rd Qu.:  0.000   3rd Qu.:1.630   3rd Qu.:0.0000  
 Max.   :38.00   Max.   :3420.0   Max.   :36.000   Max.   :21.000   Max.   :153.000   Max.   :2.610   Max.   :1.0000  
```

From the summary we can see several things:

-The ages' mean of the Top 5 leagues is 25.32, as it is the best physical moment for a human being.

-A 25% of the players have their value under 1 million€

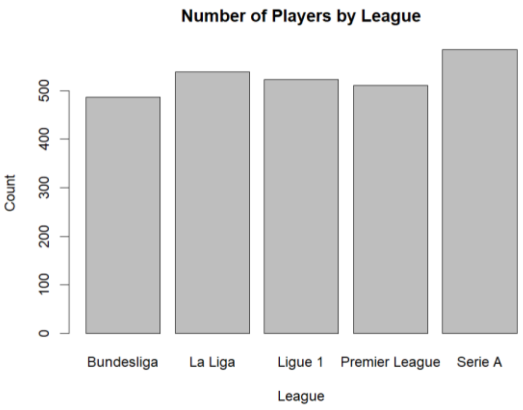
-The 50% of the players play at least 19 games, the 50% of the league games.

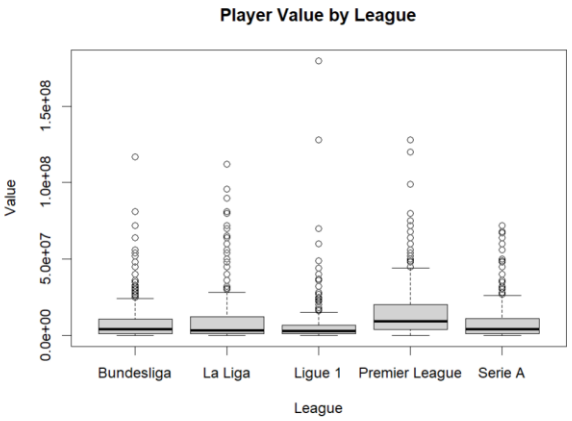
-The most part of the players didn't score a goal in the 19-20 season

-The stats about saves are useless as it includes all types of position and not only GK.

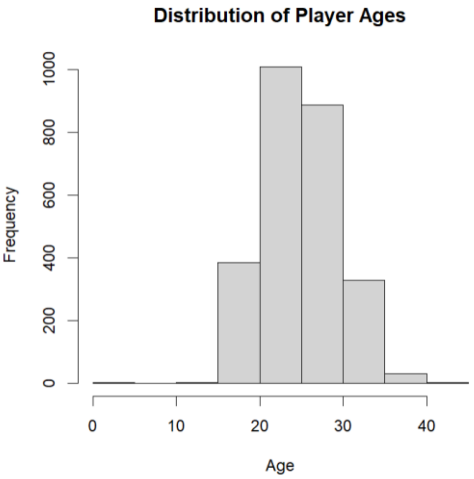
-The average of Points per game in a player are 1.368. The maximum average is 2.61, from Liverpool.

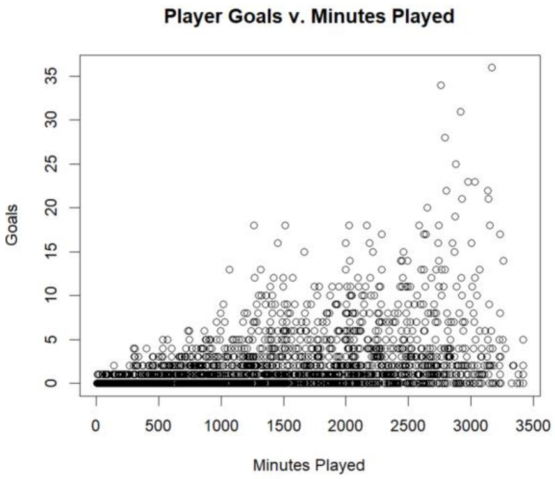
-Only a 18.12% of players are in a team which has played in the Champions League.



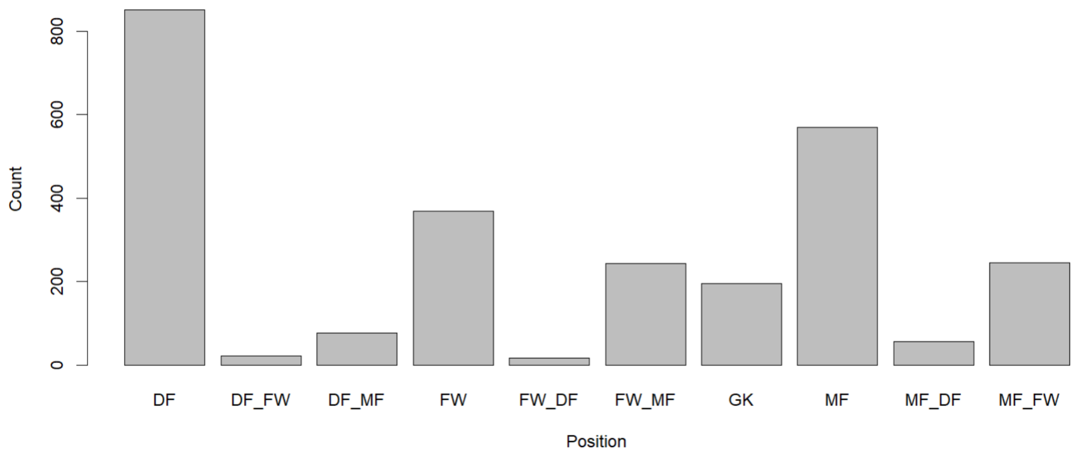


From the second graphic it is observable that Premier League players have a greater value than other leagues. There is also a player from the Ligue 1 which has the highest value, which is Mbappé with a worth value of 180M.



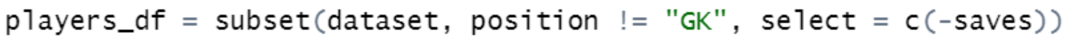


Regarding player's age, the most populated interval is from the 20 years to the 30 years. Additionally, the second graphic plots the relation between the minutes played and the goals scored. It is obvious that the more you play, the more possibilities you have to score, so it is observable an increasing tendency in the graphic.



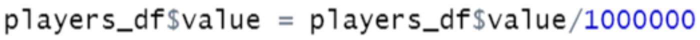
This graphic provides the number of players of each position. Notice The great amount of defenders and the small amount of Goalkeepers. Players who can play in two different positions will be considered in both positions in the following analysis.

As most of GK do not score goals or assist and are the only ones who make saves, a new dataset will be created with the purpose of predicting more accurately the field players[4] values, from now on, *players_df*.



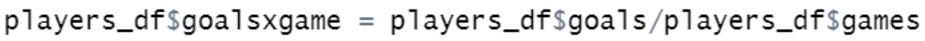
```
players_df = subset(dataset, position != "GK", select = c(-saves))
```

Because of the value will be the variable to study and predict, it is better to rescale its value. As the players' values are big numbers, dividing them by 1Milion will let us study the parameters and coefficients more easily.



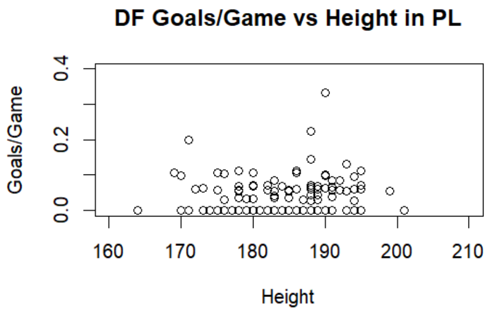
```
players_df$value = players_df$value/1000000
```

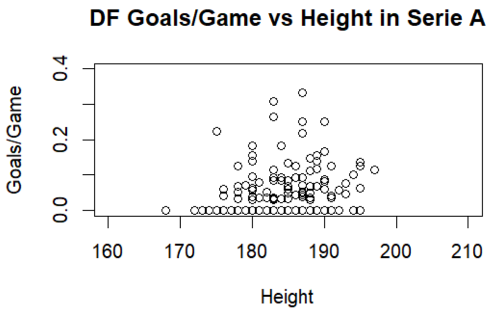
Let's continue the description of the dataset by adding a new column to the model. A column containing goals per game will facilitate the study of the future model and provide more comprehensive results. Thus, here it is shown the variable creation.



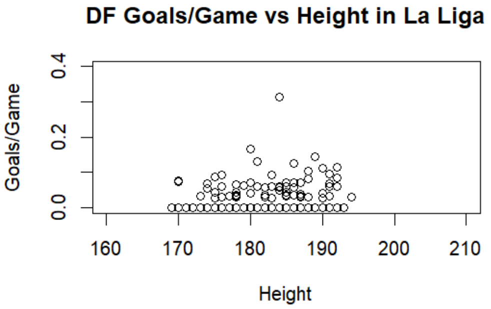
```
players_df$goalsxgame = players_df$goals/players_df$games
```

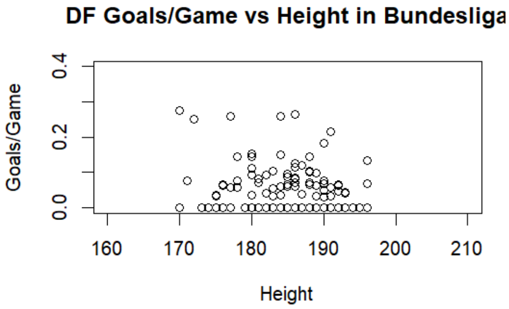
To end up the dataset's description, let's study the relation of *height* and *goalsxgame* considering the defenders of 4 leagues.





[4] *Field player: They are players who are not goalkeepers.*





As it is shown, most of the defenders have a height between 170 cm and 195 cm. In Serie A and Bundesliga there are more goals per game coming from defenders. However, in La liga there is a lower average. Is the defenders' height related to the *goalsxgame*? Is this low average in La Liga related to a lower mean of goals scored than in other leagues? In the section 4 the *goalsxgame* between La Liga and Premier League will be compared.

3.Analyzing techniques

Several analyzing techniques have been applied in the project. Some of them were applied to get a better fit of the linear regression and other were used to study some residuals or parameter means.

- **Correlation with the response variable:** It is the level of linear relationship between a variable and the one you want to predict. The value is a number contained in the [0,1] interval and the greater it is, the greater relation both variables have. If the value is 0, it means there is no dependency.

- **Linear regression:** Is a technique used for predicting data. The variable to predict is the dependent or response variable, and the others are the independent ones. The goal is to estimate the coefficients of these variables to perform a better prediction. As it is a linear regression, it creates a prediction line which fits the best way the observed data.

- **Backward Selection with AIC and p-value criteria:** It is a criterion for the selecting the best independent variables to get the best fit. The backward selection is the one which starts with a linear regression full of variables and consists in removing variables till the best fit is obtained. On the other hand, the AIC criterion minimizes the independent variables and maximizes the amount of variation in the model. For calculating this criterion, the number of independent variables (K) and the Loglikelihood of the model (L) are needed.

$$\mathrm{AIC} = 2\mathrm{K} - 2\ln(\mathrm{L})$$

The best AIC is the one which has the maximum value. In R, there is the library *MASS* which includes *stepAIC()* function which allows us to perform the criterion step by step applying different selections.

Applying backward selection with p-value criteria is a similar process but the selected and removed variables are the ones which have a greater p-value than α. The p-value of the regression variables is the probability to accepting H0: $\beta = 0$ where β is any coefficient. If the p-value is lower than α, it means that variable has impact on the response one.

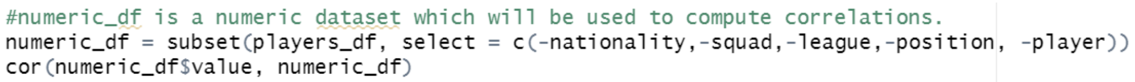
- **Predicting from linear regression and comparing the results:** It consists in creating some new inputs for the independent variables with the purpose of getting a prediction with the response one.

- **Non-parametric Bootstrapping for comparing some means:** It is a technique which uses the observed data to create new samples. *K* samples are created, each one of the observed data size *n*. These samples are created from selecting the observed data values with replacement. For each sample, estimators such as the mean, the standard deviation or the median can be computed and saved into a vector. Then, these estimators can be compared to the observed ones, or plotted into a histogram to observe the normal distribution. The larger the *K* is, the better approximation it is obtained.

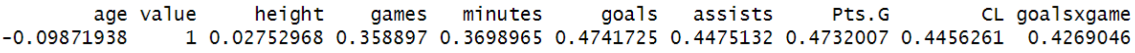
- **Parametric Bootstrapping for estimating the residuals of the fit:** The process is similar to the non-parametric one, but it considers a certain distribution of the data, so there is no replacement. Some examples of distributions are the Normal, the Poisson or the Exponential. An application of parametric bootstrapping may be estimating the sigma of the residuals of a linear regression fit, as residuals follow a Normal in LR[5].

4.Analysis and Results of the dataset

Before computing the linear regression, response variable correlations with other numeric columns will be studied. As it was explained before, the main goal was to obtain a good prediction for the players' values, so let's see its correlations.



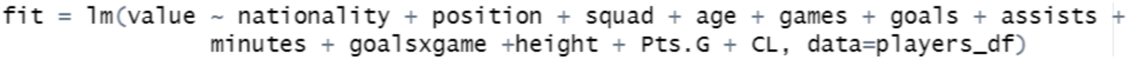
```
#numeric_df is a numeric dataset which will be used to compute correlations.
numeric_df = subset(players_df, select = c(-nationality,-squad,-league,-position, -player))
cor(numeric_df$value, numeric_df)
```



```
        age value     height    games   minutes     goals   assists     Pts.G        CL goalsxgame
-0.09871938     1 0.02752968 0.358897 0.3698965 0.4741725 0.4475132 0.4732007 0.4456261  0.4269046
```

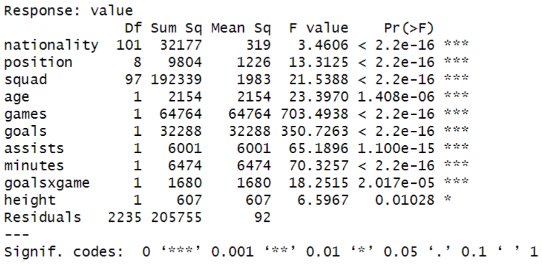
From these correlations we see that each one is different from 0, meaning there is a relation between the player's value and every column. Nevertheless, height and age present low correlations. As both parameters seem to be meaningful to predict values, other statistical studies will be performed. A linear regression will be computed, looking for the best fit for the model.

Let's continue by creating the first linear regression model.



```
fit = lm(value ~ nationality + position + squad + age + games + goals + assists +
                minutes + goalsxgame +height + Pts.G + CL, data=players_df)
```

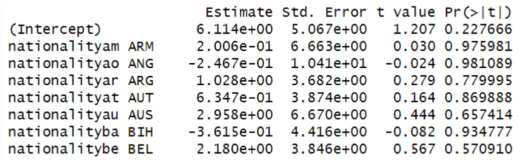
Analysis of variance:



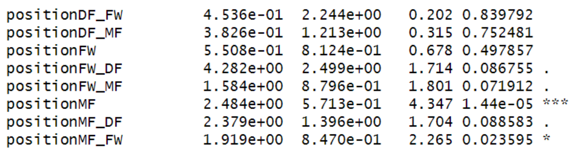
```
Response: value
              Df Sum Sq Mean Sq  F value    Pr(>F)    
nationality  101  32177     319   3.4606 < 2.2e-16 ***
position       8   9804    1226  13.3125 < 2.2e-16 ***
squad         97 192339    1983  21.5388 < 2.2e-16 ***
age            1   2154    2154  23.3970 1.408e-06 ***
games          1  64764   64764 703.4938 < 2.2e-16 ***
goals          1  32288   32288 350.7263 < 2.2e-16 ***
assists        1   6001    6001  65.1896 1.100e-15 ***
minutes        1   6474    6474  70.3257 < 2.2e-16 ***
goalsxgame     1   1680    1680  18.2515 2.017e-05 ***
height         1    607     607   6.5967   0.01028 *  
Residuals   2235 205755      92                       
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

[5] *LR: Abbreviation of 'linear regression'*

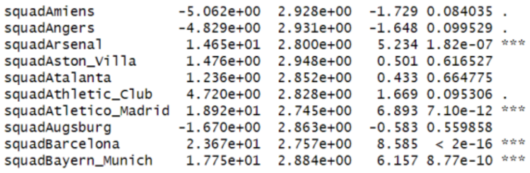
Summary of the coefficients:



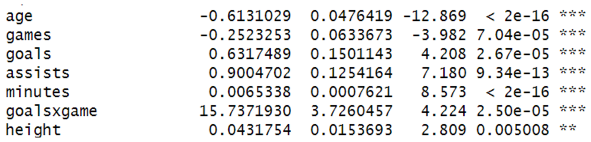
```
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)     6.114e+00  5.067e+00   1.207 0.227666
nationalityam ARM  2.006e-01  6.663e+00   0.030 0.975981
nationalityao ANG -2.467e-01  1.041e+01  -0.024 0.981089
nationalityar ARG  1.028e+00  3.682e+00   0.279 0.779995
nationalityat AUT  6.347e-01  3.874e+00   0.164 0.869888
nationalityau AUS  2.958e+00  6.670e+00   0.444 0.657414
nationalityba BIH -3.615e-01  4.416e+00  -0.082 0.934777
nationalitybe BEL  2.180e+00  3.846e+00   0.567 0.570910
```



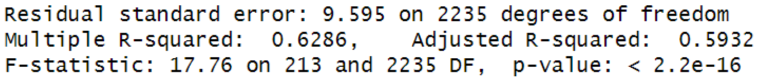
```
positionDF_FW     4.536e-01  2.244e+00   0.202 0.839792
positionDF_MF     3.826e-01  1.213e+00   0.315 0.752481
positionFW        5.508e-01  8.124e-01   0.678 0.497857
positionFW_DF     4.282e+00  2.499e+00   1.714 0.086755 .
positionFW_MF     1.584e+00  8.796e-01   1.801 0.071912 .
positionMF        2.484e+00  5.713e-01   4.347 1.44e-05 ***
positionMF_DF     2.379e+00  1.396e+00   1.704 0.088583 .
positionMF_FW     1.919e+00  8.470e-01   2.265 0.023595 *
```



```
squadAmiens         -5.062e+00  2.928e+00  -1.729 0.084035 .
squadAngers         -4.829e+00  2.931e+00  -1.648 0.099529 .
squadArsenal         1.465e+01  2.800e+00   5.234 1.82e-07 ***
squadAston_Villa     1.476e+00  2.948e+00   0.501 0.616527
squadAtalanta        1.236e+00  2.852e+00   0.433 0.664775
squadAthletic_Club   4.720e+00  2.828e+00   1.669 0.095306 .
squadAtletico_Madrid 1.892e+01  2.745e+00   6.893 7.10e-12 ***
squadAugsburg       -1.670e+00  2.863e+00  -0.583 0.559858
squadBarcelona       2.367e+01  2.757e+00   8.585  < 2e-16 ***
squadBayern_Munich   1.775e+01  2.884e+00   6.157 8.77e-10 ***
```



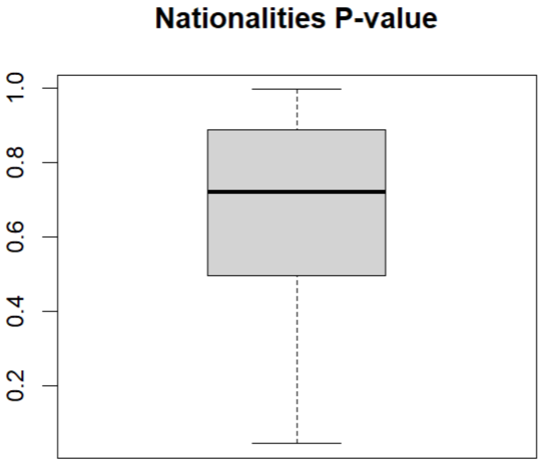
```
age           -0.6131029  0.0476419 -12.869  < 2e-16 ***
games         -0.2523253  0.0633673  -3.982 7.04e-05 ***
goals          0.6317489  0.1501143   4.208 2.67e-05 ***
assists        0.9004702  0.1254164   7.180 9.34e-13 ***
minutes        0.0065338  0.0007621   8.573  < 2e-16 ***
goalsxgame    15.7371930  3.7260457   4.224 2.50e-05 ***
height         0.0431754  0.0153693   2.809 0.005008 **
```



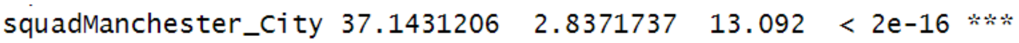
```
Residual standard error: 9.595 on 2235 degrees of freedom
Multiple R-squared:  0.6286,	Adjusted R-squared:  0.5932 
F-statistic: 17.76 on 213 and 2235 DF,  p-value: < 2.2e-16
```

Firstly, notice there are some nationalities and squads missing, as they are a lot. These variables and position ones are called *dummy variables.* They are binary and they are created when a column has non-numeric values. The CL and Pts.G variables have collinearity with other variables, so the result in the summary is NA.

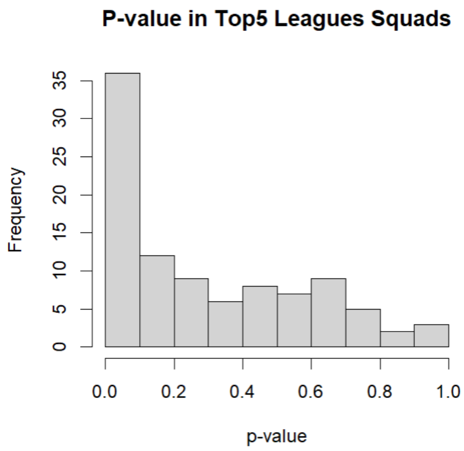
- From the summary function, we can see that the nationality has not a great impact in the regression, as there is not any nationality which has a lower p-value than 0.1. In the following graphic, p-values of different nationalities will be plotted.

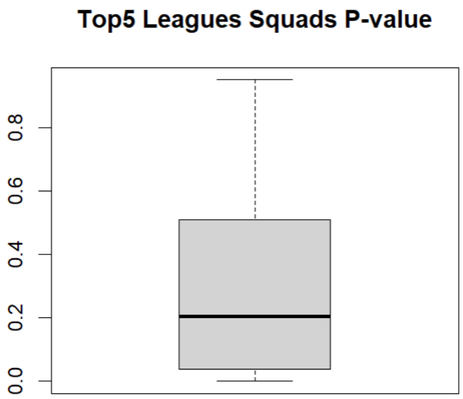


- Is is also observable that the bigger the club is, the more valuables players it has. For example, FC Barcelona or Bayern Munich have a tiny p-value. Teams such as Manchester City, which make huge inversions in players every summer, have the tiniest p-values, meaning that belonging to these clubs has a huge repercussion while predicting player's value.



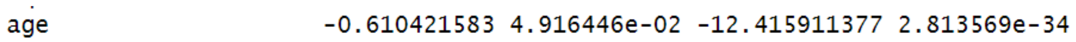
```
squadManchester_City 37.1431206  2.8371737  13.092  < 2e-16 ***
```





Players' squads must be considered in the linear regression, as there are great amount of them with a tiny p-value.

- Regarding age, it is visible that the age coefficient is negative, meaning the older a player is, the cheaper it costs.



```
age                         -0.610421583 4.916446e-02 -12.415911377 2.813569e-34
```

- Besides, fixing the parameters and increasing the number of games played will reduce the player's cost.

- In addition, *goalsxgame* has the greatest coefficient positive value, as it implies a great correlation with the response variable

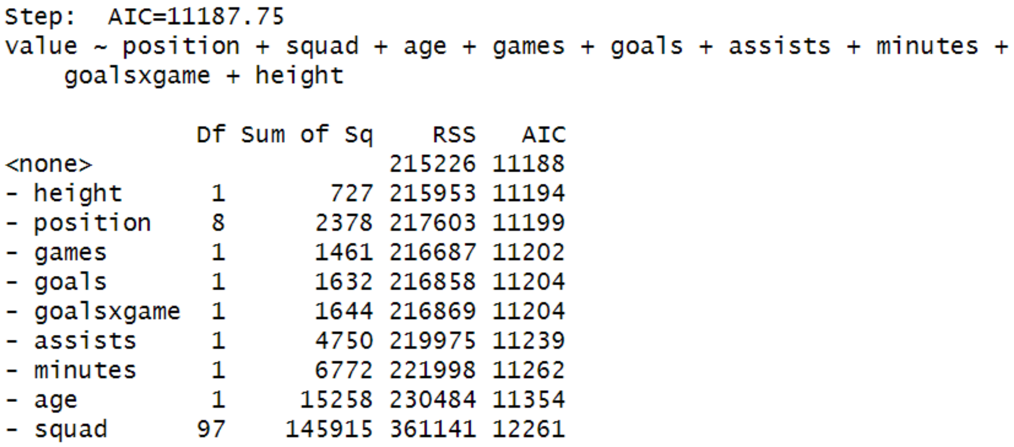
- Moreover, parameters such as the assists, minutes and the height are determinant too.

- Nevertheless, Pts.G or CL result in a NA, even not containing missing values.

A new model must be created, with just the reliable variables Backward selection by AIC will be applied since it is pretended to minimize the number of variables used (as there is few observed data) and maximize the goodness of the fit.

BACKWARD SELECTION CRITERIA:

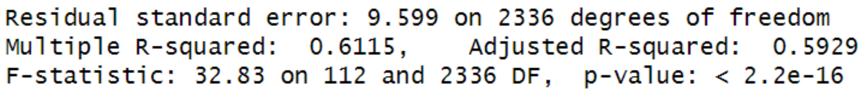
Backward selection will be applied to realize which is the best possible fit of the parameters following AIC[6]. The resulting fit is this one:



```
Step:  AIC=11187.75
value ~ position + squad + age + games + goals + assists + minutes +
    goalsxgame + height

             Df Sum of Sq    RSS   AIC
<none>                    215226 11188
- height      1       727 215953 11194
- position    8      2378 217603 11199
- games       1      1461 216687 11202
- goals       1      1632 216858 11204
- goalsxgame  1      1644 216869 11204
- assists     1      4750 219975 11239
- minutes     1      6772 221998 11262
- age         1     15258 230484 11354
- squad      97    145915 361141 12261
```

The optimal fit following AIC is the one which minimizes it. The variables *CL*, *Pts.G* and *nationality* have been removed. Removing more variables would increase AIC value, as it is shown.

These are the values of some useful estimators:



```
Residual standard error: 9.599 on 2336 degrees of freedom
Multiple R-squared:  0.6115,    Adjusted R-squared:  0.5929
F-statistic: 32.83 on 112 and 2336 DF,  p-value: < 2.2e-16
```

Regarding new model estimators, there have slight differences with the previous ones. The most important changes are found in the F-statistic. A greater F-Statistic value implies a better fit of the data, as it describes more accurately the variation of the dependent variable. In the current model this value is 32.83, and in the previous one it was 17.16. Clearly, the optimized model explains better this variation. The p-value is almost 0, so the F-statistic has verified the goodness of the fit.

Once the fit is optimized, it is time to predict.

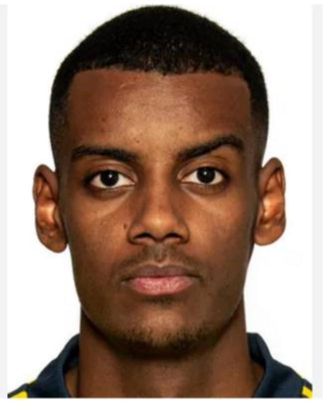
PREDICTIONS

This step consists in focusing on 4 cases of players and check if the predictions work well.

Reminder*: these stats are from the 2020-2021 league season, one year after the dataset.*

[6] *AIC: Criterion explained in section 3: Analyzing Techniques*

Case 1: Alexander Isak.



Position: FW

Squad: Real Sociedad (La Liga)

Age: 21 (In 2021)

Games: 34

Goals: 17

Assists: 2

Minutes: 2361

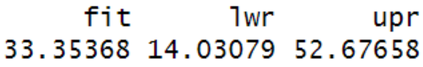
Goals per game: 0.49

Height: 192 cm

Market value in 2020: 22M[7]

Market value in 2021: 30-40M[8]

Predicted value for 2021:



```
     fit      lwr      upr
33.35368 14.03079 52.67658
```

The predicted value fits with the real value. The error of the prediction is about 4.7%. The 95% Confidence Interval of the prediction is not narrow, as it is a young and a goal scorer player.

Case 2: Germán Pezzella.

Position: DF

Squad: AFC Fiorentina (Serie A)

Age: 29 (In 2021)

Games: 32

Goals: 1

Assists: 2

Minutes: 2748

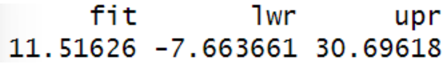
Goals per game: 0.032

Height: 187 cm

Market value in 2020: 14M

Market value in 2021: 7.5-10M

Predicted value for 2021:



```
     fit       lwr      upr
11.51626 -7.663661 30.69618
```

[7] *Value from the kaggle dataset*

[8] *Values from Transfermarkt, a reliable football web source.*

The predicted value is like the market value. The error is about 20%. The 95% Confidence Interval contains negative values, but they do not have economic sense.

Case 3: Isco Alarcón.

Position: MF

Squad: Real Madrid (La Liga)

Age: 29 (In 2021)

Games: 25

Goals: 0

Assists: 2

Minutes: 898

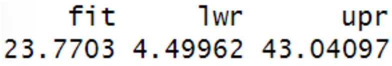
Goals per game: 0.00

Height: 176 cm

Market value in 2020: 20M

Market value in 2021: 18M

Predicted value for 2021:



```
    fit     lwr      upr
23.7703 4.49962 43.04097
```

In this case the predicted value is a bit far from the market value. The error is around 32%. Isco is a low value player who has not done a great performance in the season 2020-2021. Nevertheless, as he plays in Real Madrid, one of the teams with the most valuable players, the prediction tends to overestimate his value. Thereby, it is observable that big clubs have two much influence in the model, so low valued players of top teams will not be predicted accurately.

Case 4: Wesley Fofana.

Position: DF

Squad: Leicester City (Premier League)

Age: 20 (In 2021)

Games: 28

Goals: 0

Assists: 1

Minutes: 2265

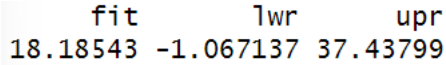
Goals per game: 0.00

Height: 186 cm

Market value in 2020: 5M

Market value in 2021: 30-40M

Predicted value for 2021:



```
     fit       lwr      upr
18.18543 -1.067137 37.43799
```

In this case the predicted value is far from the market value. The error is about 48%. The reason why is Fofana was not known the previous season, and 2020-2021 season was excellent for him. However, as he is a defense, he didn't score any goal, one of the reasons he has such a low predicted value. Also, he changed the team, he was in Saint Etienne in (Ligue 1) and went to Leicester City. Moreover, the model only considers his league stats, even though he had a great performance in FA Cup[9] and Europa League championships. Thus, the model analyzed is not usable for player who have unexpected outcomes.

NON-PARAMETRIC BOOTSTRAPPING FOR PREDICTING VALUES MEDIAN

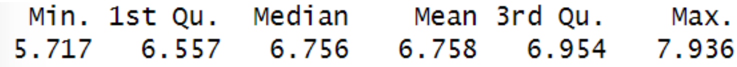
Computing the median of the players' value will allow us to obtain an estimation of the value of a medium level player.

Estimated observed median of predicted players' values: 4M

However, the median of the predicted players' value will also be performed by bootstrapping, so *k* samples with replacement from the *players_df* dataset will be considered. To each auxiliar dataset, a linear regression and prediction has been carried out. And for each prediction, the median has been computed and stored. Once there is a vector of k medians, calculating the mean will give us a more reliable result.

Reminder: *The whole code can be found in the appendix.*

These are the stats of the vector of medians for k=10,000:



```
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
5.717   6.557   6.756   6.758   6.954   7.936 
```

Mean value for non-Parametric bootstrapping for the players' values median: 6.758087M

95% CI for non-Parametric bootstrapping for the players' values median: [6.194M, 7.333M]

The mean of the vector of predicted values median named *boot_median* is 6,758,087€. The interval of minimum and maximum medians obtained is [5.717M, 7.936M], and thicker than the CI. So, the medium value of the players in the Top5 Leagues is in the interval [6.194M, 7.333M].

Other measures could be calculated, such as the standard deviation in the players' value in two different leagues to compare how spread are the values from the mean. Thus, some outliers could be detected, and it could be determined which league has the most dispersed players' values.

[9] FA Cup: One of the national trophies in England.

PARAMETRIC BOOTSTRAPPING FOR ANALYZING SIGMA

An additional application of bootstrapping in the model would be studying the residuals and the sigma from the linear regression fit.

Sigma is the Estimated Residuals Standard Error, from now on estimated RSE.

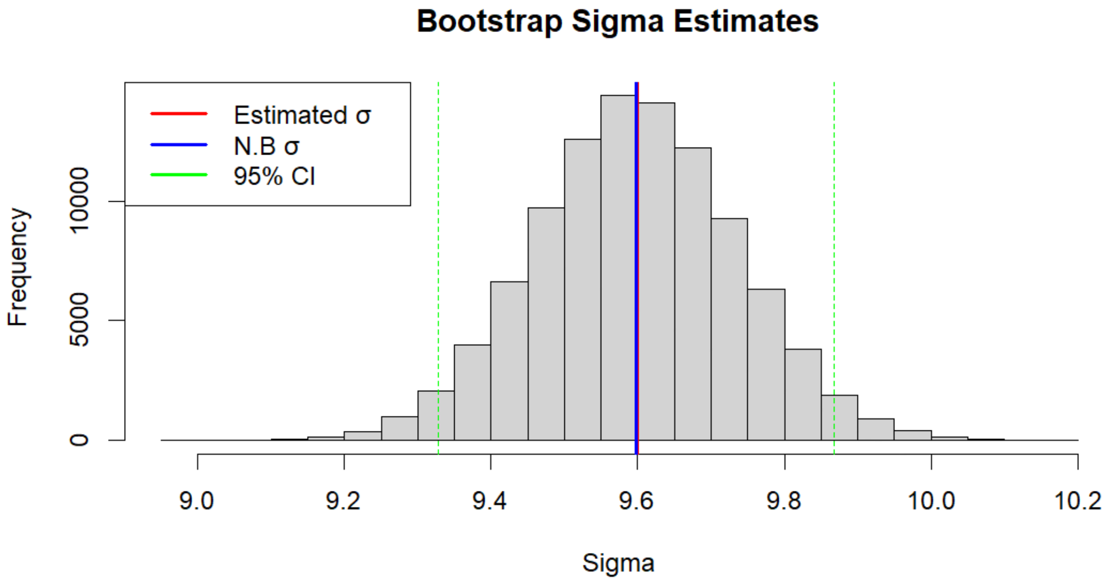
Observed Estimated RSE: 9.598666

To estimate sigma with bootstrap, parametric bootstrapping will be taken place. The distribution of the residuals in a linear regression is the Normal. Accordingly, the k samples created will be n Normal distribution with μ = the residuals mean and σ^2 = the observed Estimated RSE, where n is the length of the residuals. Then, for each sample, the standard deviation will be computed and saved in a vector.

Mean of Normal Bootstrapping for the Estimated RSE: 9.597615

95% CI for Normal bootstrapping for the Estimated RSE: [9.32846, 9.86712]

The observed estimated RSE is nearly the same as the mean of the bootstrapping one.



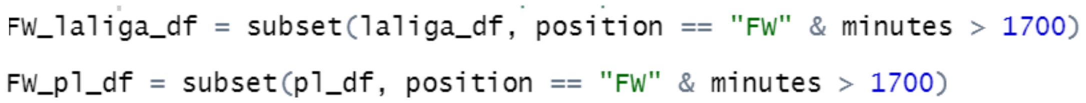
In the graphic is shown the similarity between the mean observed σ and the one calculated from the mean of Normal bootstrapping vector. The green lines are the limits of the 95% bootstrapping standard deviation Confidence Interval.

OTHER ESTIMATIONS AND ANALYZABLE FEATURES

Another concerning goal could be estimating means of goals with different restrictions. The current section has the purpose of showing multiple approaches of the dataset analysis.

Therefore, the following analysis consists in comparing the mean of goals per game of the *Forwards* who have played more than 1700 minutes between the two most important football leagues, La Liga and Premier League. To do so, parametric and nonparametric bootstrapping will be computed to estimate these means.

Firstly, two new subsets need to be created.

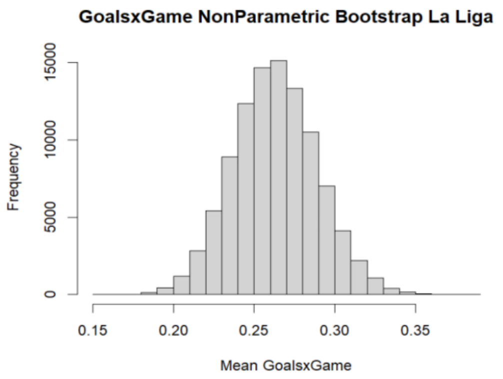


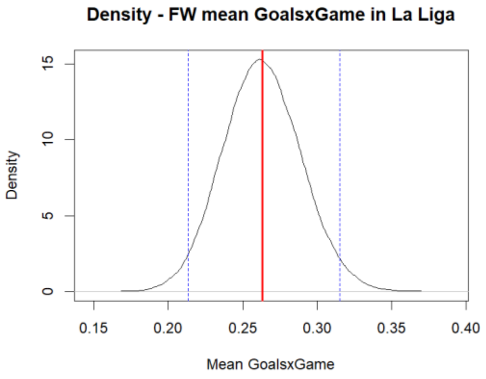
```
FW_laliga_df = subset(laliga_df, position == "FW" & minutes > 1700)
FW_pl_df = subset(pl_df, position == "FW" & minutes > 1700)
```

Non-Parametric Bootstrap

Mean value of observed FW goals per game in La Liga: 0.263166

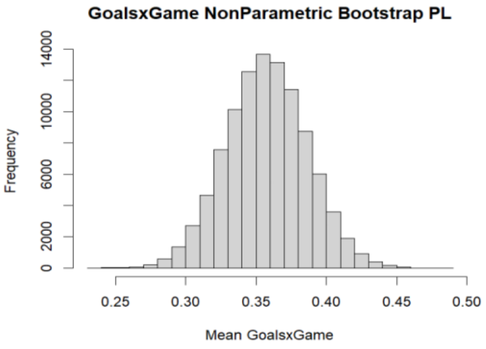
Mean value of non-Parametric bootstrapping FW goals per game in La Liga: 0.263041

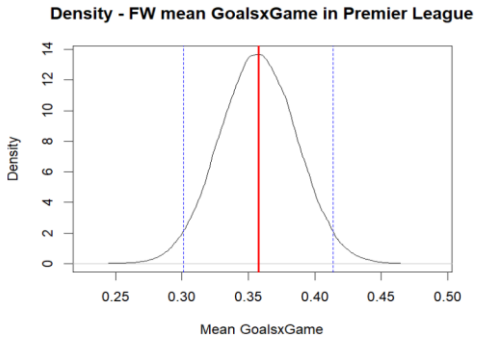




Mean value of observed FW goals per game in Premier League: 0.3575448

Mean value of non-Parametric bootstrapping FW goals per game in Premier League: 0.3574307





Comparison

It has been proved that Premier League Forwards score more than La Liga ones. Premier League is famous for having matches with lots of goals and a more explosive way of playing than in the Spanish league. The difference of goals per game between the two leagues is almost 0.1, which is 1 goal every 10 matches.

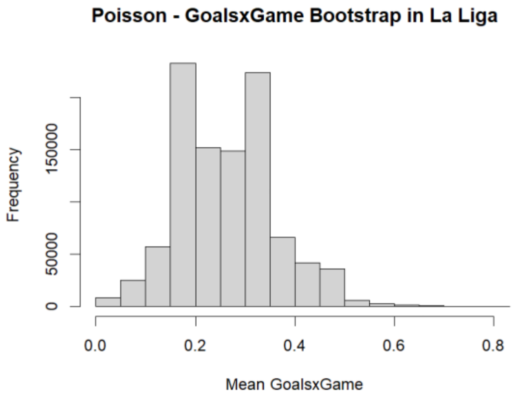
Nonetheless, both league graphics follow a Normal distribution according to the Central Limit Theorem[10]. As the *k* increases, the means tend to follow a Normal distribution, independently of the original distribution.

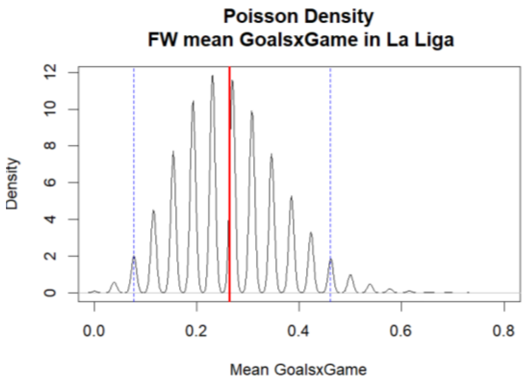
Parametric Bootstrap

The distribution chosen to compute the parametric bootstrap is the Poisson.

Mean value of observed FW goals per game in La Liga: 0.263166

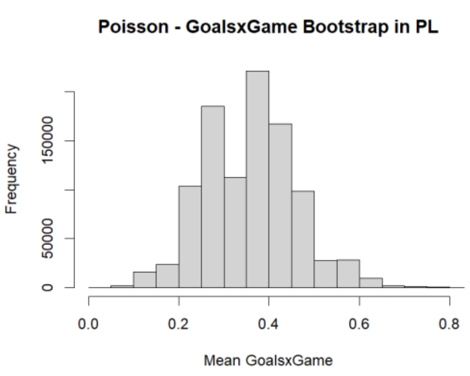
Mean value of Poisson bootstrapping FW goals per game in La Liga: 0.2628895

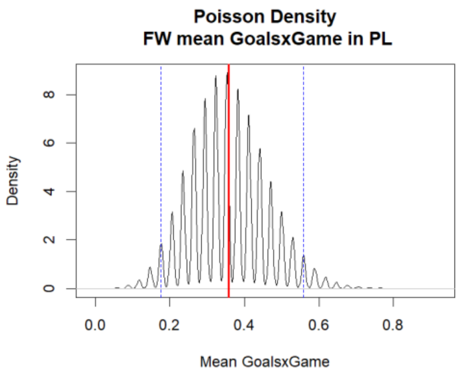




Mean value of observed FW goals per game in Premier League: 0.3575448

Mean value of Poisson bootstrapping FW goals per game in Premier League: 0.3575288





[10] *Central Limit Theorem: One of the most important Theorems in Mathematics. https://www.investopedia.com/terms/c/central_limit_theorem.asp*

Comparison

First, assuming a Poisson distribution will return a bootstrapping GoalsxGame mean really similar to the one observed. Nonetheless, there are significant observable differences. The bootstrapping vector histograms of both leagues are really distinct from a Normal distribution. Moreover, the Poisson density plots have lots of highs and lows. The reason why is Poisson is a discrete distribution, but in this case is used for fitting values in the [0,1] interval, so Poisson has been used for fitting continuous data. Despite of the highs and lows, the Normal distribution can be seen by only considering the density peaks. This gaussian distribution it is centered in the observed mean.

5.Conclusions

The dataset provided offers a plenty of possibilities to explore and study. A linear regression with the initial 400 rows could be computed, but the cleaning process would have been harder and even worthless, as there would be many columns for only 2644 rows. So, to compute a better model and study, other player leagues must be considered.

Additionally, in the original dataset lacked two key parameters such as the date of the player's contract end, or the value of the Euro the day when data was introduced.

However, the regression model provided is optimal for some predicting certain players' values. On the one hand, the fit is accurate for normal stats players who are constant and do not stand out from one year to another. Also, forward position is the best one to predict, as goals and assists normally come from them.

On the other hand, the model is not optimal for defenses, as it has no specifical columns such as 1vs1 performance or clean sheets. It is not optimal for football promises or athletes who have great changes of level performance too. Besides, players who have been injured the whole season and their player value has decreased won't have an accurate value prediction.

Beyond predicting and analyzing players' values, this dataset offers a great bunch of opportunities. Comparing the mean of the Forwards GoalsxGame in the two main leagues, could be extended to study the differences in the five leagues, or to study the assists in different leagues by different positions. Another way of studying the dataset would be predicting the number of goals in a season depending on the preseason player's value and the other dependent variables used in the current model.

6.Bibliography

1. Kriegsmaschine. Soccer Players' Values and Their Statistics. Kaggle. URL: https://www.kaggle.com/datasets/kriegsmaschine/soccer-players-values-and-their-statistics

2. Kang, H. (2013). The prevention and handling of the missing data. Korean Journal of Anesthesiology, 64(5), 402-406. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC374386/#:~:text=The%20most%20commonly%20used%20techniques,the%20form%20of%20an%20equation.

3. IBM. What is linear regression? URL: https://www.ibm.com/topics/linear-regression#:~:text=Resources-,What%20is%20linear%20regression%3F,is%20called%20the%20independent%20variable.

4. Scribbr. Akaike Information Criterion (AIC) explained. URL: https://www.scribbr.com/statistics/akaike-information-criterion/#:~:text=AIC%20model%20selection%20can%20help,used%20to%20reach%20that%20likelihood.

5. Penn State University. Parametric Bootstrap. URL: https://online.stat.psu.edu/stat555/node/119/#:~:text=Parametric%20bootstrap,or%20normal%20for%20continuous%20distribution.

6. Transfermarkt. Alexander Isak - Market value development. URL: https://www.transfermarkt.es/alexander-isak/marktwertverlauf/spieler/349066

7. Transfermarkt. German Pezzella - Market value development. URL: https://www.transfermarkt.com/german-pezzella/marktwertverlauf/spieler/113133

8. Transfermarkt. Isco - Performance data. URL: https://www.transfermarkt.com/isco/leistungsdaten/spieler/85288/saison/2020/plus/1

9. Transfermarkt. Wesley Fofana - Market value development. URL: https://www.transfermarkt.com/wesley-fofana/marktwertverlauf/spieler/475411

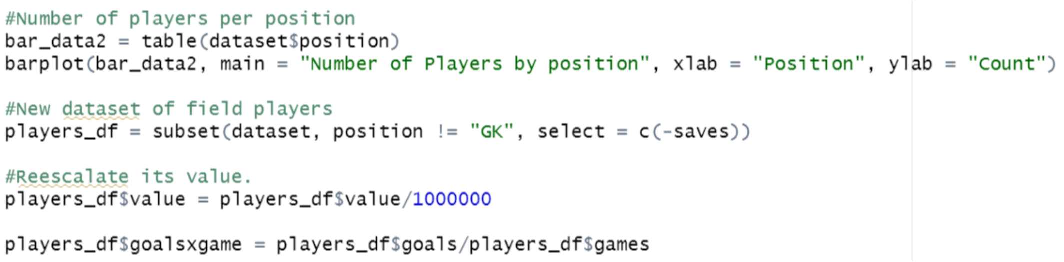
10. Puig, P. Campus Virtual pdfs: 1-Multiple Regression Models, 2-Residuals and Departures, 5-Parametric Bootstrap, 6-Non-Parametric Bootstrap. URL: https://e-aules.uab.cat/2022-23/course/view.php?idnumber=MO59684&profile=student

7.Appendix

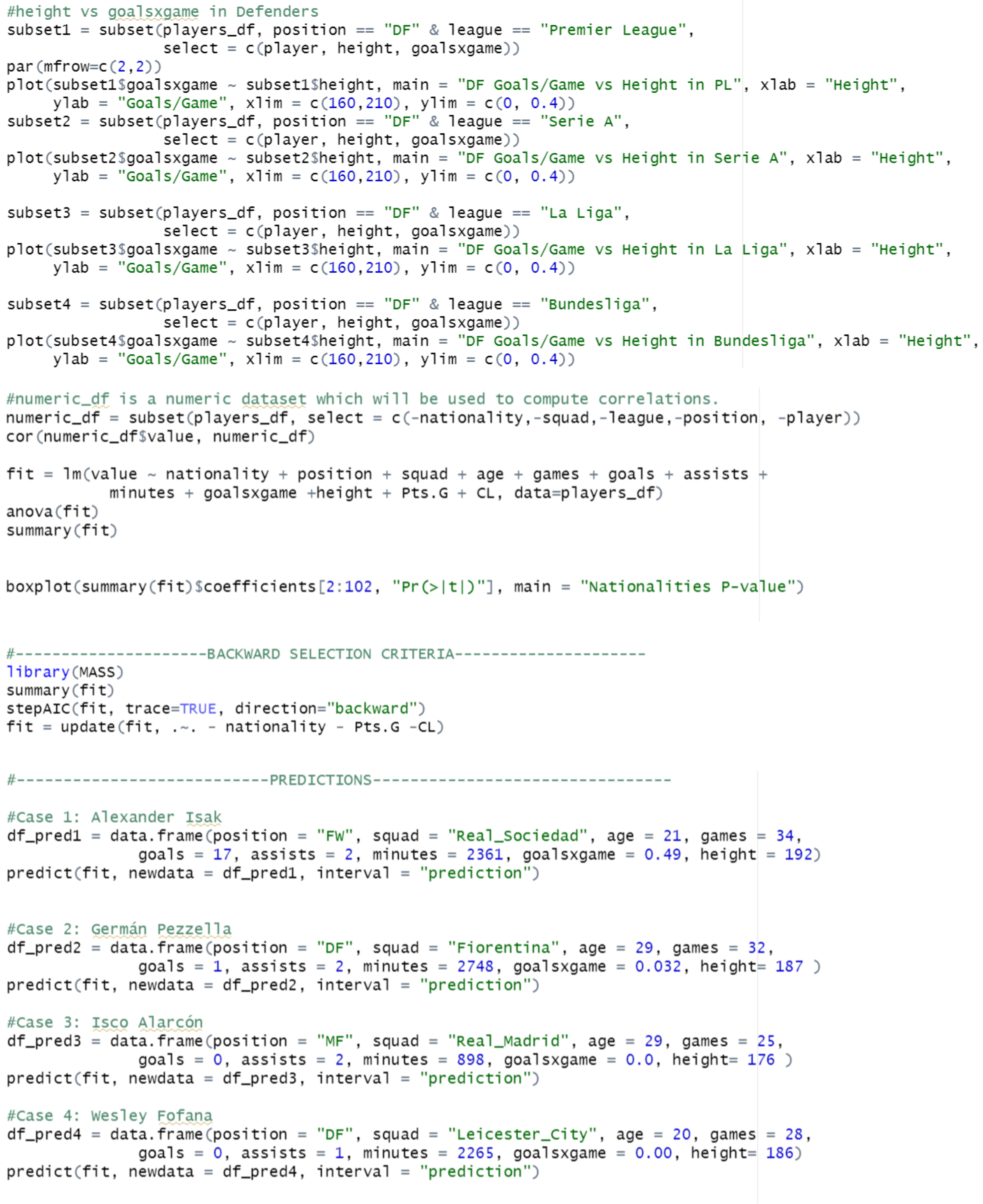
Firstly, the QR code to access to the dataset:

Click on it to download the .csv file.

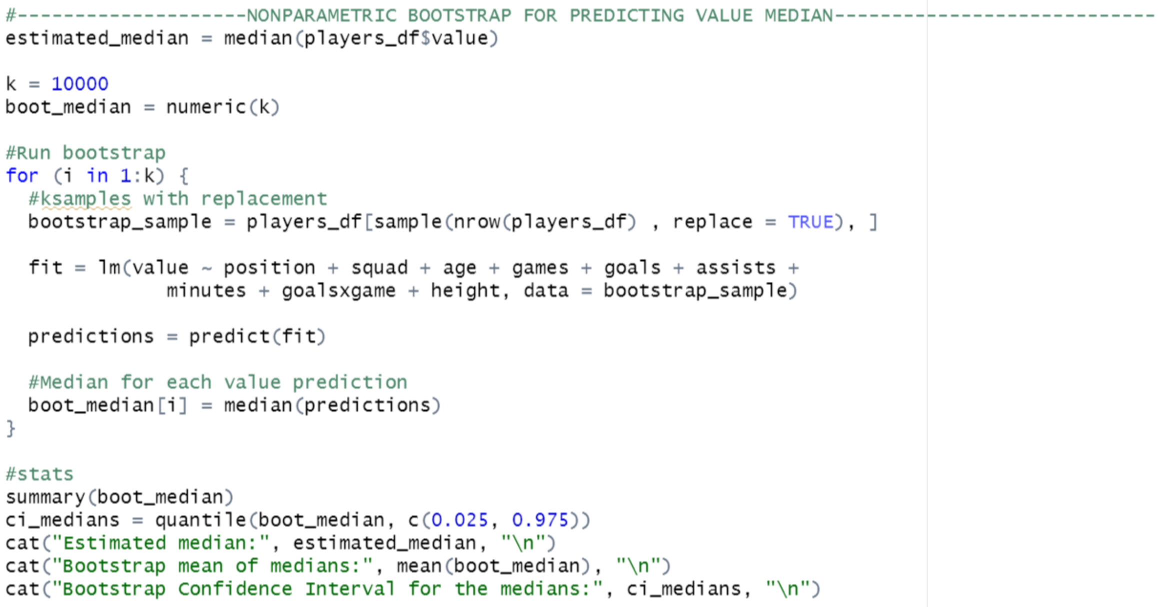
```
dataset = read.csv("transfermarkt_fbref_201920_copia.csv", sep=';', row.names = NULL, header=TRUE)

summary(dataset)
head(dataset)
str(dataset)

#Number of players per league
bar_data = table(dataset$league)
barplot(bar_data, main = "Number of Players by League", xlab = "League", ylab = "Count")

boxplot(dataset$value ~ dataset$league, main = "Player Value by League",
        xlab = "League", ylab = "Value")
#We can see that the outliers in Ligue1 might be Mbappe and Neymar, who are highly valued

hist(dataset$age, main = "Distribution of Player Ages", xlab = "Age", ylab = "Frequency")

plot(dataset$minutes, dataset$goals, main = "Player Goals v. Minutes Played",
     xlab = "Minutes Played", ylab = "Goals")
```



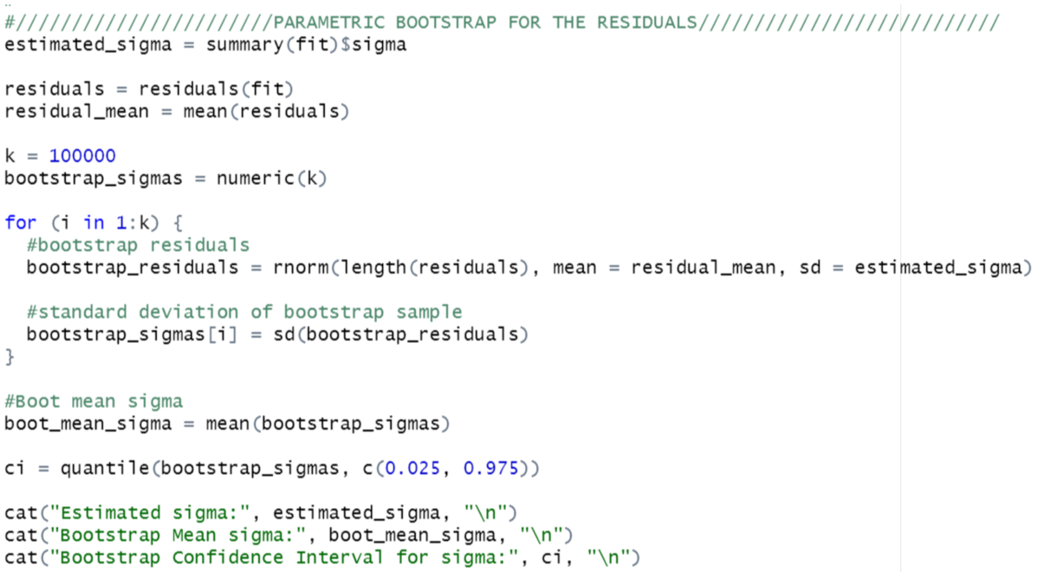
```
#Number of players per position
bar_data2 = table(dataset$position)
barplot(bar_data2, main = "Number of Players by position", xlab = "Position", ylab = "Count")

#New dataset of field players
players_df = subset(dataset, position != "GK", select = c(-saves))

#Reescalate its value.
players_df$value = players_df$value/1000000

players_df$goalsxgame = players_df$goals/players_df$games
```



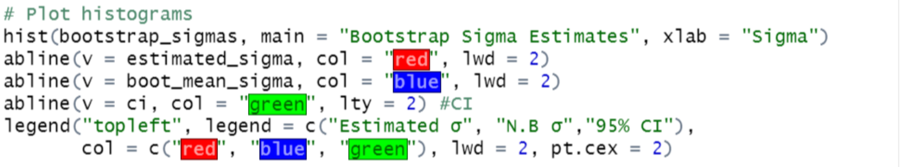
```
#height vs goalsxgame in Defenders
subset1 = subset(players_df, position == "DF" & league == "Premier League",
                 select = c(player, height, goalsxgame))
par(mfrow=c(2,2))
plot(subset1$goalsxgame ~ subset1$height, main = "DF Goals/Game vs Height in PL", xlab = "Height",
     ylab = "Goals/Game", xlim = c(160,210), ylim = c(0, 0.4))
subset2 = subset(players_df, position == "DF" & league == "Serie A",
                 select = c(player, height, goalsxgame))
plot(subset2$goalsxgame ~ subset2$height, main = "DF Goals/Game vs Height in Serie A", xlab = "Height",
     ylab = "Goals/Game", xlim = c(160,210), ylim = c(0, 0.4))

subset3 = subset(players_df, position == "DF" & league == "La Liga",
                 select = c(player, height, goalsxgame))
plot(subset3$goalsxgame ~ subset3$height, main = "DF Goals/Game vs Height in La Liga", xlab = "Height",
     ylab = "Goals/Game", xlim = c(160,210), ylim = c(0, 0.4))

subset4 = subset(players_df, position == "DF" & league == "Bundesliga",
                 select = c(player, height, goalsxgame))
plot(subset4$goalsxgame ~ subset4$height, main = "DF Goals/Game vs Height in Bundesliga", xlab = "Height",
     ylab = "Goals/Game", xlim = c(160,210), ylim = c(0, 0.4))

#numeric_df is a numeric dataset which will be used to compute correlations.
numeric_df = subset(players_df, select = c(-nationality,-squad,-league,-position, -player))
cor(numeric_df$value, numeric_df)

fit = lm(value ~ nationality + position + squad + age + games + goals + assists +
           minutes + goalsxgame +height + Pts.G + CL, data=players_df)
anova(fit)
summary(fit)


boxplot(summary(fit)$coefficients[2:102, "Pr(>|t|)"], main = "Nationalities P-value")


#---------------------BACKWARD SELECTION CRITERIA---------------------
library(MASS)
summary(fit)
stepAIC(fit, trace=TRUE, direction="backward")
fit = update(fit, .~. - nationality - Pts.G -CL)


#---------------------------PREDICTIONS-------------------------------

#Case 1: Alexander Isak
df_pred1 = data.frame(position = "FW", squad = "Real_Sociedad", age = 21, games = 34,
              goals = 17, assists = 2, minutes = 2361, goalsxgame = 0.49, height = 192)
predict(fit, newdata = df_pred1, interval = "prediction")


#Case 2: Germán Pezzella
df_pred2 = data.frame(position = "DF", squad = "Fiorentina", age = 29, games = 32,
              goals = 1, assists = 2, minutes = 2748, goalsxgame = 0.032, height= 187 )
predict(fit, newdata = df_pred2, interval = "prediction")

#Case 3: Isco Alarcón
df_pred3 = data.frame(position = "MF", squad = "Real_Madrid", age = 29, games = 25,
              goals = 0, assists = 2, minutes = 898, goalsxgame = 0.0, height= 176 )
predict(fit, newdata = df_pred3, interval = "prediction")

#Case 4: Wesley Fofana
df_pred4 = data.frame(position = "DF", squad = "Leicester_City", age = 20, games = 28,
              goals = 0, assists = 1, minutes = 2265, goalsxgame = 0.00, height= 186)
predict(fit, newdata = df_pred4, interval = "prediction")
```



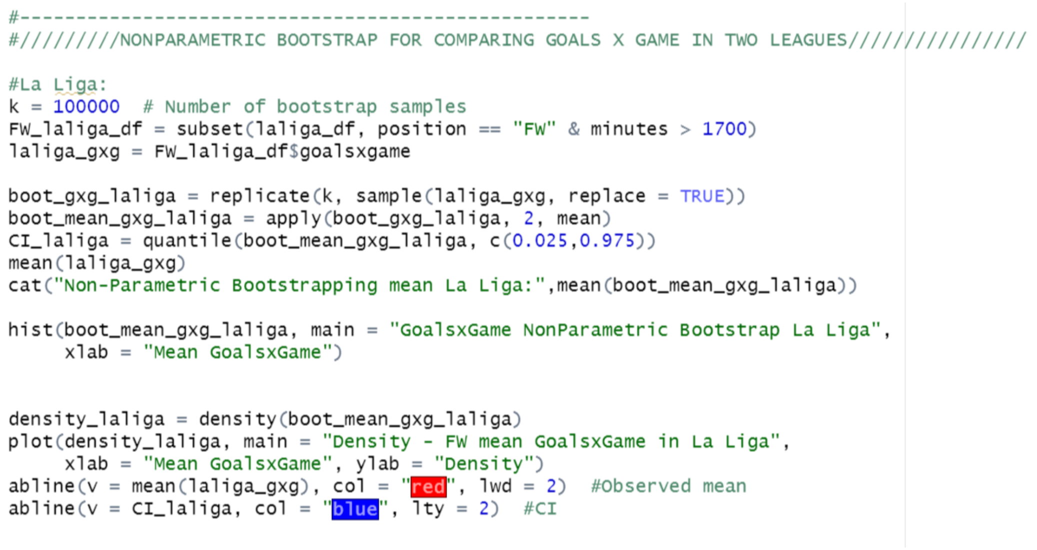
```
#--------------------NONPARAMETRIC BOOTSTRAP FOR PREDICTING VALUE MEDIAN----------------------------
estimated_median = median(players_df$value)

k = 10000
boot_median = numeric(k)

#Run bootstrap
for (i in 1:k) {
  #ksamples with replacement
  bootstrap_sample = players_df[sample(nrow(players_df) , replace = TRUE), ]

  fit = lm(value ~ position + squad + age + games + goals + assists +
             minutes + goalsxgame + height, data = bootstrap_sample)

  predictions = predict(fit)

  #Median for each value prediction
  boot_median[i] = median(predictions)
}

#stats
summary(boot_median)
ci_medians = quantile(boot_median, c(0.025, 0.975))
cat("Estimated median:", estimated_median, "\n")
cat("Bootstrap mean of medians:", mean(boot_median), "\n")
cat("Bootstrap Confidence Interval for the medians:", ci_medians, "\n")
```



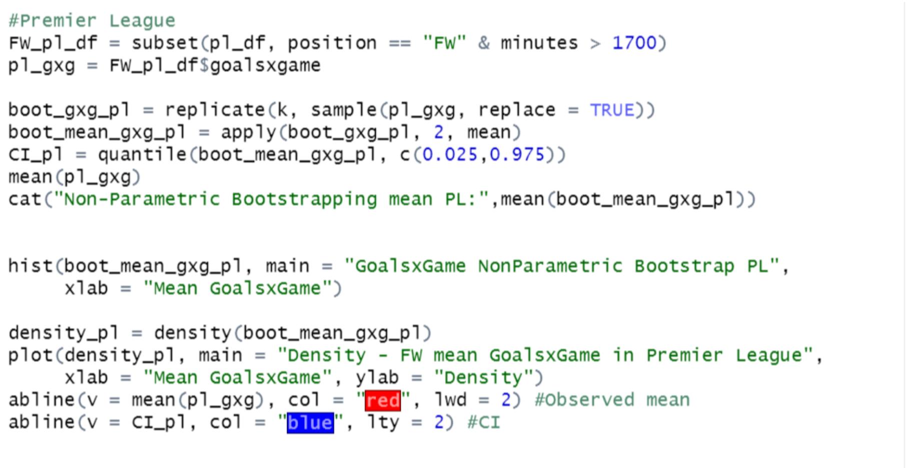
```
#/////////////////////PARAMETRIC BOOTSTRAP FOR THE RESIDUALS/////////////////////////
estimated_sigma = summary(fit)$sigma

residuals = residuals(fit)
residual_mean = mean(residuals)

k = 100000
bootstrap_sigmas = numeric(k)

for (i in 1:k) {
  #bootstrap residuals
  bootstrap_residuals = rnorm(length(residuals), mean = residual_mean, sd = estimated_sigma)

  #standard deviation of bootstrap sample
  bootstrap_sigmas[i] = sd(bootstrap_residuals)
}

#Boot mean sigma
boot_mean_sigma = mean(bootstrap_sigmas)

ci = quantile(bootstrap_sigmas, c(0.025, 0.975))

cat("Estimated sigma:", estimated_sigma, "\n")
cat("Bootstrap Mean sigma:", boot_mean_sigma, "\n")
cat("Bootstrap Confidence Interval for sigma:", ci, "\n")
```



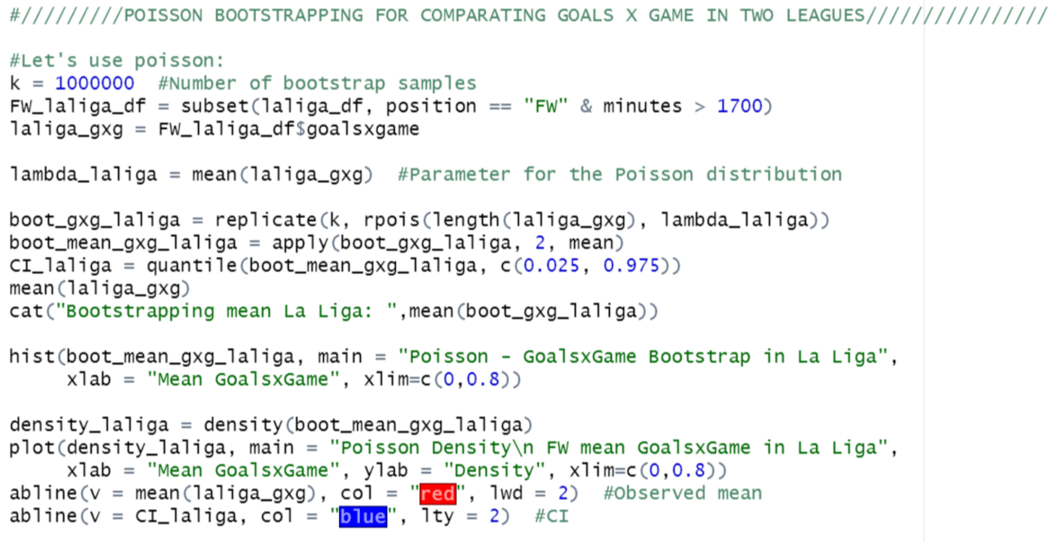
```
# Plot histograms
hist(bootstrap_sigmas, main = "Bootstrap Sigma Estimates", xlab = "Sigma")
abline(v = estimated_sigma, col = "red", lwd = 2)
abline(v = boot_mean_sigma, col = "blue", lwd = 2)
abline(v = ci, col = "green", lty = 2) #CI
legend("topleft", legend = c("Estimated σ", "N.B σ","95% CI"),
       col = c("red", "blue", "green"), lwd = 2, pt.cex = 2)
```



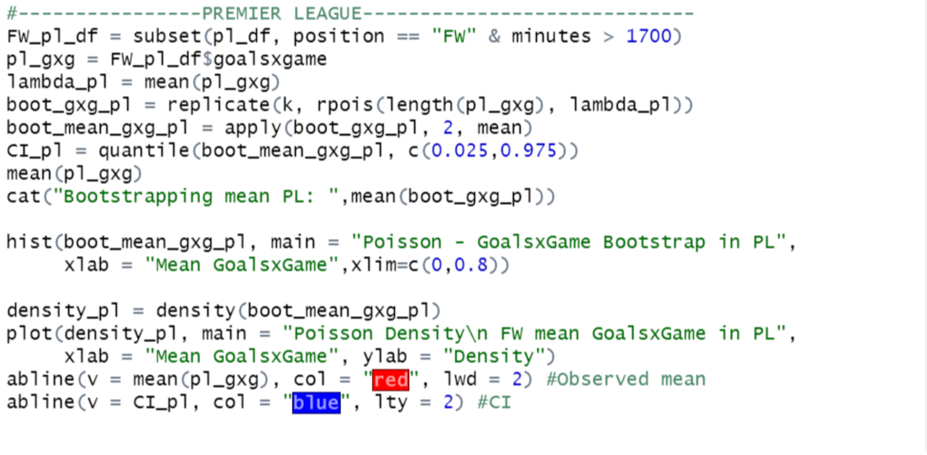
```
#---------------------------------------------------
#/////////NONPARAMETRIC BOOTSTRAP FOR COMPARING GOALS X GAME IN TWO LEAGUES///////////////

#La Liga:
k = 100000  # Number of bootstrap samples
FW_laliga_df = subset(laliga_df, position == "FW" & minutes > 1700)
laliga_gxg = FW_laliga_df$goalsxgame

boot_gxg_laliga = replicate(k, sample(laliga_gxg, replace = TRUE))
boot_mean_gxg_laliga = apply(boot_gxg_laliga, 2, mean)
CI_laliga = quantile(boot_mean_gxg_laliga, c(0.025,0.975))
mean(laliga_gxg)
cat("Non-Parametric Bootstrapping mean La Liga:",mean(boot_mean_gxg_laliga))

hist(boot_mean_gxg_laliga, main = "GoalsxGame NonParametric Bootstrap La Liga",
     xlab = "Mean GoalsxGame")


density_laliga = density(boot_mean_gxg_laliga)
plot(density_laliga, main = "Density - FW mean GoalsxGame in La Liga",
     xlab = "Mean GoalsxGame", ylab = "Density")
abline(v = mean(laliga_gxg), col = "red", lwd = 2)  #Observed mean
abline(v = CI_laliga, col = "blue", lty = 2)  #CI
```



```
#Premier League
FW_pl_df = subset(pl_df, position == "FW" & minutes > 1700)
pl_gxg = FW_pl_df$goalsxgame

boot_gxg_pl = replicate(k, sample(pl_gxg, replace = TRUE))
boot_mean_gxg_pl = apply(boot_gxg_pl, 2, mean)
CI_pl = quantile(boot_mean_gxg_pl, c(0.025,0.975))
mean(pl_gxg)
cat("Non-Parametric Bootstrapping mean PL:",mean(boot_mean_gxg_pl))


hist(boot_mean_gxg_pl, main = "GoalsxGame NonParametric Bootstrap PL",
     xlab = "Mean GoalsxGame")

density_pl = density(boot_mean_gxg_pl)
plot(density_pl, main = "Density - FW mean GoalsxGame in Premier League",
     xlab = "Mean GoalsxGame", ylab = "Density")
abline(v = mean(pl_gxg), col = "red", lwd = 2) #Observed mean
abline(v = CI_pl, col = "blue", lty = 2) #CI
```



```
#/////////POISSON BOOTSTRAPPING FOR COMPARATING GOALS X GAME IN TWO LEAGUES///////////////

#Let's use poisson:
k = 1000000  #Number of bootstrap samples
FW_laliga_df = subset(laliga_df, position == "FW" & minutes > 1700)
laliga_gxg = FW_laliga_df$goalsxgame

lambda_laliga = mean(laliga_gxg)  #Parameter for the Poisson distribution

boot_gxg_laliga = replicate(k, rpois(length(laliga_gxg), lambda_laliga))
boot_mean_gxg_laliga = apply(boot_gxg_laliga, 2, mean)
CI_laliga = quantile(boot_mean_gxg_laliga, c(0.025, 0.975))
mean(laliga_gxg)
cat("Bootstrapping mean La Liga: ",mean(boot_gxg_laliga))

hist(boot_mean_gxg_laliga, main = "Poisson - GoalsxGame Bootstrap in La Liga",
     xlab = "Mean GoalsxGame", xlim=c(0,0.8))

density_laliga = density(boot_mean_gxg_laliga)
plot(density_laliga, main = "Poisson Density\n FW mean GoalsxGame in La Liga",
     xlab = "Mean GoalsxGame", ylab = "Density", xlim=c(0,0.8))
abline(v = mean(laliga_gxg), col = "red", lwd = 2)  #Observed mean
abline(v = CI_laliga, col = "blue", lty = 2)  #CI
```



```
#----------------PREMIER LEAGUE----------------------------
FW_pl_df = subset(pl_df, position == "FW" & minutes > 1700)
pl_gxg = FW_pl_df$goalsxgame
lambda_pl = mean(pl_gxg)
boot_gxg_pl = replicate(k, rpois(length(pl_gxg), lambda_pl))
boot_mean_gxg_pl = apply(boot_gxg_pl, 2, mean)
CI_pl = quantile(boot_mean_gxg_pl, c(0.025,0.975))
mean(pl_gxg)
cat("Bootstrapping mean PL: ",mean(boot_gxg_pl))

hist(boot_mean_gxg_pl, main = "Poisson - GoalsxGame Bootstrap in PL",
     xlab = "Mean GoalsxGame",xlim=c(0,0.8))

density_pl = density(boot_mean_gxg_pl)
plot(density_pl, main = "Poisson Density\n FW mean GoalsxGame in PL",
     xlab = "Mean GoalsxGame", ylab = "Density")
abline(v = mean(pl_gxg), col = "red", lwd = 2) #Observed mean
abline(v = CI_pl, col = "blue", lty = 2) #CI
```