Department of Computer Science
American International University-Bangladesh
Mid Term Report

Course Name: INTRODUCTION TO DATA SCIENCE
Final Term Report.
Section: B

Supervised By:

Dr. Akinul Islam Jony

Associate Professor, Computer Science-AIUB

Submission Date: December 12, 2022.

Name	ID
SAJID, IBNA MAHBUB	20-42109-1
RATRI, SAIMA SADIA	20-43793-2
HAMIM, SULTANUL ARIFEEN	20-42017-1
ISLAM, ASHRAFUL	20-42010-1

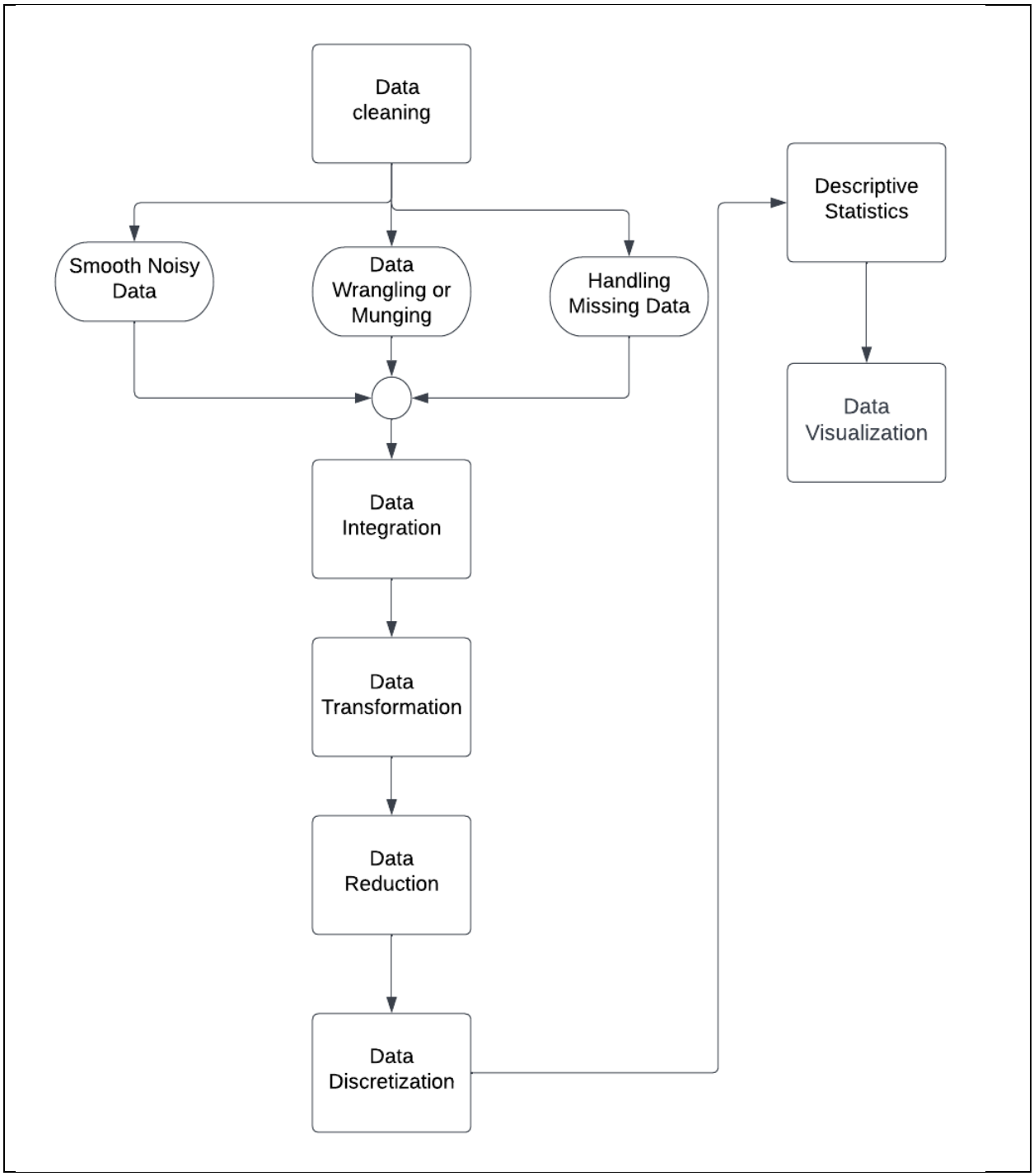
Project Overview:

For this project, we have been assigned to scrap data from webpages, perform preprocessing techniques on them, describe them in the light of descriptive statistics and visualize them using R language.

In our project firstly, we chose footballers' data of the two most successful clubs of this generation, namely Barcelona and Real Madrid. We collected Real Madrid & Barca data for session 2015-2016 from ESPN website. Then we merged two datasets. After that, we did many comparisons on data like why Barcelona was the champion, and the performance of players and analyzed the dataset. Real-world data is frequently incomplete, noisy, and inconsistent, meaning it needs to be cleaned up before it can be put to the intended use. Data pre-processing is a common term for this. Data preprocessing is a data mining technique used to turn raw data into a practical and effective format. The most important tasks involved in data pre-processing are Data Cleaning, Data Integration, Data Transformation, Data Reduction, and Data Discretization. We did data pre-processing where it was needed. In Descriptive analysis, we described our data with the help of descriptive methods. In the descriptive analysis, we describe our data in some manner and present it in a meaningful way so that it can be easily understood. To describe a comparison between different things we did the Mean, Median, Mode, Range, Variance, Quartile & Percentile. Lastly, we did data visualization to see and understand as visualizations can more effectively allow the reader to digest information. Graphics can allow users to deliver insights in a much easier fashion than describing through text and can also have a greater impact. Here we tried to visualize almost every aspect of comparison & relation.

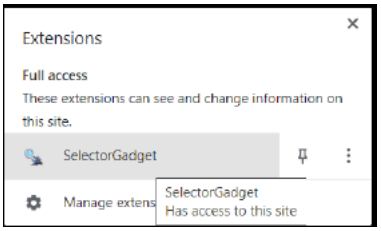
Project Solution Design:

We initially gathered our player lists and performance information for Barcelona and Real Madrid from several websites in order to prepare the dataset for data analysis. We then recorded the information in a CSV file. The data pre-processing is then done. Data cleaning is the process of inspecting a raw dataset to find and eliminate errors, duplication, and superfluous data. The table had some missing data, which we replaced with N/A and then filled up with the median. Then we tried to manage every item of noisy data that was in the dataset. After performing data cleaning, measures for data integration, data transformation, data reduction, and data discretization were taken to further clean the data set. We concentrated on using descriptive statistics to rationally simplify our enormous volumes of data after completing the data preprocessing. Moreover, to sum up, the dataset's approximate data. In our data collection, we used the following metrics: Mean, Median, Mode, Range, Variance, Standard Deviation, Quartiles, Percentiles, and Interquartile Ranges. We used data visualization to present facts and data graphically after finishing the descriptive statistics.

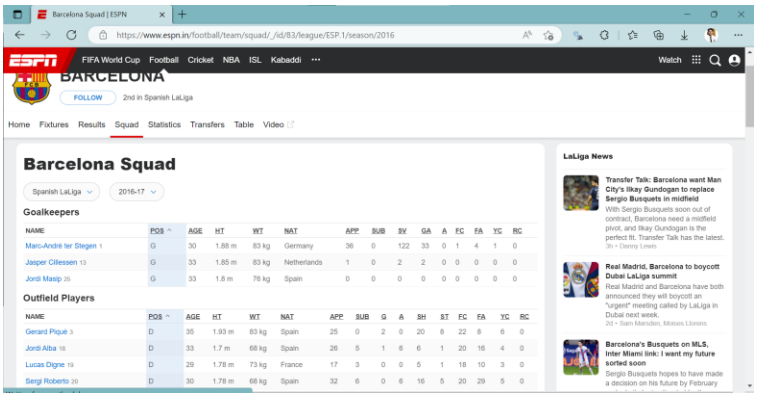

Data cleaning
Smooth Noisy Data
Data Wrangling or Munging
Handling Missing Data
Descriptive Statistics
Data Visualization
Data Integration
Data Transformation
Data Reduction
Data Discretization

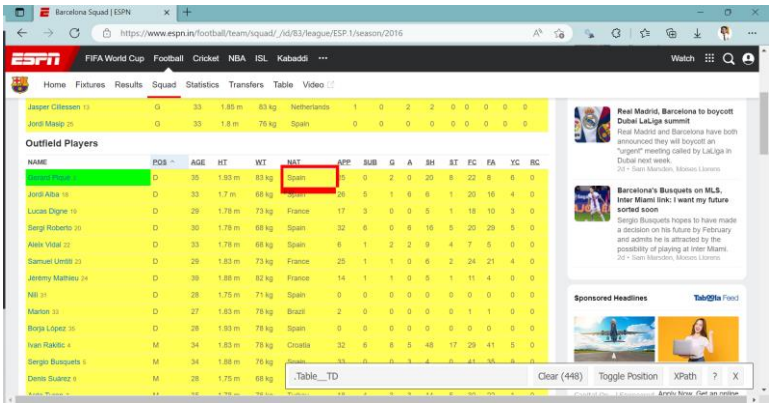
Data Collection:

For this project, we start to scrap the data from the website. First, we start to scrap the data from team Barcelona. In this process, we use a selector gadget to simply select data on a website and it will determine its HTML/CSS tags, ids and classes.



Getting Barca Data:

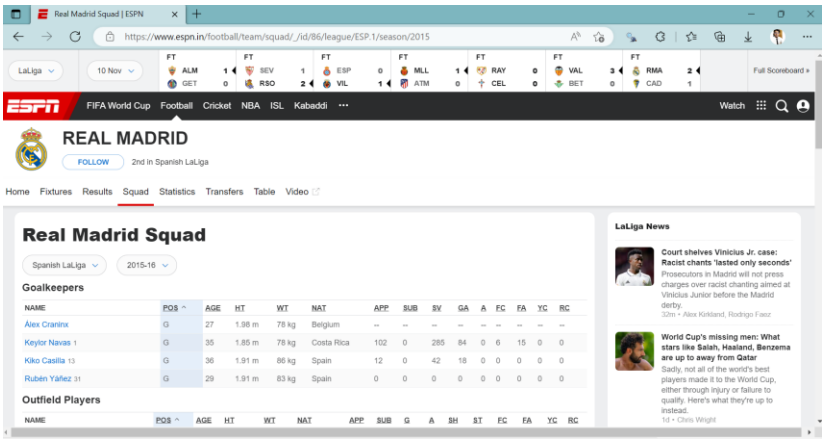


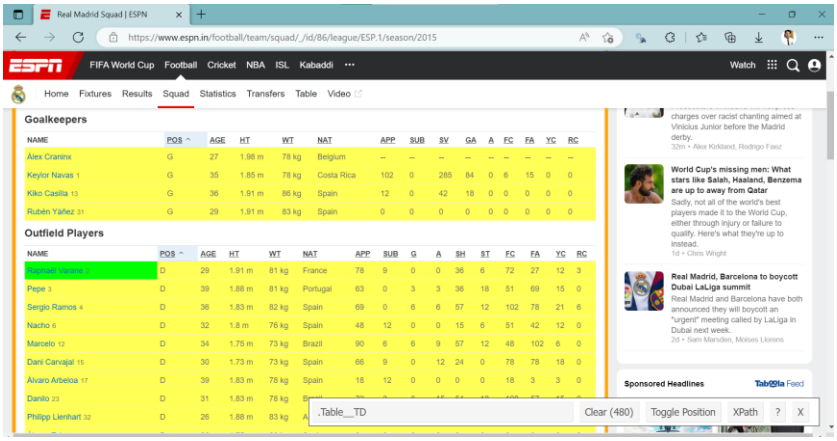


Code:

```
library(rvest)



players =
read_html("https://www.espn.in/football/team/squad/_/id/83/league/ESP.1/season/2016")


pl = html_nodes(players, css=".Table__TD")

pl

barca <-data.frame(html_table(players, header = TRUE)[[2]])


View(barca)

write.csv(barca,"F:\\barca.csv")
```

Getting Real Madrid Data:





real = read_html("https://www.espn.in/football/team/squad/_/id/86/league/ESP.1/season/2015")

pl = html_nodes(real, css=".Table__TD")

pl

real <-data.frame(html_table(real, header = TRUE)[[2]])

View(real)

write.csv(barca,"F:\\real.csv")

Output:

Filter

	Name	POS	Age	HT	WT	NAT	APP	SUB	G	A
1	Gerard Piqué3	D	35	1.93 m	83 kg	Spain	25	0	2	
2	Jordi Alba18	D	33	1.7 m	68 kg	Spain	26	5	1	
3	Lucas Digne19	D	29	1.78 m	73 kg	France	17	3	0	
4	Sergi Roberto20	D	30	1.78 m	68 kg	Spain	32	6	0	
5	Aleix Vidal22	D	33	1.78 m	68 kg	Spain	6	1	2	
6	Samuel Umtiti23	D	29	1.83 m	73 kg	France	25	1	1	
7	Jérémy Mathieu24	D	39	1.88 m	82 kg	France	14	1	1	
8	Nili31	D	28	1.75 m	71 kg	Spain	0	NA	0	
9	Marlon33	D	27	1.83 m	78 kg	Brazil	2	0	0	
10	Borja López35	D	28	1.93 m	78 kg	Spain	0	0	0	
11	Ivan Rakitic4	M	34	1.83 m	78 kg	Croatia	32	6	8	
12	Sergio Busquets5	M	34	1.88 m	76 kg	Spain	33	0	0	

	Name	POS	Age	HT	WT	NAT	APP	SUB	G	A
13	Casemiro14	M	30	1.85 m	85 kg	Brazil	69	18	5	
14	Mateo Kovacic16	M	28	1.78 m	78 kg	Croatia	75	51	0	
15	Lucas Vázquez18	M	31	1.73 m	68 kg	Spain	75	45	12	18
16	Luka Modric19	M	37	1.73 m	66 kg	Croatia	96	3	6	12
17	Isco22	M	30	1.75 m	78 kg	Spain	93	30	9	21
18	Martin Ødegaard27	M	23	1.78 m	999	Norway	0	NA	0	NA
19	Marcos Llorente28	M	27	1.83 m	73 kg	Spain	6	6	0	
20	Lazo33	M	26	1.8 m	68 kg	Spain	NA	0	NA	NA
21	Marco Asensio	F	26	1.83 m	76 kg	Spain	102	3	12	30
22	Cristiano Ronaldo7	F	37	1.88 m	83 kg	Portugal	108	0	105	33
23	Karim Benzema9	F	34	1.85 m	81 kg	France	81	3	72	21
24	Gareth Bale11	F	33	1.85 m	81 kg	Wales	69	6	57	30

Data Pre-processing:

Now the most important phase of the data analysis starts which is data pre-processing. We are going to use pre-processing techniques on these two datasets to prepare a complete dataset for analysis and visualization.

1. **Data Cleaning**
    - **Handling Missing Data:** To handle missing data we first need to search the data set for any value that is not assigned. To do so we write a code that will show us the row which contains the missing value,

      Code:

```
missing <- real[!complete.cases(real),]
print(missing)

missing2 <- barca[!complete.cases(barca),]
print(missing2)
```

      Output:

```
> missing <- real[!complete.cases(real),]
> print(missing)
                Name POS Age     HT    WT    NAT APP SUB  G  A SH
18 Martin Ødegaard27   M  23 1.78 m   999 Norway   0  NA  0 NA NA
20            Lazo33   M  26  1.8 m 68 kg  Spain  NA   0 NA NA NA
   ST FC FA YC RC
18 NA NA NA NA NA
20  0  0  0 NA  0
>
> missing2 <- barca[!complete.cases(barca),]
> print(missing2)
                Name POS Age     HT    WT   NAT APP SUB  G A SH ST
8            Nili31   D  28 1.75 m 71 kg Spain   0  NA  0 0  0 NA
20 Àlex Carbonell30   M  25 1.83 m 76 kg Spain  NA   0 NA 0  0  0
25   Marc Cardona39   F  27 1.83 m 68 kg Spain   0  NA  0 0 NA  0
   FC FA YC RC
8   0  0  0 NA
20  0  0  0  0
25 NA  0  0 NA
>
```

      Now that we have ratted out the missing data, The next step is to perform the handling procedure. As we can see these are crucial player data about the performance and overall season. So, any player with missing data must be removed from the data set as they can't be filled with any method or assumption.

      Code:

```
real <- na.omit(real)
barca <- na.omit(barca)
real
barca
```

Output:

```
1        Gerard Piqué3   D  35 1.93 m 83 kg     Spain  25   0  2
2         Jordi Alba18   D  33  1.7 m 68 kg     Spain  26   5  1
3        Lucas Digne19   D  29 1.78 m 73 kg    France  17   3  0
4      Sergi Roberto20   D  30 1.78 m 68 kg     Spain  32   6  0
5        Aleix Vidal22   D  33 1.78 m 68 kg     Spain   6   1  2
6      Samuel Umtiti23   D  29 1.83 m 73 kg    France  25   1  1
7     Jérémy Mathieu24   D  39 1.88 m 82 kg    France  14   1  1
9            Marlon33   D  27 1.83 m 78 kg    Brazil   2   0  0
10       Borja López35   D  28 1.93 m 78 kg     Spain   0   0  0
11       Ivan Rakitic4   M  34 1.83 m 78 kg   Croatia  32   6  8
12    Sergio Busquets5   M  34 1.88 m 76 kg     Spain  33   0  0
13       Denis Suárez6   M  28 1.75 m 68 kg     Spain  26  14  1
14         Arda Turan7   M  35 1.78 m 76 kg    Turkey  18   4  3
15     Andrés Iniesta8   M  38  1.7 m 68 kg     Spain  23  10  0
16           Rafinha12   M  30 1.75 m 71 kg    Brazil  18   4  6
17 Javier Mascherano14   M  38 1.75 m 73 kg Argentina  25   5  1
18       André Gomes21   M  29 1.88 m 83 kg  Portugal  30  13  3
19      Carles Aleñá28   M  24  1.8 m 73 kg     Spain   3   3  0
21        Luis Suárez9   F  35 1.83 m 86 kg   Uruguay  35   1 29
22      Lionel Messi10   F  35  1.7 m 72 kg Argentina  34   2 37
23            Neymar11   F  30 1.75 m 68 kg    Brazil  30   0 13
24       Paco Alcácer17   F  29 1.75 m 72 kg     Spain  21  14  6
4             Nacho6   D  32  1.8 m 76 kg     Spain  48  12  0  0
5           Marcelo12   D  34 1.75 m 73 kg    Brazil  90   6  6  9
6     Dani Carvajal15   D  30 1.73 m 73 kg     Spain  66   9  0 12
7    Álvaro Arbeloa17   D  39 1.83 m 78 kg     Spain  18  12  0  0
8           Danilo23   D  31 1.83 m 78 kg    Brazil  72   3  6 15
9  Philipp Lienhart32   D  26 1.88 m 83 kg   Austria   0   0  0  0
10    Álvaro Tejero34   D  26 1.75 m 68 kg     Spain   0   0  0  0
11       Toni Kroos8   M  32 1.83 m 76 kg   Germany  96   0  3 30
12  James Rodríguez10   M  31  1.8 m 73 kg  Colombia  78  27 21 24
13        Casemiro14   M  30 1.85 m 83 kg    Brazil  69  18  3  9
14    Mateo Kovacic16   M  28 1.78 m 78 kg   Croatia  75  51  0  6
15    Lucas Vázquez18   M  31 1.73 m 68 kg     Spain  75  45 12 18
16      Luka Modric19   M  37 1.73 m 66 kg   Croatia  96   3  6 12
17            Isco22   M  30 1.75 m 78 kg     Spain  93  30  9 21
19  Marcos Llorente28   M  27 1.83 m 73 kg     Spain   6   6  0  0
21      Marco Asensio   F  26 1.83 m 76 kg     Spain 102   3 12 30
22 Cristiano Ronaldo7   F  37 1.88 m 83 kg  Portugal 108   0 105 33
23     Karim Benzema9   F  34 1.85 m 81 kg    France  81   3  72 21
24      Gareth Bale11   F  33 1.85 m 81 kg     Wales  69   6  57 30
25            Jesé20   F  29 1.78 m 72 kg     Spain  84  63  15 18
26    Borja Mayoral29   F  25 1.83 m 73 kg     Spain  18   9   0  0
   SH  ST  FC  FA YC RC
```

- **Smooth Noisy Data:** In the dataset, we can see that some columns contain a mixture of both numerical and character data. Like Weight contains extra kg and height contains m as a meter. For the betterment of the calculation, we have to remove those noises from the dataset.
- **Smooth Noisy Data:**

  Code:

  To remove kg and m from the Height and Weight column,

  ```
  barca$HT <- sub("[[:space:]].*", "", barca$HT)

  barca$WT <- sub("[[:space:]].*", "", barca$WT)
  ```

real$HT <- sub("[[:space:]].*", "", real$HT)

real$WT <- sub("[[:space:]].*", "", real$WT)

	Name	POS	Age	HT	WT	NAT	APP	SUB	G	A	SH	ST	FC	FA	YC	RC
1	Raphaël Varane2	D	29	1.91	81	France	78	9	0	0	36	6	72	27	12	3
2	Pepe3	D	39	1.88	81	Portugal	63	0	3	3	36	18	51	69	15	0
3	Sergio Ramos4	D	36	1.83	82	Spain	69	0	6	6	57	12	102	78	21	6
4	Nacho6	D	32	1.8	76	Spain	48	12	0	0	15	6	51	42	12	0
5	Marcelo12	D	34	1.75	73	Brazil	90	6	6	9	57	12	48	102	6	0
6	Dani Carvajal15	D	30	1.73	73	Spain	66	9	0	12	24	0	78	78	18	0
7	Álvaro Arbeloa17	D	39	1.83	78	Spain	18	12	0	0	0	0	18	3	3	0
8	Danilo23	D	31	1.83	78	Brazil	72	3	6	15	54	18	108	57	15	0
9	Philipp Lienhart32	D	26	1.88	83	Austria	0	0	0	0	0	0	0	0	0	0
10	Álvaro Tejero34	D	26	1.75	68	Spain	0	0	0	0	0	0	0	0	0	0
11	Toni Kroos8	M	32	1.83	76	Germany	96	0	3	30	60	12	102	141	9	0
12	James Rodríguez10	M	31	1.8	73	Colombia	78	27	21	24	117	54	36	51	3	0

To remove the number from player name,
Code:
barca$Name <-gsub("[1-50]","",as.character(barca$Name))

real$Name <- gsub("[1-50]","",as.character(real$Name))

Output:

	Name	POS	Age	HT	WT	NAT	APP	SUB	G	A	SH	ST	FC	FA	YC	RC
1	Gerard Piqué	D	35	1.93	83	Spain	25	0	2	0	20	8	22	8	6	0
2	Jordi Alba	D	33	1.7	68	Spain	26	5	1	6	6	1	20	16	4	0
3	Lucas Digne	D	29	1.78	73	France	17	3	0	0	5	1	18	10	3	0
4	Sergi Roberto	D	30	1.78	68	Spain	32	6	0	6	16	5	20	29	5	0
5	Aleix Vidal	D	33	1.78	68	Spain	6	1	2	2	9	4	7	5	0	0
6	Samuel Umtiti	D	29	1.83	73	France	25	1	1	0	6	2	24	21	4	0
7	Jérémy Mathieu	D	39	1.88	82	France	14	1	1	0	5	1	11	4	0	0
8	Nili	D	28	1.75	71	Spain	0	0	0	0	0	0	0	0	0	0
9	Marlon	D	27	1.83	78	Brazil	2	0	0	0	0	0	1	1	0	0
10	Borja López	D	28	1.93	78	Spain	0	0	0	0	0	0	0	0	0	0
11	Ivan Rakitic	M	34	1.83	78	Croatia	32	6	8	5	48	17	29	41	5	0
12	Sergio Busquets	M	34	1.88	76	Spain	33	0	0	3	4	0	41	35	9	0

- **Data Munging:** The dataset does not require munging because all the data are within the same range.

2. Data Integration:

For the purpose of better analysis, we need to integrate these two data into one complete dataset.

Binding two datasets into one:

Code:

data <- rbind(real, barca)

View(data)

Output:

Jérémy Mathieu	D	39	1.88	82	France	14	1	1	0	5	1	11	4	0	0	1
Marlon	D	27	1.83	78	Brazil	2	0	0	0	0	0	1	1	0	0	0
Borja López	D	28	1.93	78	Spain	0	0	0	0	0	0	0	0	0	0	0
Ivan Rakitic	M	34	1.83	78	Croatia	32	6	8	5	48	17	29	41	5	0	13
Sergio Busquets	M	34	1.88	76	Spain	33	0	0	3	4	0	41	35	9	0	3
Denis Suárez	M	28	1.75	68	Spain	26	14	1	3	14	7	16	21	1	0	4
Arda Turan	M	35	1.78	76	Turkey	18	4	3	3	14	5	30	22	1	0	6
Andrés Iniesta	M	38	1.70	68	Spain	23	10	0	3	18	8	17	31	2	0	3
Rafinha	M	30	1.75	71	Brazil	18	4	6	2	25	10	9	33	1	0	8
Javier Mascherano	M	38	1.75	73	Argentina	25	5	1	3	10	2	21	19	6	0	4
André Gomes	M	29	1.88	83	Portugal	30	13	3	1	19	7	40	17	3	0	4

For a better understanding of the players, we integrate a new column named Performance, which is the sum of the goals and assists of each individual player.

A new column named Performance which is the sum of Goal and Assist,

Code:

```
new <- data %>%  mutate(Performace = data$G + data$A)
data <- data.frame(new)
```

Output:

Performace
0
6
12
0
15
12
0
21
120
0
33

Then we try to categorize the age into a new variable to have a better understanding of the players condition.

A new Column categorizing the age in which age less than 23 is categorized as 1, age less than 36 is categorized as 2, and age greater than or equal to 36 is categorized as 3,

Code:

```
mew <- data %>%  mutate(AgeCat = case_when(
  data$Age < 23 ~ "1",
  data$Age < 36 ~ "2",
  data$Age >= 36 ~ "3"

))

 data <- data.frame(mew)
```

Output:

e	POS	Age	HT	WT	NAT	APP	SUB	G	A	SH	ST	FC	FA	YC	RC	Performace	AgeCat
aël Varane	D	29	1.91	1.91	France	78	9	0	0	36	6	72	27	12	3	0	2
	D	39	1.88	1.88	Portugal	63	0	3	3	36	18	51	69	15	0	6	3
o Ramos	D	36	1.83	1.83	Spain	69	0	6	6	57	12	102	78	21	6	12	3
o	D	32	1.80	1.80	Spain	48	12	0	0	15	6	51	42	12	0	0	2
elo	D	34	1.75	1.75	Brazil	90	6	6	9	57	12	48	102	6	0	15	2
Carvajal	D	30	1.73	1.73	Spain	66	9	0	12	24	0	78	78	18	0	12	2
o Arbeloa	D	39	1.83	1.83	Spain	18	12	0	0	0	0	18	3	3	0	0	3
o	D	31	1.83	1.83	Brazil	72	3	6	15	54	18	108	57	15	0	21	2
p Lienhart	D	26	1.88	1.88	Austria	0	0	0	120	0	0	0	0	0	0	120	2
o Tejero	D	26	1.75	1.75	Spain	0	0	0	0	0	0	0	0	0	0	0	2
Kroos	M	32	1.83	1.83	Germany	96	0	3	30	60	12	102	141	9	0	33	2

3. Data Transformation

In this phase, we need to transform some variables for better analysis of the dataset.

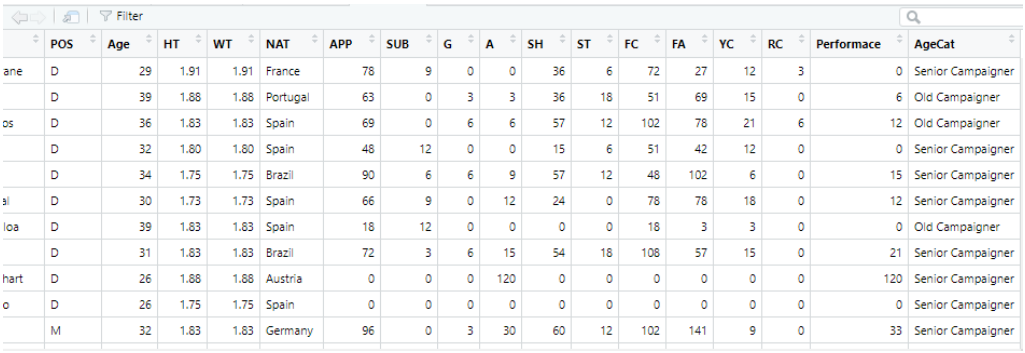
We need to transform the variables such as pos, HT, WT, NAT, AgeCat.

Code:

```
data$POS <- factor(data$POS, ordered = TRUE)

 data$HT <- as.numeric(data$HT)
 data$WT <- as.numeric(data$WT)

 data$NAT <- factor(data$NAT, ordered = TRUE)
```

```
data$Team <- factor(data$Team, ordered = TRUE)
data$AgeCat <- factor(data$AgeCat,
                      levels =c(1,2,3),labels=c("Young Campaigner","Senior Campaigner","Old Campaigner"))
```

Output:



	POS	Age	HT	WT	NAT	APP	SUB	G	A	SH	ST	FC	FA	YC	RC	Performace	AgeCat
ane	D	29	1.91	1.91	France	78	9	0	0	36	6	72	27	12	3	0	Senior Campaigner
	D	39	1.88	1.88	Portugal	63	0	3	3	36	18	51	69	15	0	6	Old Campaigner
os	D	36	1.83	1.83	Spain	69	0	6	6	57	12	102	78	21	6	12	Old Campaigner
	D	32	1.80	1.80	Spain	48	12	0	0	15	6	51	42	12	0	0	Senior Campaigner
	D	34	1.75	1.75	Brazil	90	6	6	9	57	12	48	102	6	0	15	Senior Campaigner
al	D	30	1.73	1.73	Spain	66	9	0	12	24	0	78	78	18	0	12	Senior Campaigner
loa	D	39	1.83	1.83	Spain	18	12	0	0	0	0	18	3	3	0	0	Old Campaigner
	D	31	1.83	1.83	Brazil	72	3	6	15	54	18	108	57	15	0	21	Senior Campaigner
hart	D	26	1.88	1.88	Austria	0	0	0	120	0	0	0	0	0	0	120	Senior Campaigner
o	D	26	1.75	1.75	Spain	0	0	0	0	0	0	0	0	0	0	0	Senior Campaigner
	M	32	1.83	1.83	Germany	96	0	3	30	60	12	102	141	9	0	33	Senior Campaigner

Some of the column names are pretty hard to understand, for this reason, we need to change some of the column names for understanding the database more thoroughly.

Changing some of the column names,
Code:

```
data<- rename(data, "Height(m)"=HT)

data<- rename(data, "Weight(kg)"="Weight(m)")
data<-rename(data, "Goal"=G)
data<-rename(data, "Assists"=A)
data<- rename(data, "RED"=RC)
data<-rename(data, "Yellow"=YC)
```

Output:

Filter

	Name	POS	Age	Height(m)	Weight(kg)	NAT	APP	SUB	Goal	Assists	SH	ST	FC	FA	Yellow
1	Raphaël Varane	D	29	1.91	81	France	78	9	0	0	36	6	72	27	12
2	Pepe	D	39	1.88	81	Portugal	63	0	3	3	36	18	51	69	15
3	Sergio Ramos	D	36	1.83	82	Spain	69	0	6	6	57	12	102	78	21
4	Nacho	D	32	1.80	76	Spain	48	12	0	0	15	6	51	42	12
5	Marcelo	D	34	1.75	73	Brazil	90	6	6	9	57	12	48	102	6
6	Dani Carvajal	D	30	1.73	73	Spain	66	9	0	12	24	0	78	78	18
7	Álvaro Arbeloa	D	39	1.83	78	Spain	18	12	0	0	0	0	18	3	3
8	Danilo	D	31	1.83	78	Brazil	72	3	6	15	54	18	108	57	15
9	Philipp Lienhart	D	26	1.88	83	Austria	0	0	0	0	0	0	0	0	0
10	Álvaro Tejero	D	26	1.75	68	Spain	0	0	0	0	0	0	0	0	0
11	Toni Kroos	M	32	1.83	76	Germany	96	0	3	30	60	12	102	141	9

4. Data Reduction:

In our dataset, we can see that some columns are not necessary for analysis. So we remove those columns from the dataset.

Code:

```
data <- subset(data, select = -c(ST))
data <- subset(data, select = -c(SH))
```

Output:

ht(kg)	NAT	APP	SUB	Goal	Assists	Fouls Commited	Fouls Suffered	Yellow	RED	Perfor
81	France	78	9	0	0	72	27	12	3	
81	Portugal	63	0	3	3	51	69	15	0	
82	Spain	69	0	6	6	102	78	21	6	
76	Spain	48	12	0	0	51	42	12	0	
73	Brazil	90	6	6	9	48	102	6	0	
73	Spain	66	9	0	12	78	78	18	0	
78	Spain	18	12	0	0	18	3	3	0	
78	Brazil	72	3	6	15	108	57	15	0	
83	Austria	0	0	0	0	0	0	0	0	
68	Spain	0	0	0	0	0	0	0	0	
76	Germany	96	0	3	30	102	141	9	0	

5. **Data Discretization:**
   No discretization is needed for this dataset as it is already in a better shape. So we skip this process and move on to descriptive statistics.

Descriptive Statistics:

Now, we are going to compute various descriptive statistics parameters for our dataset.

Firstly, let's try to inspect the central tendency for the various variables of our dataset.

- **MEAN:**
  Mean of all player's ages, weights and heights,

  Code:

```
MeanAge <- mean(data$Age)
MeanAge

meanheight <- mean(data$Height.m.)
meanheight

meanweight <- mean(data$Weight.kg.)
meanweight
```

  Output:

```
> meanweight <- mean(data$Weight.kg.)
> meanweight
[1] 75.41304
> MeanAge <- mean(data$Age)
> MeanAge
[1] 31.6087
>
> meanheight <- mean(data$Height.m.)
> meanheight
[1] 1.806522
>
> meanweight <- mean(data$Weight.kg.)
> meanweight
[1] 75.41304
>
```

- **MEDIAN:**
  Now we calculate the median for the amount of fouls committed and fouls suffered,

  Code:
  1)
  l <- sort(data$Fouls.Commited)
  l <-median(l)
  l

  2)
  median(data$Fouls.Suffered)
  Output:

```
> median(data$Fouls.Commited)
[1] 31.5
>
>
> median(data$Fouls.Suffered)
[1] 32
>
```

- **MODE:**

As the mode doesn’t have a built-in function, we first implement the function.

Code:

```
mode <- function(x){
  unique_values <- unique(x)
  table <- tabulate(match(x, unique_values))
  unique_values[table == max(table)]
}

mode(data$NAT)
```

Output:

```
> mode <- function(x){
+   unique_values <- unique(x)
+   table <- tabulate(match(x, unique_values))
+   unique_values[table == max(table)]
+ }
>
> mode(data$NAT)
[1] Spain
```

Range:

Now we calculate the range of variables.

Code:

```
rgoal <- max(data$Goal) - min(data$Goal)
rgoal
rapp <- max(data$APP) - min(data$APP)
rapp

rfoulc <- max(data$Fouls.Commited)- min(data$Fouls.Commited)
rfoulc

rfouls <- max(data$Fouls.Suffered)- min(data$Fouls.Suffered)
rfouls
```

Output:

```
> rgoal <- max(data$Goal) - min(data$Goal)
> rgoal
[1] 105
> rapp <- max(data$APP) - min(data$APP)
> rapp
[1] 108
>
> rfoulc <- max(data$Fouls.Commited)- min(data$Fouls.Commited)
> rfoulc
[1] 117
>
> rfouls <- max(data$Fouls.Suffered)- min(data$Fouls.Suffered)
> rfouls
[1] 171
>
```

Quartile & Percentile:

Here we find the Quartile & Percentile

Code:

```
quantile(data$Age, prob = c(0.0,0.25,0.50, 0.75 , 0.100))
quantile(data$Weight.kg., prob = c(0.0,0.25,0.50, 0.75 , 0.100))
quantile(data$Yellow)
```

Output:

```
> quantile(data$Age, prob = c(0.0,0.25,0.50, 0.75 , 0.100))
   0%   25%   50%   75%   10%
24.00 29.00 31.00 34.75 26.50
> quantile(data$Weight.kg., prob = c(0.0,0.25,0.50, 0.75 , 0.100))
   0%   25%   50%   75%   10%
66.00 72.00 76.00 80.25 68.00
> quantile(data$Yellow)
  0%  25%  50%  75% 100%
   0    1    5    9   21
```

Interquartile Range:

Code:

```
IQR(data$Age)
```

Output:

```
> IQR(data$Age)
[1] 5.75
```

Variance:

To calculate the variance in R, use the var () function.

Code:

```
var(data$Age)
var(data$Height.m.)
var(data$Weight.kg.)
```

Output:

```
> var(data$Age)
[1] 16.28792
> var(data$Height.m.)
[1] 0.003707633
> var(data$Weight.kg.)
[1] 28.78116
```

Standard Deviation:

To compute the standard deviation, we use the sd () function. The sd () function calculates the standard deviation of the values in the input R object.

Code:

```
sd(data$Age)
sd(data$Height.m.)
sd(data$Weight.kg.)
```

Output:

```
> sd(data$Age)
[1] 4.03583
> sd(data$Height.m.)
[1] 0.06089033
> sd(data$Weight.kg.)
[1] 5.364807
>
```

Here by computing dispersion, we can say that the values for the Age, Height, and Weight are closely clustered around the mean.

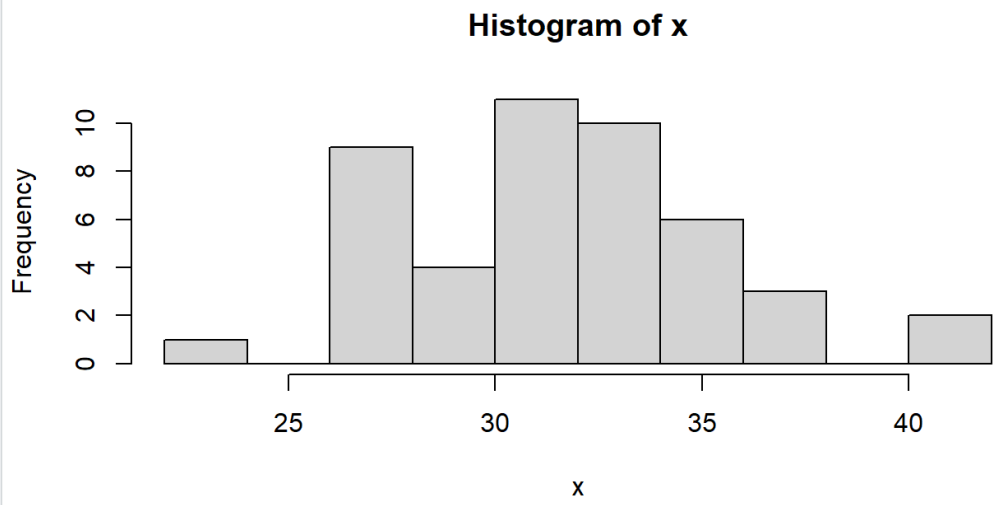
Normal Distribution:

Code:

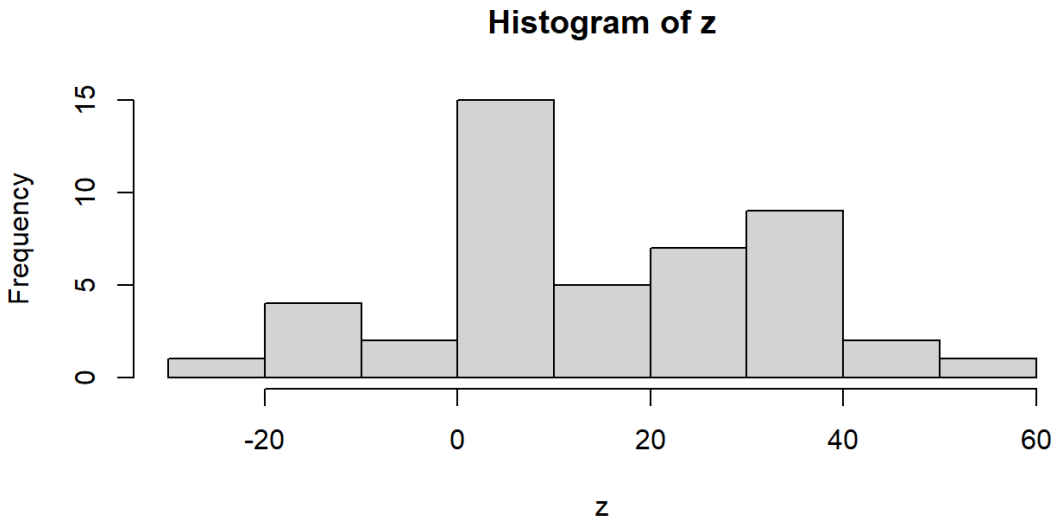
```
1)  x = rnorm(data$Age, mean = mean(data$Age), sd=
          sd(data$Age))
hist(x)

2)  z = rnorm(data$Goal, mean = mean(data$Goal),sd = sd(data$Goal)
    )
hist(z)

3)  y = dnorm(data$APP , mean = mean(data$APP), sd=
    sd(data$APP))
plot(data$APP,y)
```

Output:

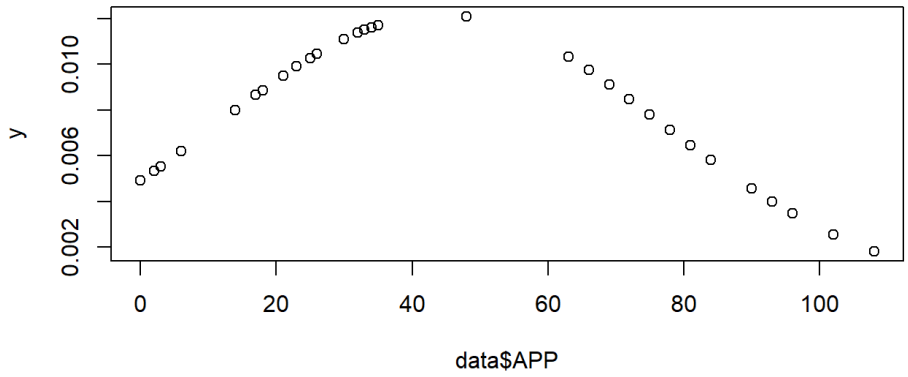
1)


Histogram of x
Frequency
0
2
4
6
8
10
25
30
35
40
x

2)


Histogram of z
Frequency
0
5
10
15
-20
0
20
40
60
z

3)



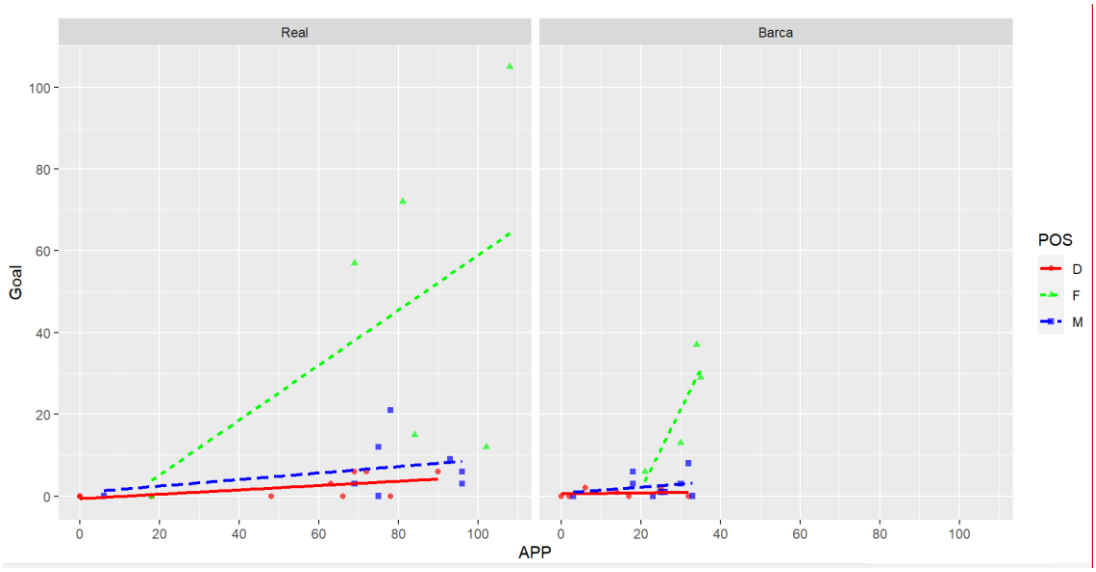
Data Visualization:

1) First lets draw a scatter plot of Appearance vs Goal for each team,

Code:

```
ggplot(data, aes(x = APP, y= Goal, shape = POS,color=POS, linetype = POS))+
  geom_point(alpha = 0.7)+
  geom_smooth(method =lm, se= FALSE)+
  scale_x_continuous(breaks = seq(0,150,20))+
  scale_y_continuous(breaks = seq(0,150,20))+
  scale_color_manual(values = c("red","green","blue"))+
  facet_wrap(~Team)
```

Output:



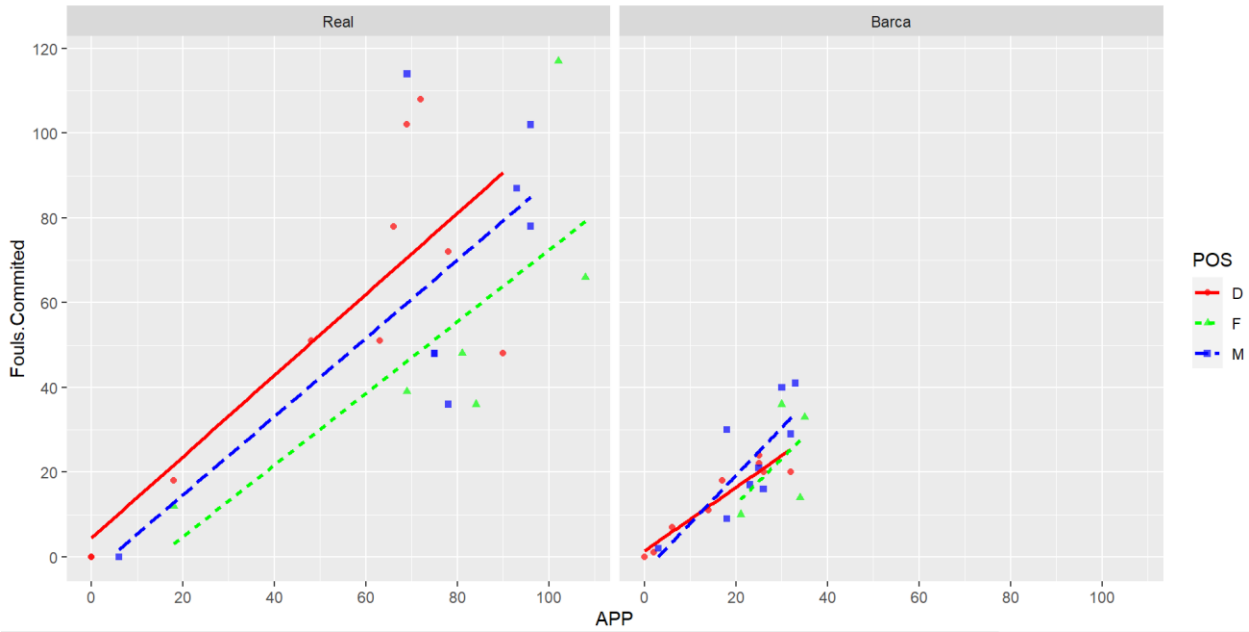
From this scatter plot, we can understand that the player with more appearances started to score more goals. In the Barca side, the forward with more appearances started to deliver more goals, and in the Real Madrid side, forwards started to show extra ordinary numbers with more appearances.

2) Now we see a scatter plot for Defenders Appearance vs Fouls Committed,

Code:

```
ggplot(data, aes(x = APP, y= Fouls.Commited, shape = POS,color=POS,
linetype = POS))+
  geom_point(alpha = 0.7)+
  geom_smooth(method =lm, se= FALSE)+
  scale_x_continuous(breaks = seq(0,150,20))+
  scale_y_continuous(breaks = seq(0,150,20))+
  scale_color_manual(values = c("red","green","blue"))+
  facet_wrap(~Team)
```

Output:



In this plot, we can see that with more appearances, Real Madrid's defenders started to be more aggressive than Barcelona's Defenders. But most of the attacks of the Barca side come from the Midfielders.

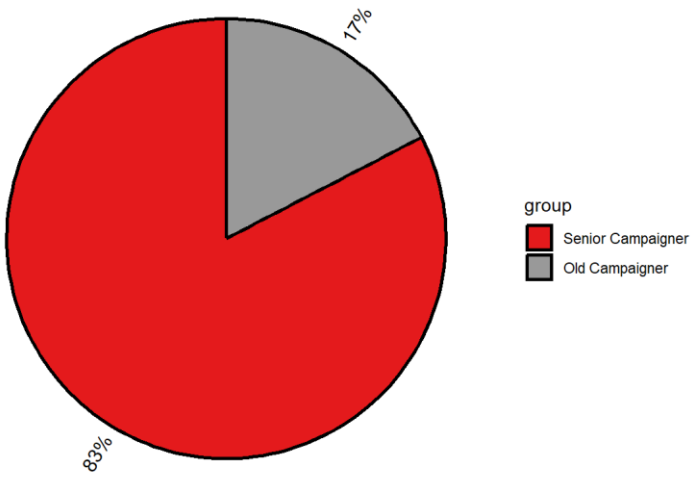
3) Next, we try to measure and analyze the age categories that the players belong to:

Code:

```
library(ggpie)
library(dplyr)
```

```
data %>% ggpie(group_key = "AgeCat",count_type = "full", label_type =
"circle",
               label_info = "ratio", label_pos = "out", nudge_x = 10)
```

Output:



In this pie chart we can see that the majority of the players belong to the senior campaigner category. This means both teams are filled with experienced players,

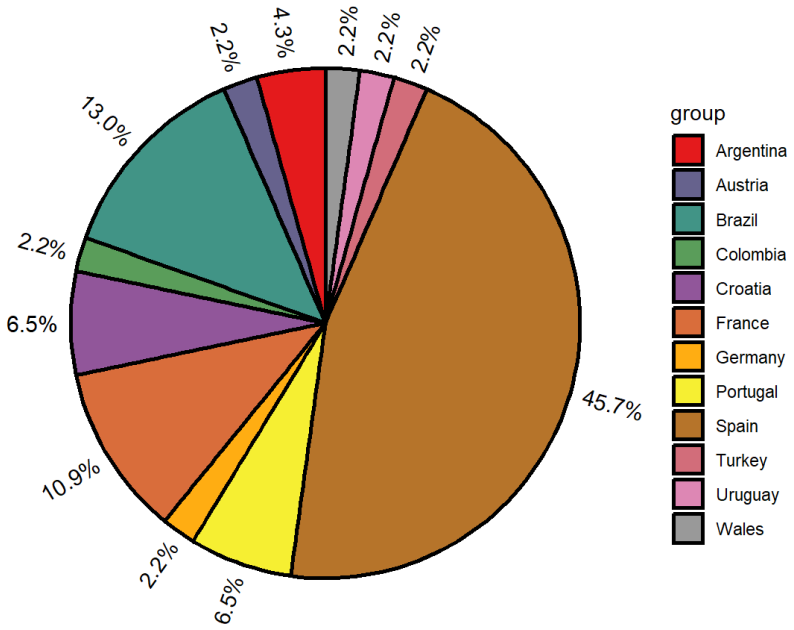
4) Furthermore, we try to identify the most number of players from and individual country.

Player By Country,

Code:

```
data %>% ggpie(group_key = "NAT",count_type = "full", label_type = "circle",
               label_info = "ratio", label_pos = "out", nudge_x = 10)
```

Output:



From the pie chart we can identify that, most of the players are from Spain which is 45.7%. The next most population is from Brazil. We also get some ideas of their playing style, as most of the players are from Spain, they prefer tiki taka.

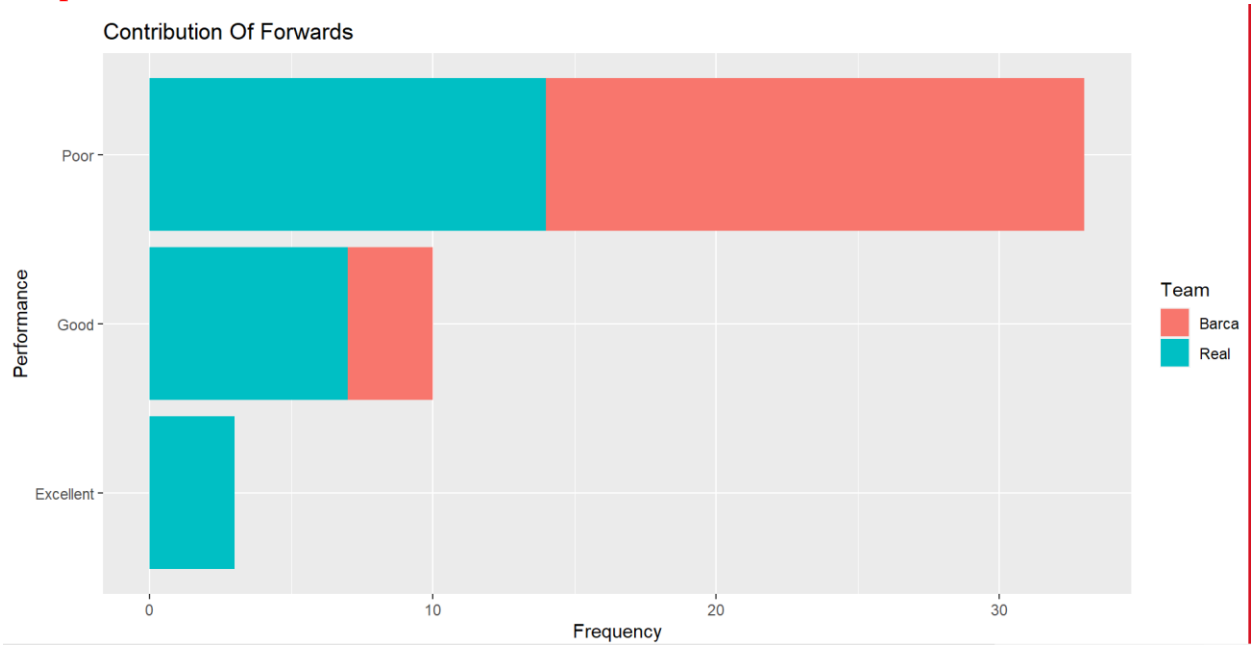
5) Now the most important part of the visualization. We need to see the contribution of the forwards for their respective teams.

Team Performance

Code:

```
ggplot(data,aes(x=Performance, fill=Team))+
  geom_bar()+
  labs(title = "Contribution Of Forwards", x ="Performance",
y="Frequency")+
    coord_flip()
```

Output:



6) Now we run a comparison between Barca and Real's two most prolific players, CRISTIANO RONALDO and LIONEL MESSI

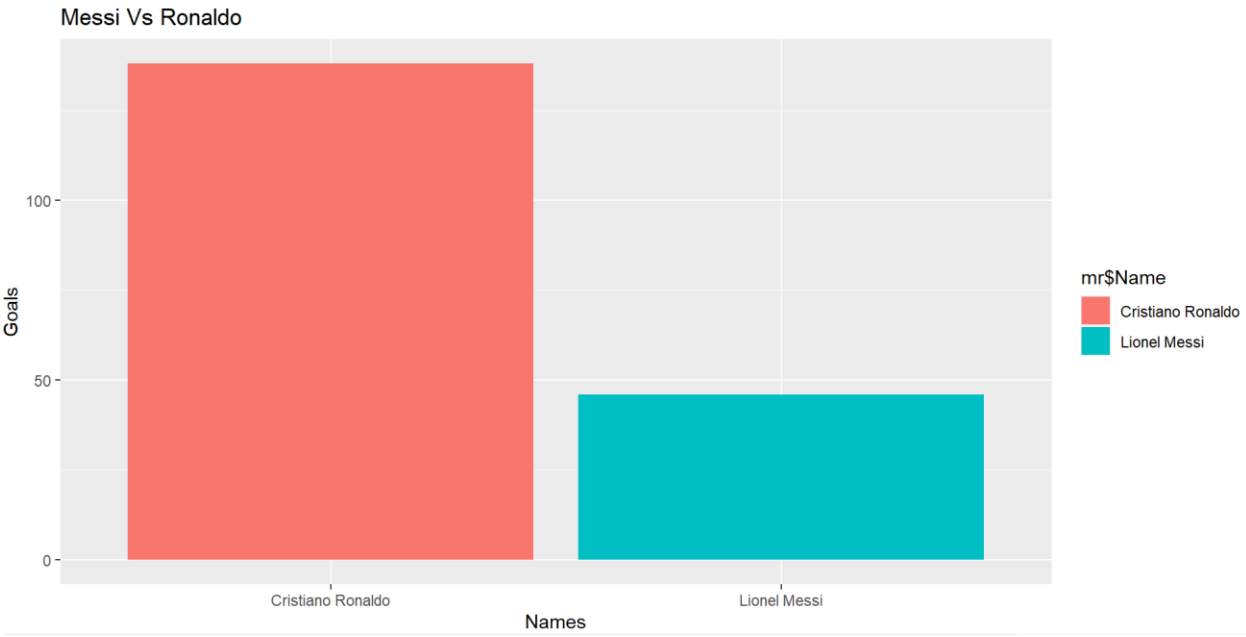
Goals between Messi & Ronaldo,

Code:

```
messi <- data[(data$Name=="Lionel Messi"),]
messi
```

```
ron <- data[(data$Name=="Cristiano Ronaldo"),]
ron

mr <- rbind(messi,ron)
mr
g=(mr$Goal+mr$Assists)
ggplot(mr,aes(x= mr$Name, y= g, fill= mr$Name))+
  geom_bar(stat = "identity")+
  labs(x="Names",y="Goals", title = "Messi Vs Ronaldo")
```

Output:



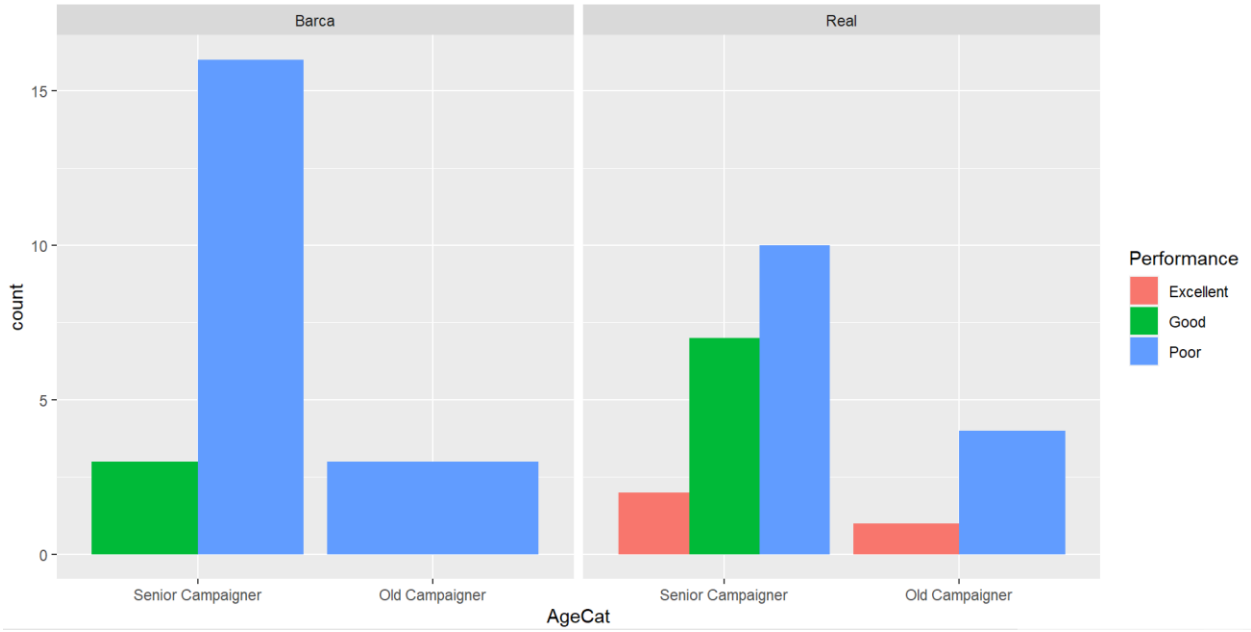
In this bar graph, we see that the contribution of goals from Ronaldo is far Greater then Lionel Messi in that season. As Ron scored over 100 goals and Messi scored near 50.

7) Now we visualize the performance of senior and old campaigners,

Code:

```
ggplot(data, aes(x= AgeCat, fill= Performance))+
  geom_bar(position = "dodge")+
  facet_wrap(~Team)
```

Output:



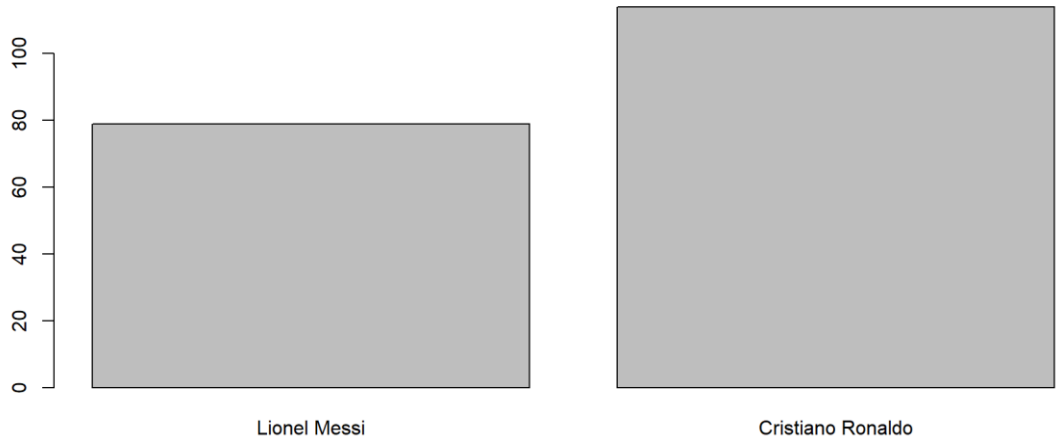
For Barca, the senior campaigners put on a good performance and in the Real dugout both senior and old campaigners shower some excellent performance in the display of football.

8) Most fouls suffered between Messi and Ronaldo

Code:

```
barplot(mr$Fouls.Suffered, names.arg = mr$Name)
```

Output:



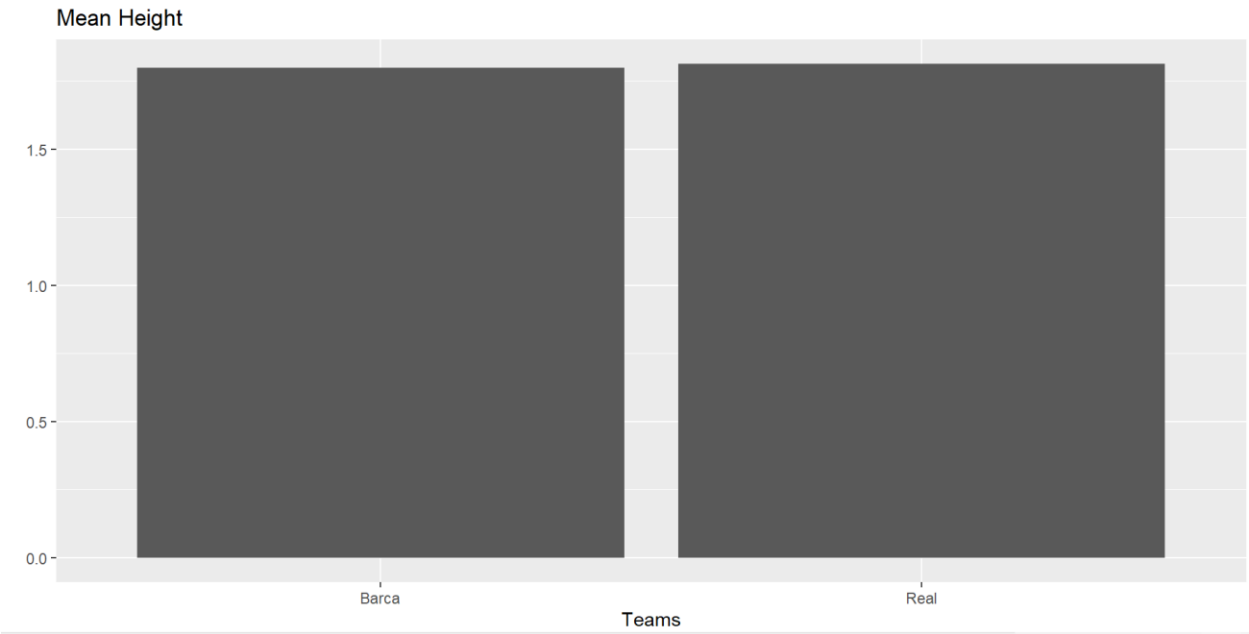
In this bar graph, it is clear that Ronaldo was the most fouled player among their rivals and from the previous graphs we saw that he also had an astonishing performance, showing why he got the Balon d'or that year.

9)
We visualize the minimum height of both teams through a bar plot:

Code:

```
c = data$Team
plotdata <- data %>%
  group_by(data$Team) %>%
  summarise(mean=mean(Height.m.))
View(plotdata)

plotdata<-rename(plotdata, "Tname"="data$Team")

ggplot(plotdata, aes(x= reorder(Tname, mean), y= mean))+
  geom_bar(stat="identity")+
  labs(x="Teams",y="", title = "Mean Height")
```

Output:

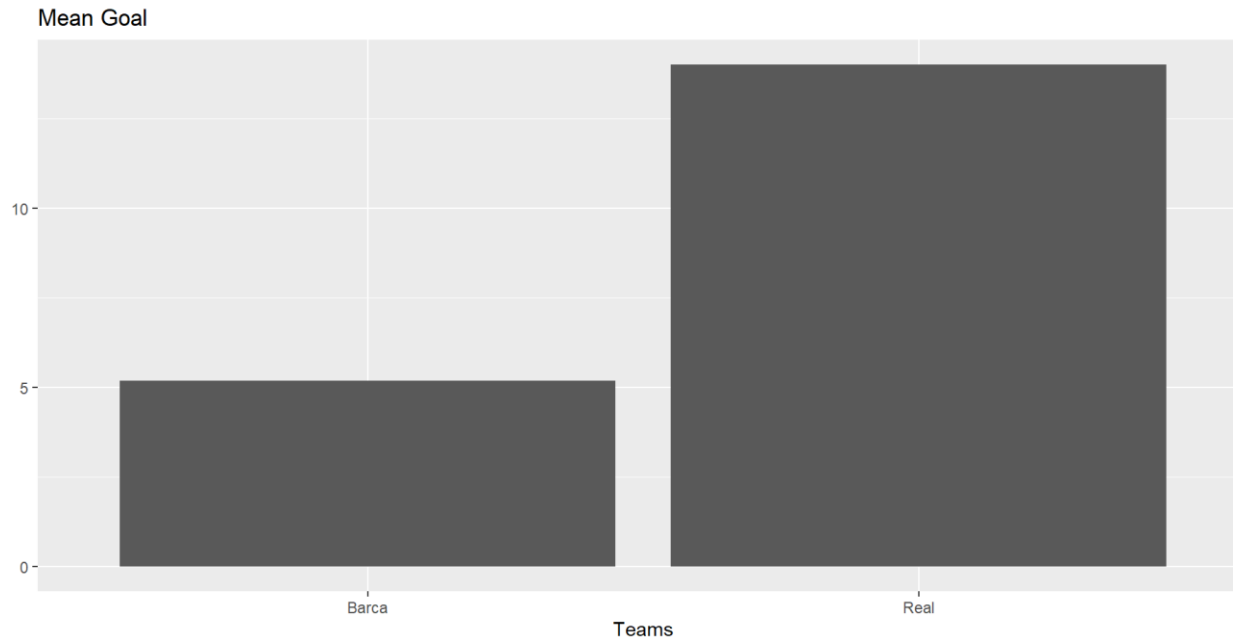


9) Mean Goal

Code:

```
c = data$Team
plotdata2 <- data %>%
  group_by(data$Team) %>%
  summarise(mean=mean(Goal))
View(plotdata2)

plotdata2<-rename(plotdata2, "Tname"="data$Team")
ggplot(plotdata2, aes(x= reorder(Tname, mean), y= mean))+
  geom_bar(stat="identity")
```

Output:



10)

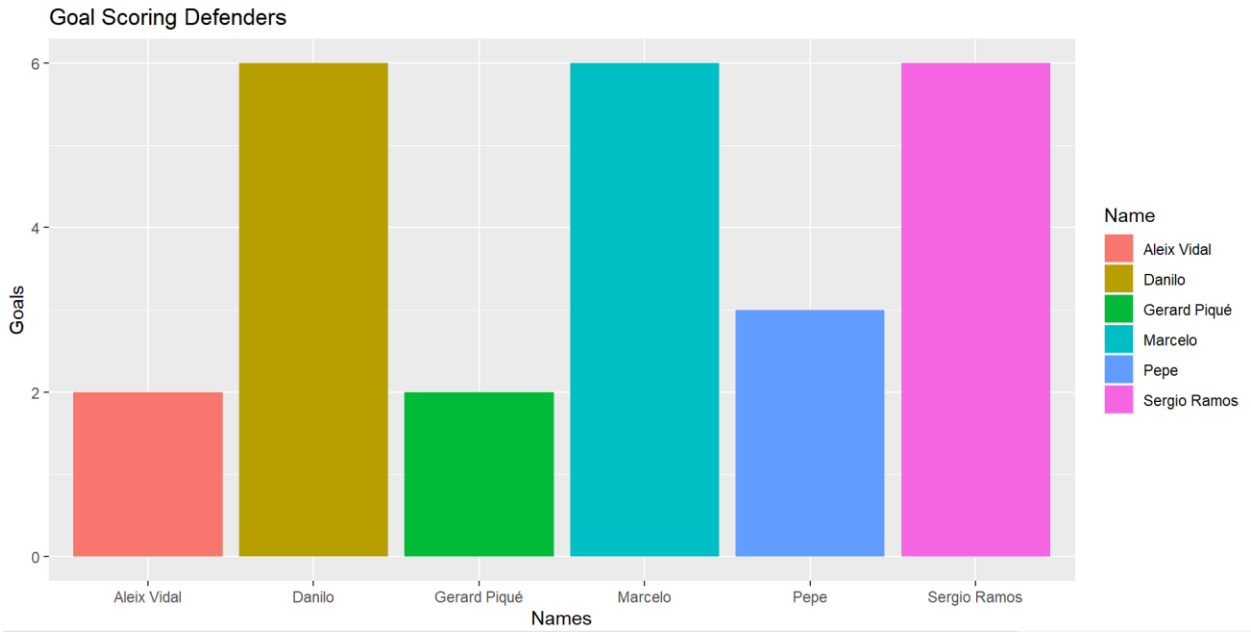
We all love players that can do both which is attack and defend. Here we try to find top goal-scoring defenders among both teams.

Goal Scoring Defenders:

Code:

```
data %>% filter(data$Goal>=2 & data$POS == "D") %>%
  ggplot(aes(x= Name, y= Goal, fill=Name))+
  geom_bar(stat = "identity")+
  labs(x="Names",y="Goals", title = "Goal Scoring Defenders")
```

Output:



Despite being a central defender, Ramos scored 6 goals through out the season. Which is a great number.

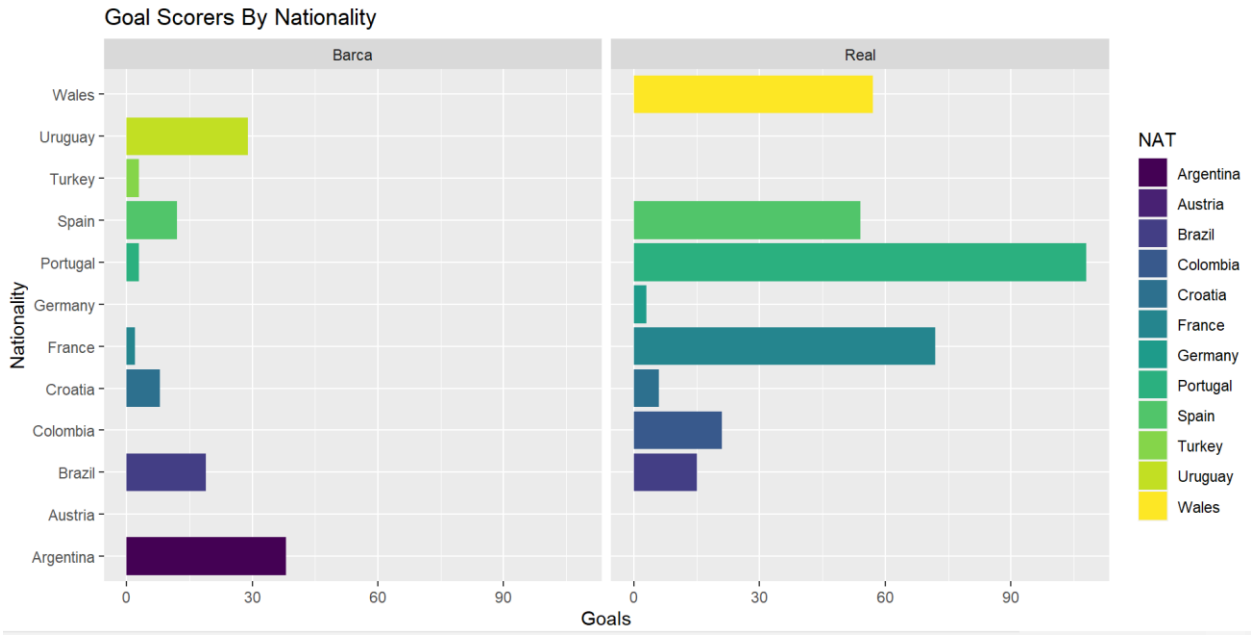
10) Next, we try to figure out the most number of goals scored by countries.

Goal Scorers By Nationality:

Code:

```
data %>% ggplot(aes(x= NAT, y= Goal, fill=NAT))+
  geom_bar(stat = "identity")+
  labs(x="Nationality",y="Goals", title = "Goal Scorers By Nationality")+
  facet_wrap(~Team)+
  coord_flip()
```

Output:



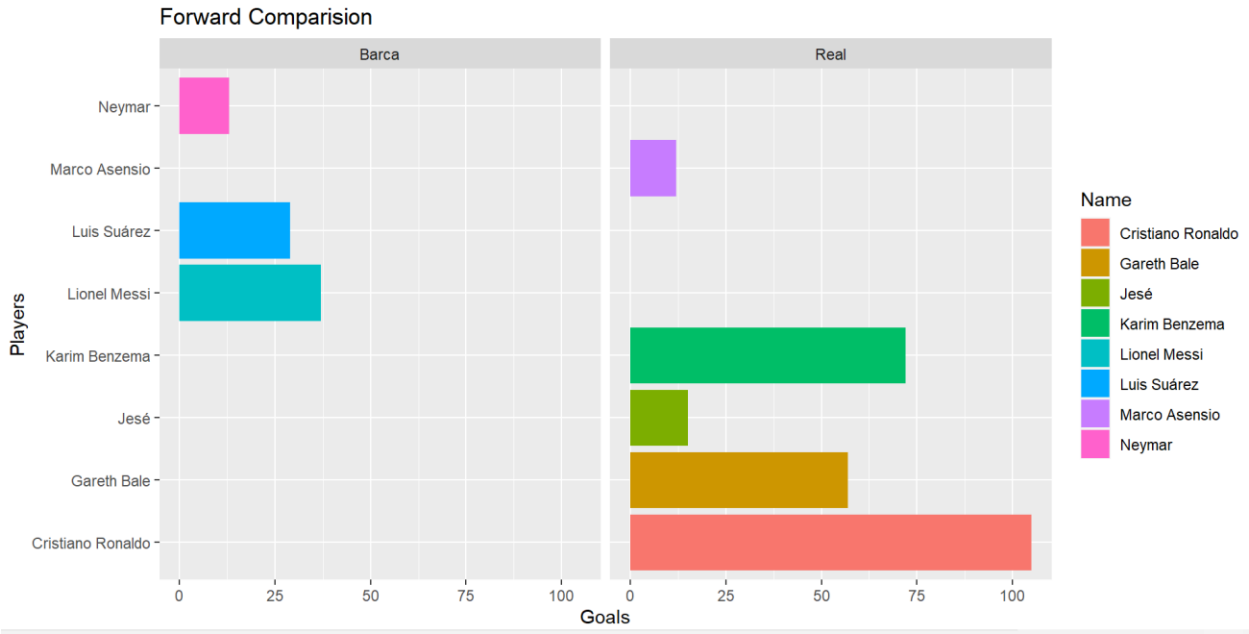
Despite spain having the most amount of players, they do not produce goal scorers. Portugal in this case has the players that scored the most goals for their respected clubs.

11) Now we compare the forward of the club based on goals, Forward Comparison

Code:

```
data %>% filter(data$POS =="F" & data$Goal >mean(data$Goal)) %>%
  ggplot(aes(x= Name, y= Goal, fill=Name))+
  geom_bar(stat = "identity")+
  labs(x="Players",y="Goals", title = "Forward Comparision")+
  facet_wrap(~Team)+
  coord_flip()
```

Output:



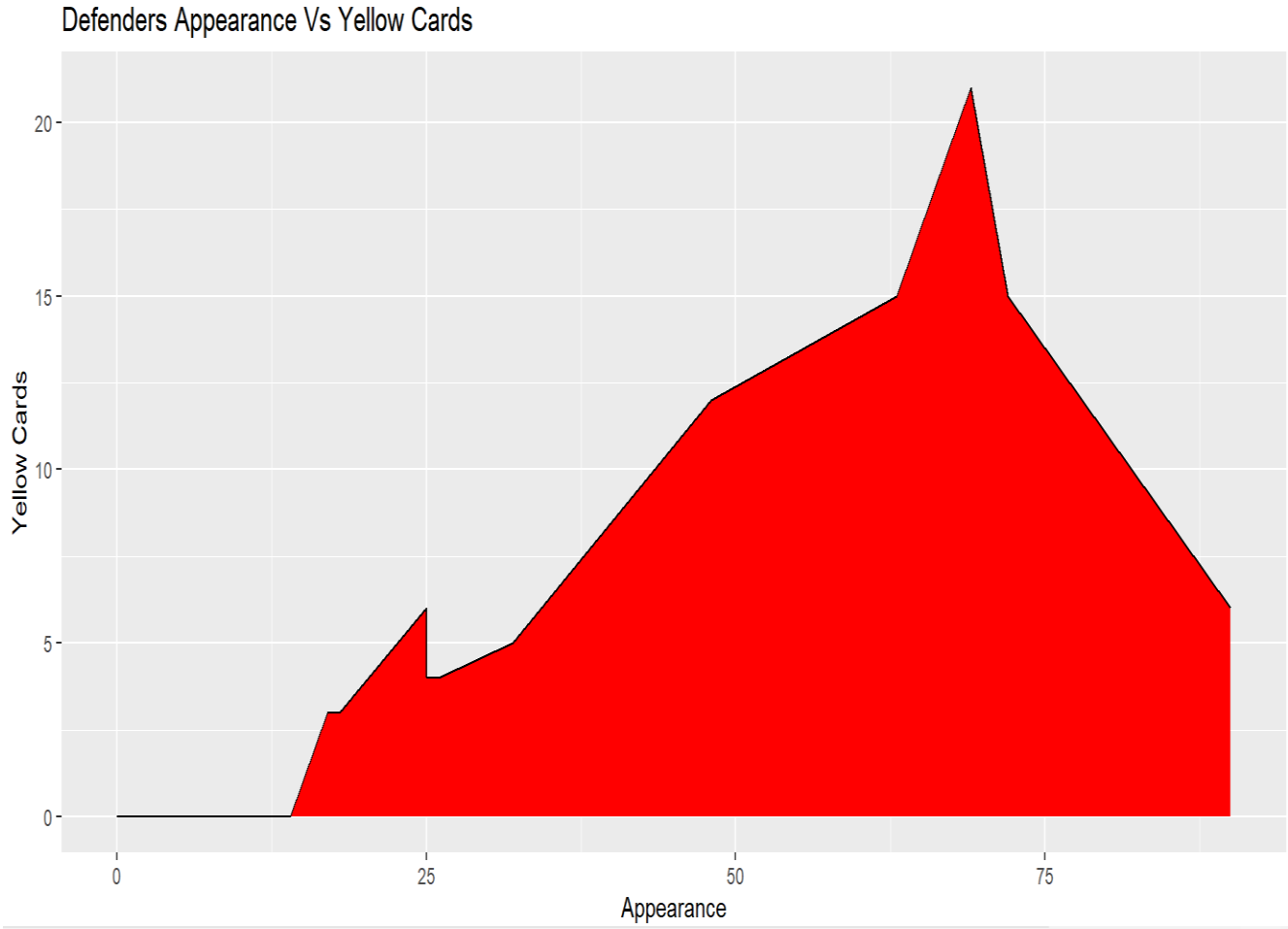
From the plot, we can easily identify that Real Madrid's forwards contributed the most to their teams. They overshadowed the performance of the Barca forwards

12) Density plot of defenders' Appearance vs Yellow Cards

Code:

```
data %>% filter(data$POS =="D") %>% ggplot(aes(x=APP, y= Yellow))+
  geom_density(stat = "identity", fill="red", bw= 0.5)+
  labs(x="Appearance",y="Yellow Cards", title = "Defenders Appearance Vs
Yellow Cards")
```

Output:



Shiny Dashboard Implementation:

For the shiny dashboard implementation, we tried to create a reactive app based on our topic. We tried to show a reactive scatter plot and a bar plot.

We also included About, Data, Structure and Summary sections.

Code:

UI

```
library(shiny)
library(shinydashboard)


dashboardPage(
  dashboardHeader(title= "Dataset of Barcelona and Real Madrid of 15/16 Season",
                  titleWidth = 650),
  dashboardSidebar(
    sidebarMenu(
      id="sidebar",


      menuItem("data", tabName = "data", icon = icon("database")),
      menuItem(text ="Visualization", tabName = "View", icon= icon("char=line")),
     selectInput(inputId = "vv", label ="select" , choices = c1, selected = "NAT"),
```

```
   selectInput(inputId = "var3" , label ="Select the X variable" , choices = c1, selected = "Goal"),

   selectInput(inputId = "var4" , label ="Select the Y variable" , choices = c1, selected = "APP")

    )

  ),

  dashboardBody(


    tabItems(

      #first tab item

      tabItem( tabName = "data",

               #tab box

               tabBox(

                 id="t1", width = 12,

                 tabPanel(title="About",icon=icon("address-card"), h4("The Data set is about the season 15/16 of la liga, where we see the data of the champions Barcelona and Runners Up Real Madrid, Here we do the comparison between these two datasets and then combine them to operate a complete Analysis")),

                 tabPanel(title="Data",icon=icon("address-card"),dataTableOutput("data") ),

                 tabPanel(title="structure",icon=icon("address-card"), verbatimTextOutput("structure")),
```

```
            tabPanel(title="Summary",icon=icon("address-card"),
verbatimTextOutput("summary"))

          )

  ),


  tabItem(tabName = "View",

          tabBox(id= "t2",width = 12,

          tabPanel(title = "Relationship among Goal/Assist and Appearance",
value="ga",

                   radioButtons(inputId ="fit" , label = "Select smooth method" ,
choices = c("loess", "lm"), selected = "lm" , inline = TRUE),

                   plotlyOutput("scatter"),

          side = "left"),

          tabPanel(title = "Player Comparison",
value="player",plotlyOutput("histo"))

  ))

 )

)

)
```

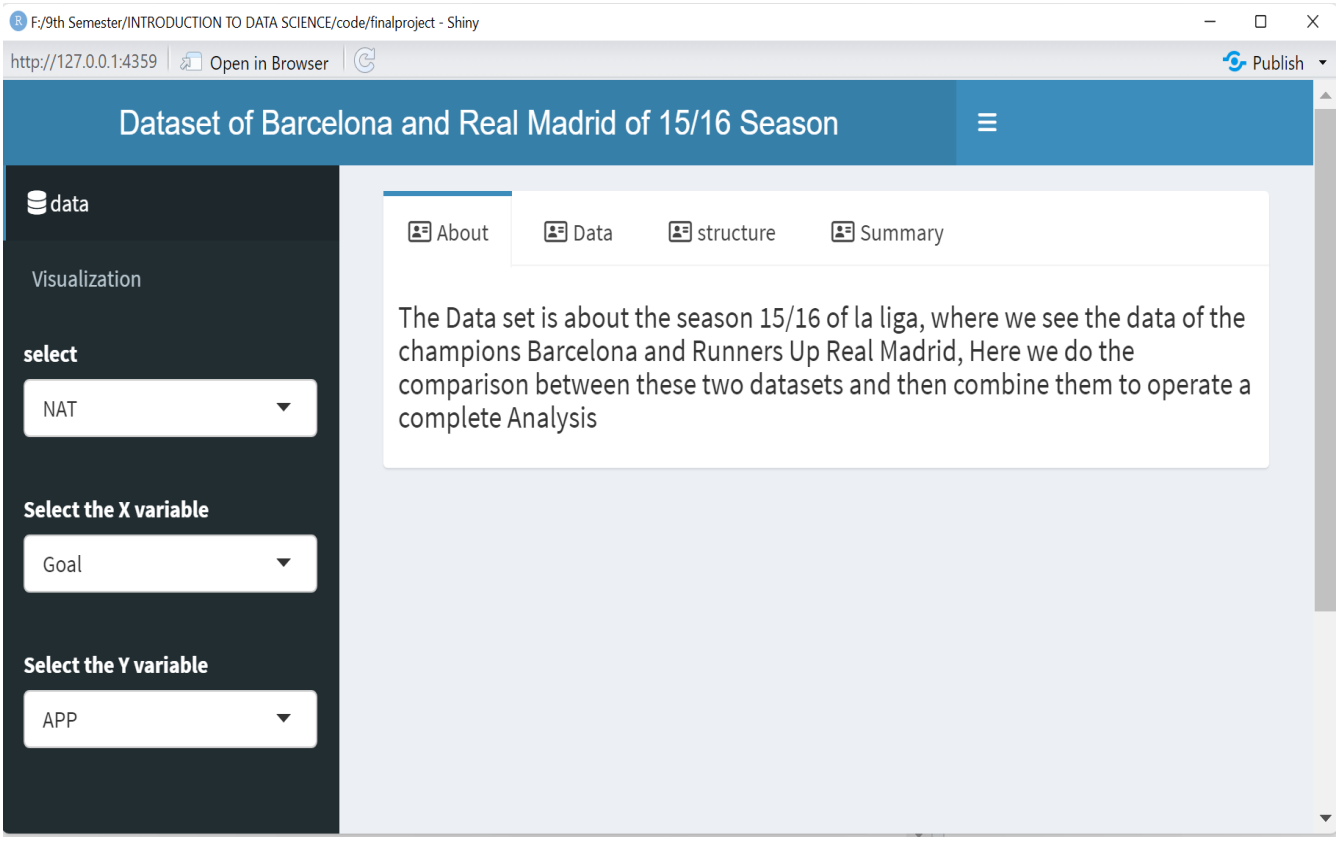
SERVER:

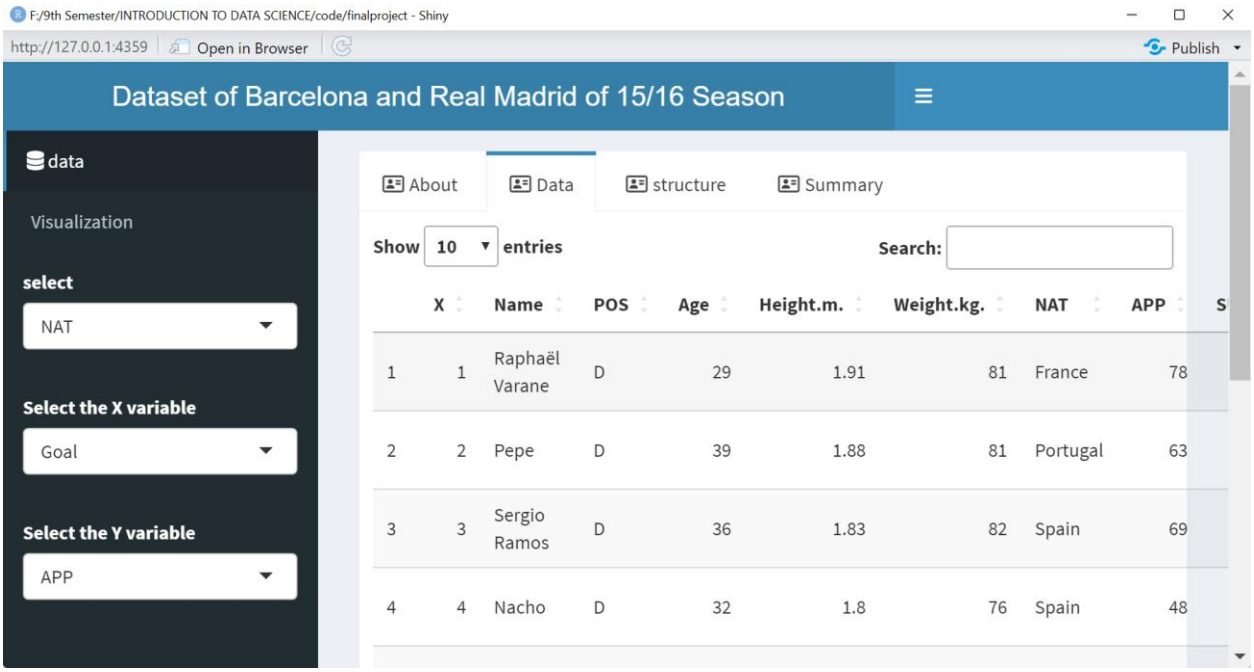
```
library(DT)

function(input,output,session){
#structure
  output$structure <- renderPrint(
    datat %>%
      str()
  )
  #summary

  output$summary <- renderPrint(
    datat %>%
      summary()
  )

  #datatable
```

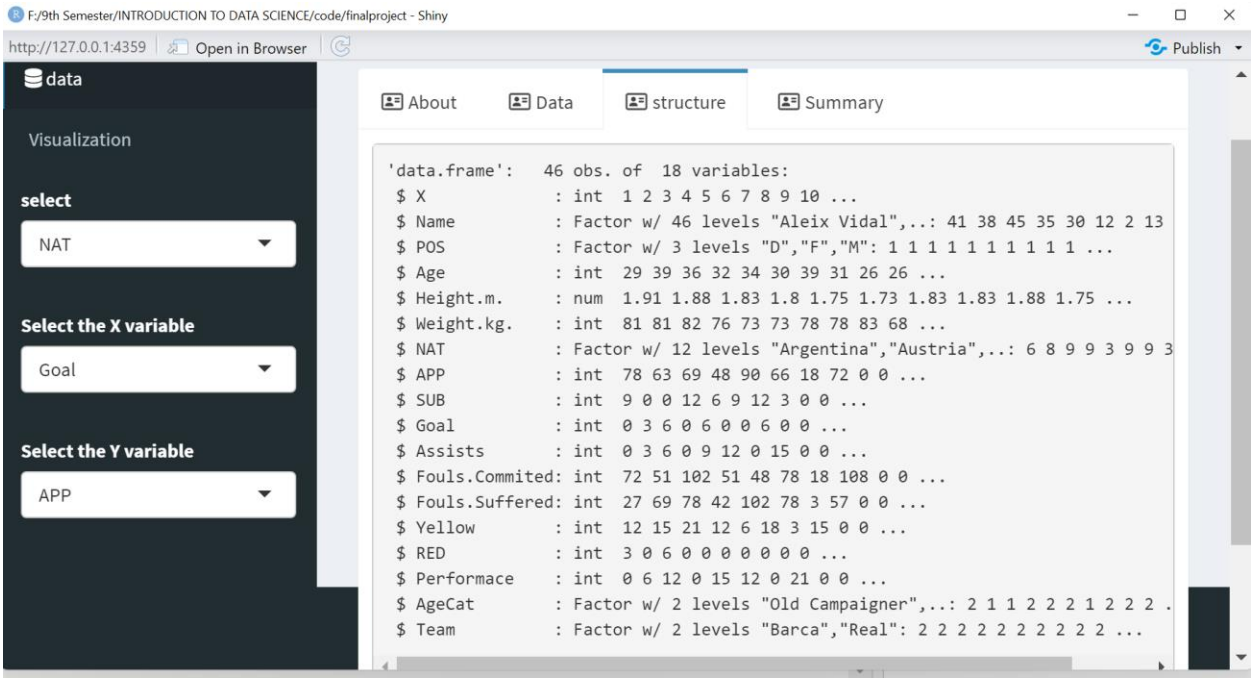
```
output$data <- renderDataTable(
  datat
)

### Scatter Charts
output$scatter <- renderPlotly({
  p = datat %>%
    ggplot(aes(x=get(input$var3), y=get(input$var4))) +
    geom_point() +
    geom_smooth(method=get(input$fit), se= FALSE) +
    labs(title = paste("Relation b/w", input$var3 , "and" , input$var4),
         x = input$var3,
         y = input$var4)
})
#histogram
output$histo <- renderPlot({
  p1 = datat %>%
   ggplot(aes(x = get(input$vv)))+
    geom_bar()
```

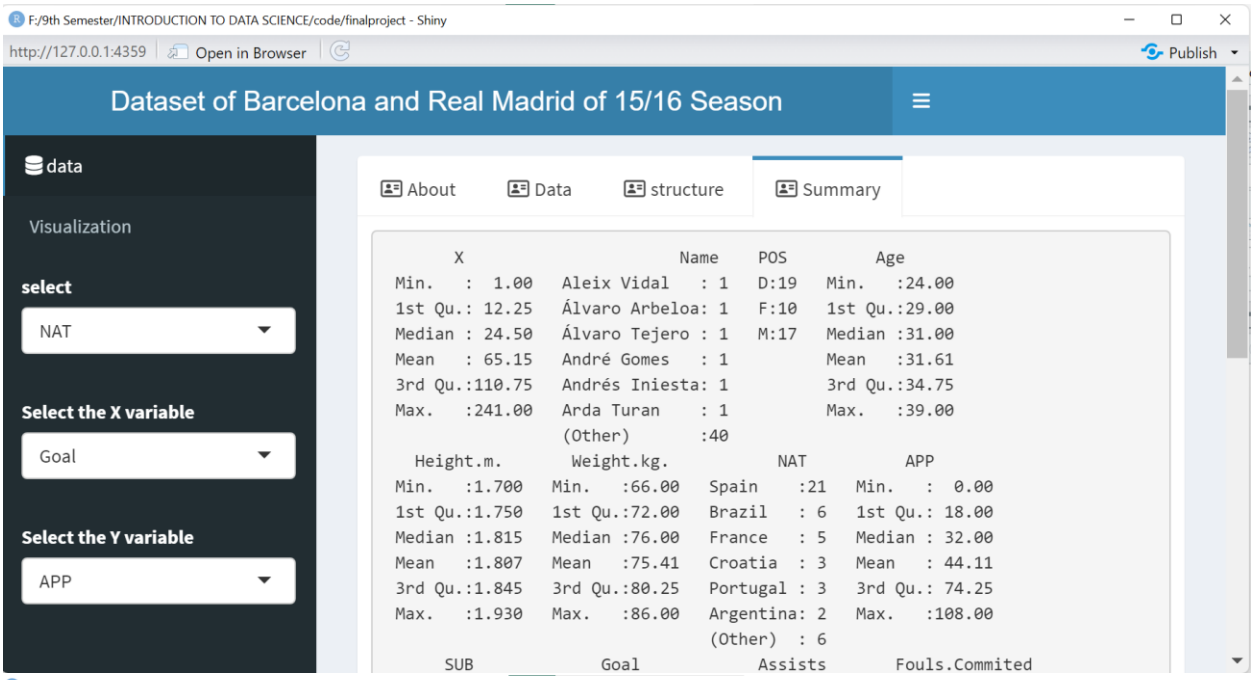
```
})
}
```

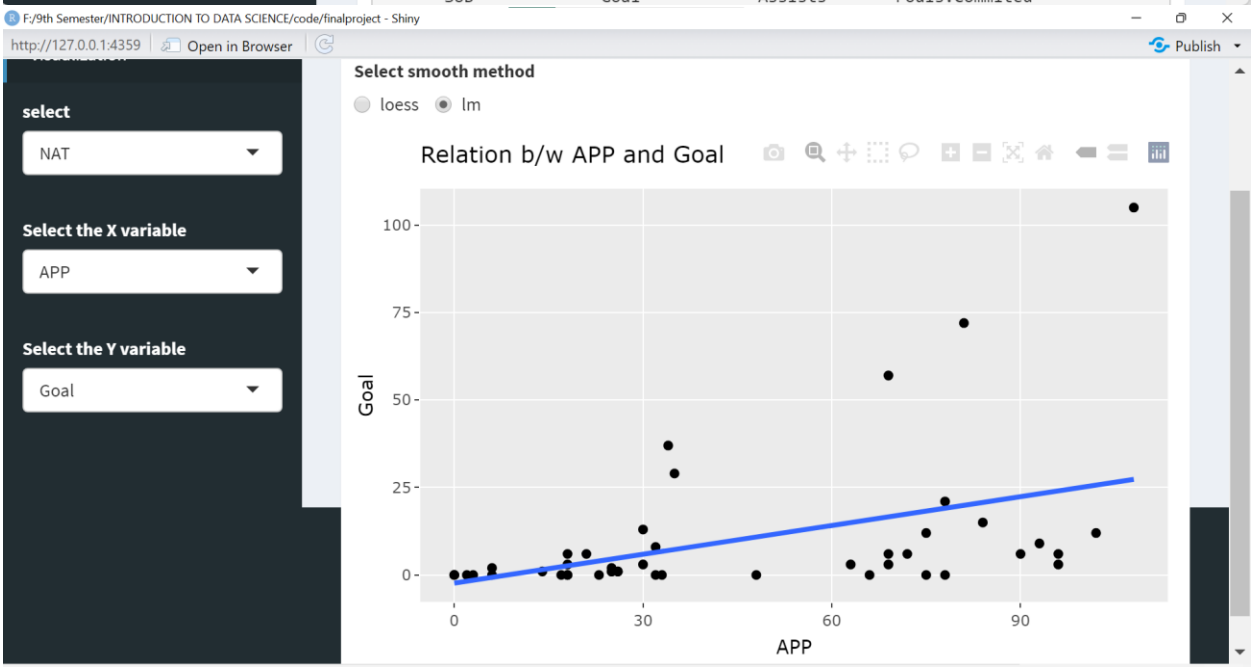
Output:

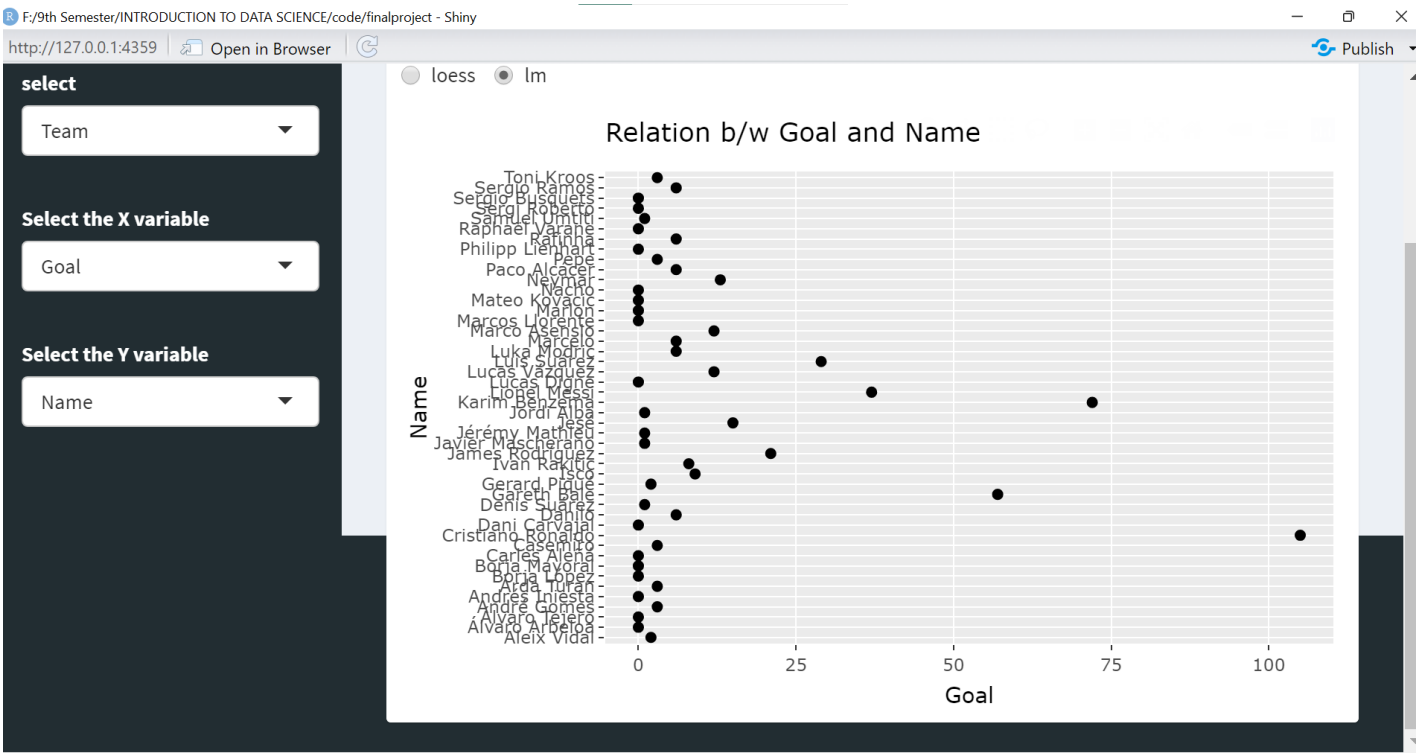

F:/9th Semester/INTRODUCTION TO DATA SCIENCE/code/finalproject - Shiny
http://127.0.0.1:4359
Open in Browser
Publish
Dataset of Barcelona and Real Madrid of 15/16 Season
data
Visualization
select
NAT
Select the X variable
Goal
Select the Y variable
APP
About
Data
structure
Summary
The Data set is about the season 15/16 of la liga, where we see the data of the champions Barcelona and Runners Up Real Madrid, Here we do the comparison between these two datasets and then combine them to operate a complete Analysis


F:/9th Semester/INTRODUCTION TO DATA SCIENCE/code/finalproject - Shiny
http://127.0.0.1:4359
Open in Browser
Publish
Dataset of Barcelona and Real Madrid of 15/16 Season
data
Visualization
select
NAT
Select the X variable
Goal
Select the Y variable
APP
About
Data
structure
Summary
Show 10 entries
Search:
X Name POS Age Height.m. Weight.kg. NAT APP
1 1 Raphaël Varane D 29 1.91 81 France 78
2 2 Pepe D 39 1.88 81 Portugal 63
3 3 Sergio Ramos D 36 1.83 82 Spain 69
4 4 Nacho D 32 1.8 76 Spain 48


F:/9th Semester/INTRODUCTION TO DATA SCIENCE/code/finalproject - Shiny
http://127.0.0.1:4359
Open in Browser
Publish
data
Visualization
select
NAT
Select the X variable
Goal
Select the Y variable
APP
About
Data
structure
Summary
'data.frame': 46 obs. of 18 variables:
$ X : int 1 2 3 4 5 6 7 8 9 10 ...
$ Name : Factor w/ 46 levels "Aleix Vidal",..: 41 38 45 35 30 12 2 13
$ POS : Factor w/ 3 levels "D","F","M": 1 1 1 1 1 1 1 1 1 1 ...
$ Age : int 29 39 36 32 34 30 39 31 26 26 ...
$ Height.m. : num 1.91 1.88 1.83 1.8 1.75 1.73 1.83 1.83 1.88 1.75 ...
$ Weight.kg. : int 81 81 82 76 73 73 78 78 83 68 ...
$ NAT : Factor w/ 12 levels "Argentina","Austria",..: 6 8 9 9 3 9 9 3
$ APP : int 78 63 69 48 90 66 18 72 0 0 ...
$ SUB : int 9 0 0 12 6 9 12 3 0 0 ...
$ Goal : int 0 3 6 0 6 0 0 6 0 0 ...
$ Assists : int 0 3 6 0 9 12 0 15 0 0 ...
$ Fouls.Commited: int 72 51 102 51 48 78 18 108 0 0 ...
$ Fouls.Suffered: int 27 69 78 42 102 78 3 57 0 0 ...
$ Yellow : int 12 15 21 12 6 18 3 15 0 0 ...
$ RED : int 3 0 6 0 0 0 0 0 0 0 ...
$ Performace : int 0 6 12 0 15 12 0 21 0 0 ...
$ AgeCat : Factor w/ 2 levels "Old Campaigner",..: 2 1 1 2 2 2 1 2 2 2 .
$ Team : Factor w/ 2 levels "Barca","Real": 2 2 2 2 2 2 2 2 2 2 ...


F:/9th Semester/INTRODUCTION TO DATA SCIENCE/code/finalproject - Shiny
http://127.0.0.1:4359
Open in Browser
Publish
Dataset of Barcelona and Real Madrid of 15/16 Season
data
Visualization
select
NAT
Select the X variable
Goal
Select the Y variable
APP
About
Data
structure
Summary
X Name POS Age
Min. : 1.00 Aleix Vidal : 1 D:19 Min. :24.00
1st Qu.: 12.25 Álvaro Arbeloa: 1 F:10 1st Qu.:29.00
Median : 24.50 Álvaro Tejero : 1 M:17 Median :31.00
Mean : 65.15 André Gomes : 1 Mean :31.61
3rd Qu.:110.75 Andrés Iniesta: 1 3rd Qu.:34.75
Max. :241.00 Arda Turan : 1 Max. :39.00
(Other) :40
Height.m. Weight.kg. NAT APP
Min. :1.700 Min. :66.00 Spain :21 Min. : 0.00
1st Qu.:1.750 1st Qu.:72.00 Brazil : 6 1st Qu.: 18.00
Median :1.815 Median :76.00 France : 5 Median : 32.00
Mean :1.807 Mean :75.41 Croatia : 3 Mean : 44.11
3rd Qu.:1.845 3rd Qu.:80.25 Portugal : 3 3rd Qu.: 74.25
Max. :1.930 Max. :86.00 Argentina: 2 Max. :108.00
(Other) : 6
SUB Goal Assists Fouls.Commited


F:/9th Semester/INTRODUCTION TO DATA SCIENCE/code/finalproject - Shiny
http://127.0.0.1:4359
Open in Browser
Publish
select
NAT
Select the X variable
APP
Select the Y variable
Goal
Select smooth method
loess
lm
Relation b/w APP and Goal
Goal
APP
0
25
50
75
100
0
30
60
90


F:/9th Semester/INTRODUCTION TO DATA SCIENCE/code/finalproject - Shiny
http://127.0.0.1:4359
Open in Browser
Publish
select
Team
Select the X variable
Goal
Select the Y variable
Name
loess
lm
Relation b/w Goal and Name
Name
Goal
0
25
50
75
100

Discussion and Conclusion:

Our primary task in this project was to perform web scraping on a dataset that contained data on various players from the Barcelona and Real Madrid teams. To gradually improve the data, we made use of a variety of R language structures and techniques. Web scraping is how we begin our project, and in order to obtain the ideal dataset, we have used the data pretreatment procedure as the idea of transforming unclean data into clean data is known as data preparation. We obtained a complete, clean data set to work with after completing the preprocessing, which included data cleaning, data integration, data transformation, data reduction, and data discretization. Then comes data visualization and descriptive statistics. After finishing the assignment, we had a clear understanding of the Real Madrid and Barcelona players. Although the information is gathered from various websites, it is usually done so in raw format, which makes analysis impossible. In this case, data preprocessing is required to turn the raw data into a clean data collection. Here, we have also added information from the clean dataset into a raw dataset that displays the various categories of states according to the density of their urban populations. So, in our opinion, the project has been successful. Our data collection includes many different factors and a large enough number of records to create a respectable dataset. Finally, it can be argued that the main result of this research is to broaden understanding of Web scraping using the R programming language.