TD2 Groupe 5

Axel-Cleris Gailloty

2023-02-10

# 1) Importer le jeu de données dans R

Le bloc de code suivant permet de charger dans R un fichier au format Excel.

library(readxl)  
fifa22 <- read\_excel("fifa22.xlsx")

Nous venons d’importer la base de données dans R en lui donnant le nom fifa22. C’est avec ce nom que nous nous référons à la base de données plus tard dans le reste du code.

# 2) Afficher les noms des colonnes du jeu de données actif

On utilise la fonction colnames(fifa22) pour afficher les noms des colonnes qui sont dans la base de données. Cela nous permet de vérifier que la base de donnée a été importée correctement.

colnames(fifa22)

## [1] "short\_name" "player\_positions"   
## [3] "overall" "potential"   
## [5] "value\_eur" "wage\_eur"   
## [7] "age" "dob"   
## [9] "height\_cm" "weight\_kg"   
## [11] "club\_team\_id" "club\_name"   
## [13] "league\_name" "league\_level"   
## [15] "club\_position" "club\_jersey\_number"   
## [17] "club\_loaned\_from" "club\_joined"   
## [19] "club\_contract\_valid\_until" "nationality\_id"   
## [21] "nationality\_name" "nation\_team\_id"   
## [23] "nation\_position" "nation\_jersey\_number"   
## [25] "preferred\_foot" "weak\_foot"   
## [27] "skill\_moves" "international\_reputation"   
## [29] "work\_rate" "body\_type"   
## [31] "real\_face" "release\_clause\_eur"   
## [33] "player\_traits" "pace"   
## [35] "shooting" "passing"   
## [37] "dribbling" "defending"   
## [39] "physic" "attacking\_crossing"   
## [41] "attacking\_finishing" "attacking\_heading\_accuracy"   
## [43] "attacking\_short\_passing" "attacking\_volleys"   
## [45] "skill\_dribbling" "skill\_curve"   
## [47] "skill\_fk\_accuracy" "skill\_long\_passing"   
## [49] "skill\_ball\_control" "movement\_acceleration"   
## [51] "movement\_sprint\_speed" "movement\_agility"   
## [53] "movement\_reactions" "movement\_balance"   
## [55] "power\_shot\_power" "power\_jumping"   
## [57] "power\_stamina" "power\_strength"   
## [59] "power\_long\_shots" "mentality\_aggression"   
## [61] "mentality\_interceptions" "mentality\_positioning"   
## [63] "mentality\_vision" "mentality\_penalties"   
## [65] "mentality\_composure" "defending\_marking\_awareness"  
## [67] "defending\_standing\_tackle" "defending\_sliding\_tackle"   
## [69] "goalkeeping\_diving" "goalkeeping\_handling"   
## [71] "goalkeeping\_kicking" "goalkeeping\_positioning"   
## [73] "goalkeeping\_reflexes" "goalkeeping\_speed"

# 3) Afficher les types de données de chaque colonne

Nous allons utiliser la fonction str(fifa22) pour afficher la structure de la base de donnée.

Cette fonction affiche aussi les types de données pour chaque type de colonne. Par exemple chr pour des données texte, num pour des données numériques. Il y a des opérations qui ne sont permises que pour certains types de données.

str(fifa22)

## tibble [19,239 × 74] (S3: tbl\_df/tbl/data.frame)  
## $ short\_name : chr [1:19239] "L, Messi" "R, Lewandowski" "Cristiano Ronaldo" "Neymar Jr" ...  
## $ player\_positions : chr [1:19239] "RW, ST, CF" "ST" "ST, LW" "LW, CAM" ...  
## $ overall : num [1:19239] 93 92 91 91 91 91 91 90 90 90 ...  
## $ potential : num [1:19239] 93 92 91 91 91 93 95 90 92 90 ...  
## $ value\_eur : num [1:19239] 7.80e+07 1.20e+08 4.50e+07 1.29e+08 1.26e+08 ...  
## $ wage\_eur : num [1:19239] 320000 270000 270000 270000 350000 130000 230000 86000 250000 240000 ...  
## $ age : num [1:19239] 34 32 36 29 30 28 22 35 29 27 ...  
## $ dob : POSIXct[1:19239], format: "1987-06-24" "1988-08-21" ...  
## $ height\_cm : num [1:19239] 170 185 187 175 181 188 182 193 187 188 ...  
## $ weight\_kg : num [1:19239] 72 81 83 68 70 87 73 93 85 89 ...  
## $ club\_team\_id : num [1:19239] 73 21 11 73 10 240 73 21 241 18 ...  
## $ club\_name : chr [1:19239] "Paris Saint-Germain" "FC Bayern München" "Manchester United" "Paris Saint-Germain" ...  
## $ league\_name : chr [1:19239] "French Ligue 1" "German 1, Bundesliga" "English Premier League" "French Ligue 1" ...  
## $ league\_level : num [1:19239] 1 1 1 1 1 1 1 1 1 1 ...  
## $ club\_position : chr [1:19239] "RW" "ST" "ST" "LW" ...  
## $ club\_jersey\_number : num [1:19239] 30 9 7 10 17 13 7 1 1 10 ...  
## $ club\_loaned\_from : chr [1:19239] NA NA NA NA ...  
## $ club\_joined : POSIXct[1:19239], format: "2021-08-10" "2014-07-01" ...  
## $ club\_contract\_valid\_until : num [1:19239] 2023 2023 2023 2025 2025 ...  
## $ nationality\_id : num [1:19239] 52 37 38 54 7 44 18 21 21 14 ...  
## $ nationality\_name : chr [1:19239] "Argentina" "Poland" "Portugal" "Brazil" ...  
## $ nation\_team\_id : num [1:19239] 1369 1353 1354 NA 1325 ...  
## $ nation\_position : chr [1:19239] "RW" "RS" "ST" NA ...  
## $ nation\_jersey\_number : num [1:19239] 10 9 7 NA 7 NA 10 1 NA 9 ...  
## $ preferred\_foot : chr [1:19239] "Left" "Right" "Right" "Right" ...  
## $ weak\_foot : num [1:19239] 4 4 4 5 5 3 4 4 4 5 ...  
## $ skill\_moves : num [1:19239] 4 4 5 5 4 1 5 1 1 3 ...  
## $ international\_reputation : num [1:19239] 5 5 5 5 4 5 4 5 4 4 ...  
## $ work\_rate : chr [1:19239] "Medium/Low" "High/Medium" "High/Low" "High/Medium" ...  
## $ body\_type : chr [1:19239] "Unique" "Unique" "Unique" "Unique" ...  
## $ real\_face : chr [1:19239] "Yes" "Yes" "Yes" "Yes" ...  
## $ release\_clause\_eur : num [1:19239] 1.44e+08 1.97e+08 8.33e+07 2.39e+08 2.32e+08 ...  
## $ player\_traits : chr [1:19239] "Finesse Shot, Long Shot Taker (AI), Playmaker (AI), Outside Foot Shot, One Club Player, Chip Shot (AI), Technic"| \_\_truncated\_\_ "Solid Player, Finesse Shot, Outside Foot Shot, Chip Shot (AI)" "Power Free-Kick, Flair, Long Shot Taker (AI), Speed Dribbler (AI), Outside Foot Shot" "Injury Prone, Flair, Speed Dribbler (AI), Playmaker (AI), Outside Foot Shot, Technical Dribbler (AI)" ...  
## $ pace : num [1:19239] 85 78 87 91 76 NA 97 NA NA 70 ...  
## $ shooting : num [1:19239] 92 92 94 83 86 NA 88 NA NA 91 ...  
## $ passing : num [1:19239] 91 79 80 86 93 NA 80 NA NA 83 ...  
## $ dribbling : num [1:19239] 95 86 88 94 88 NA 92 NA NA 83 ...  
## $ defending : num [1:19239] 34 44 34 37 64 NA 36 NA NA 47 ...  
## $ physic : num [1:19239] 65 82 75 63 78 NA 77 NA NA 83 ...  
## $ attacking\_crossing : num [1:19239] 85 71 87 85 94 13 78 15 18 80 ...  
## $ attacking\_finishing : num [1:19239] 95 95 95 83 82 11 93 13 14 94 ...  
## $ attacking\_heading\_accuracy : num [1:19239] 70 90 90 63 55 15 72 25 11 86 ...  
## $ attacking\_short\_passing : num [1:19239] 91 85 80 86 94 43 85 60 61 85 ...  
## $ attacking\_volleys : num [1:19239] 88 89 86 86 82 13 83 11 14 88 ...  
## $ skill\_dribbling : num [1:19239] 96 85 88 95 88 12 93 30 21 83 ...  
## $ skill\_curve : num [1:19239] 93 79 81 88 85 13 80 14 18 83 ...  
## $ skill\_fk\_accuracy : num [1:19239] 94 85 84 87 83 14 69 11 12 65 ...  
## $ skill\_long\_passing : num [1:19239] 91 70 77 81 93 40 71 68 63 86 ...  
## $ skill\_ball\_control : num [1:19239] 96 88 88 95 91 30 91 46 30 85 ...  
## $ movement\_acceleration : num [1:19239] 91 77 85 93 76 43 97 54 38 65 ...  
## $ movement\_sprint\_speed : num [1:19239] 80 79 88 89 76 60 97 60 50 74 ...  
## $ movement\_agility : num [1:19239] 91 77 86 96 79 67 92 51 39 71 ...  
## $ movement\_reactions : num [1:19239] 94 93 94 89 91 88 93 87 86 92 ...  
## $ movement\_balance : num [1:19239] 95 82 74 84 78 49 83 35 43 70 ...  
## $ power\_shot\_power : num [1:19239] 86 90 94 80 91 59 86 68 66 91 ...  
## $ power\_jumping : num [1:19239] 68 85 95 64 63 78 78 77 79 79 ...  
## $ power\_stamina : num [1:19239] 72 76 77 81 89 41 88 43 35 83 ...  
## $ power\_strength : num [1:19239] 69 86 77 53 74 78 77 80 78 85 ...  
## $ power\_long\_shots : num [1:19239] 94 87 93 81 91 12 82 16 10 86 ...  
## $ mentality\_aggression : num [1:19239] 44 81 63 63 76 34 62 29 43 80 ...  
## $ mentality\_interceptions : num [1:19239] 40 49 29 37 66 19 38 30 22 44 ...  
## $ mentality\_positioning : num [1:19239] 93 95 95 86 88 11 92 12 11 94 ...  
## $ mentality\_vision : num [1:19239] 95 81 76 90 94 65 82 70 70 87 ...  
## $ mentality\_penalties : num [1:19239] 75 90 88 93 83 11 79 47 25 91 ...  
## $ mentality\_composure : num [1:19239] 96 88 95 93 89 68 88 70 70 91 ...  
## $ defending\_marking\_awareness: num [1:19239] 20 35 24 35 68 27 26 17 25 50 ...  
## $ defending\_standing\_tackle : num [1:19239] 35 42 32 32 65 12 34 10 13 36 ...  
## $ defending\_sliding\_tackle : num [1:19239] 24 19 24 29 53 18 32 11 10 38 ...  
## $ goalkeeping\_diving : num [1:19239] 6 15 7 9 15 87 13 88 88 8 ...  
## $ goalkeeping\_handling : num [1:19239] 11 6 11 9 13 92 5 88 85 10 ...  
## $ goalkeeping\_kicking : num [1:19239] 15 12 15 15 5 78 7 91 88 11 ...  
## $ goalkeeping\_positioning : num [1:19239] 14 8 14 15 10 90 11 89 88 14 ...  
## $ goalkeeping\_reflexes : num [1:19239] 8 10 11 11 13 90 6 88 90 11 ...  
## $ goalkeeping\_speed : num [1:19239] NA NA NA NA NA 50 NA 56 43 NA ...

# 4) Est-ce que R a correctement reconnu le type de chaque colonne ?

Pour la plupart des données, R semble avoir bien reconnu le type des données. Par contre toutes les données qui comportent du texte ont été lus comme chr alors que certaines données texte peuvent représenter des catégories. Par exemple la variable work\_rate représente une catégorie (plusieurs joueurs peuvent être caractérisés pour un même work\_rate, cette donnée n’est pas unique à chaque joueur).

# 5) Convertir les colonnes dans le bon type si besoin

fifa22$nationality\_name <- as.factor(fifa22$nationality\_name)  
fifa22$work\_rate <- as.factor(fifa22$work\_rate)  
fifa22$club\_name <- as.factor(fifa22$club\_name)

# 6) Afficher le salaire mensuel moyen des joueurs par ligue. Commenter quelques uns

Dans la base de données il y a une variable qui s’appelle wage\_eur qui représente le salaire mensuel des jouers de foot.

aggregate(wage\_eur ~ league\_name, data=fifa22, FUN = mean)

## league\_name wage\_eur  
## 1 Argentina Primera División 6140.9904  
## 2 Australian Hyundai A-League 1638.4477  
## 3 Austrian Football Bundesliga 5206.2121  
## 4 Belgian Jupiler Pro League 7422.7459  
## 5 Campeonato Brasileiro Série A 15263.8889  
## 6 Chilian Campeonato Nacional 2016.4516  
## 7 Chinese Super League 4255.8296  
## 8 Colombian Liga Postobón 951.0471  
## 9 Croatian Prva HNL 591.0714  
## 10 Cypriot First Division 601.7857  
## 11 Czech Republic Gambrinus Liga 739.2405  
## 12 Danish Superliga 4234.2767  
## 13 Ecuadorian Serie A 550.0000  
## 14 English League Championship 11440.9344  
## 15 English League One 2879.3131  
## 16 English League Two 2592.9853  
## 17 English National League 500.0000  
## 18 English Premier League 50847.6994  
## 19 Finnish Veikkausliiga 500.0000  
## 20 French Ligue 1 21462.7383  
## 21 French Ligue 2 2242.2495  
## 22 German 1, Bundesliga 24407.7132  
## 23 German 2, Bundesliga 6136.1765  
## 24 German 3, Bundesliga 1229.4280  
## 25 Greek Super League 804.4643  
## 26 Holland Eredivisie 5058.0285  
## 27 Hungarian Nemzeti Bajnokság I 617.8571  
## 28 Indian Super League 606.6308  
## 29 Italian Serie A 31004.5293  
## 30 Italian Serie B 5298.1061  
## 31 Japanese J, League Division 1 2664.6010  
## 32 Korean K League 1 2326.1905  
## 33 Liga de Fútbol Profesional Boliviano 516.8449  
## 34 Mexican Liga MX 12220.0820  
## 35 Norwegian Eliteserien 1415.7107  
## 36 Paraguayan Primera División 575.5747  
## 37 Peruvian Primera División 511.0429  
## 38 Polish T-Mobile Ekstraklasa 2034.6774  
## 39 Portuguese Liga ZON SAGRES 6335.8416  
## 40 Rep, Ireland Airtricity League 619.7674  
## 41 Romanian Liga I 2616.7421  
## 42 Russian Premier League 24213.4146  
## 43 Saudi Abdul L, Jameel League 8381.0897  
## 44 Scottish Premiership 6782.5000  
## 45 South African Premier Division 552.6786  
## 46 Spain Primera Division 31128.8310  
## 47 Spanish Segunda División 4734.3200  
## 48 Swedish Allsvenskan 1553.5623  
## 49 Swiss Super League 5498.7037  
## 50 Turkish Süper Lig 11915.1013  
## 51 UAE Arabian Gulf League 580.3571  
## 52 Ukrainian Premier League 716.9643  
## 53 Uruguayan Primera División 540.5303  
## 54 USA Major League Soccer 3255.3178  
## 55 Venezuelan Primera División 500.9317

Dans FUN on peut préciser median par exemple.

Il me génère un tableau avec deux colonnes. En moyenne les joueurs de Argentina Primera División gagnent 6140€.

On peut trier les données par ordre décroissant. On utilise la fonction arrange du package dplyr.

salaire\_moyen\_league <- aggregate(wage\_eur ~ league\_name, data=fifa22, FUN = mean)  
library(dplyr)

##   
## Attachement du package : 'dplyr'

## Les objets suivants sont masqués depuis 'package:stats':  
##   
## filter, lag

## Les objets suivants sont masqués depuis 'package:base':  
##   
## intersect, setdiff, setequal, union

arrange(salaire\_moyen\_league, -wage\_eur)

## league\_name wage\_eur  
## 1 English Premier League 50847.6994  
## 2 Spain Primera Division 31128.8310  
## 3 Italian Serie A 31004.5293  
## 4 German 1, Bundesliga 24407.7132  
## 5 Russian Premier League 24213.4146  
## 6 French Ligue 1 21462.7383  
## 7 Campeonato Brasileiro Série A 15263.8889  
## 8 Mexican Liga MX 12220.0820  
## 9 Turkish Süper Lig 11915.1013  
## 10 English League Championship 11440.9344  
## 11 Saudi Abdul L, Jameel League 8381.0897  
## 12 Belgian Jupiler Pro League 7422.7459  
## 13 Scottish Premiership 6782.5000  
## 14 Portuguese Liga ZON SAGRES 6335.8416  
## 15 Argentina Primera División 6140.9904  
## 16 German 2, Bundesliga 6136.1765  
## 17 Swiss Super League 5498.7037  
## 18 Italian Serie B 5298.1061  
## 19 Austrian Football Bundesliga 5206.2121  
## 20 Holland Eredivisie 5058.0285  
## 21 Spanish Segunda División 4734.3200  
## 22 Chinese Super League 4255.8296  
## 23 Danish Superliga 4234.2767  
## 24 USA Major League Soccer 3255.3178  
## 25 English League One 2879.3131  
## 26 Japanese J, League Division 1 2664.6010  
## 27 Romanian Liga I 2616.7421  
## 28 English League Two 2592.9853  
## 29 Korean K League 1 2326.1905  
## 30 French Ligue 2 2242.2495  
## 31 Polish T-Mobile Ekstraklasa 2034.6774  
## 32 Chilian Campeonato Nacional 2016.4516  
## 33 Australian Hyundai A-League 1638.4477  
## 34 Swedish Allsvenskan 1553.5623  
## 35 Norwegian Eliteserien 1415.7107  
## 36 German 3, Bundesliga 1229.4280  
## 37 Colombian Liga Postobón 951.0471  
## 38 Greek Super League 804.4643  
## 39 Czech Republic Gambrinus Liga 739.2405  
## 40 Ukrainian Premier League 716.9643  
## 41 Rep, Ireland Airtricity League 619.7674  
## 42 Hungarian Nemzeti Bajnokság I 617.8571  
## 43 Indian Super League 606.6308  
## 44 Cypriot First Division 601.7857  
## 45 Croatian Prva HNL 591.0714  
## 46 UAE Arabian Gulf League 580.3571  
## 47 Paraguayan Primera División 575.5747  
## 48 South African Premier Division 552.6786  
## 49 Ecuadorian Serie A 550.0000  
## 50 Uruguayan Primera División 540.5303  
## 51 Liga de Fútbol Profesional Boliviano 516.8449  
## 52 Peruvian Primera División 511.0429  
## 53 Venezuelan Primera División 500.9317  
## 54 English National League 500.0000  
## 55 Finnish Veikkausliiga 500.0000

En moyenne les joueurs de l’English Premier League gagnent 50847.6994€ par mois.

# 7) Afficher le potentiel moyen des joueurs par ligue

potentiel\_moyen\_league <- aggregate(potential ~ league\_name, data=fifa22, FUN = mean)  
arrange(potentiel\_moyen\_league, -potential)

## league\_name potential  
## 1 English Premier League 78.99693  
## 2 Spain Primera Division 78.69510  
## 3 Ukrainian Premier League 77.57143  
## 4 Italian Serie A 77.19361  
## 5 German 1, Bundesliga 77.14519  
## 6 French Ligue 1 76.69671  
## 7 Russian Premier League 76.34146  
## 8 Czech Republic Gambrinus Liga 75.27848  
## 9 Greek Super League 75.00893  
## 10 Croatian Prva HNL 74.64286  
## 11 Portuguese Liga ZON SAGRES 74.56634  
## 12 Holland Eredivisie 73.59959  
## 13 Belgian Jupiler Pro League 73.25410  
## 14 Spanish Segunda División 72.79040  
## 15 Argentina Primera División 72.74553  
## 16 English League Championship 72.62343  
## 17 Hungarian Nemzeti Bajnokság I 72.60714  
## 18 Mexican Liga MX 72.20492  
## 19 Italian Serie B 71.95455  
## 20 Campeonato Brasileiro Série A 71.78611  
## 21 Turkish Süper Lig 71.78269  
## 22 Swiss Super League 71.68519  
## 23 German 2, Bundesliga 71.25882  
## 24 Austrian Football Bundesliga 70.70606  
## 25 USA Major League Soccer 70.61479  
## 26 Danish Superliga 69.98742  
## 27 French Ligue 2 69.96219  
## 28 Cypriot First Division 69.82143  
## 29 UAE Arabian Gulf League 69.75000  
## 30 Chilian Campeonato Nacional 69.74194  
## 31 Colombian Liga Postobón 69.70681  
## 32 Scottish Premiership 69.62188  
## 33 South African Premier Division 69.57143  
## 34 Ecuadorian Serie A 69.50877  
## 35 Uruguayan Primera División 69.27273  
## 36 Paraguayan Primera División 69.23563  
## 37 Norwegian Eliteserien 68.99501  
## 38 English League One 68.96326  
## 39 Peruvian Primera División 68.88957  
## 40 Polish T-Mobile Ekstraklasa 68.18952  
## 41 Venezuelan Primera División 68.15528  
## 42 Finnish Veikkausliiga 68.12000  
## 43 Japanese J, League Division 1 68.09847  
## 44 German 3, Bundesliga 67.97048  
## 45 Liga de Fútbol Profesional Boliviano 67.77005  
## 46 Swedish Allsvenskan 67.35369  
## 47 Romanian Liga I 67.32353  
## 48 Australian Hyundai A-League 67.22383  
## 49 Saudi Abdul L, Jameel League 66.80769  
## 50 Korean K League 1 66.52976  
## 51 English League Two 66.48287  
## 52 English National League 64.51852  
## 53 Rep, Ireland Airtricity League 63.31008  
## 54 Chinese Super League 62.94170  
## 55 Indian Super League 61.63082

# 8) Afficher le salaire moyen des joueurs par niveau de réputation

salaire\_moyen\_reputation <- aggregate(wage\_eur ~ international\_reputation, data=fifa22, FUN = mean)  
arrange(salaire\_moyen\_reputation, -wage\_eur)

## international\_reputation wage\_eur  
## 1 5 191500.000  
## 2 4 153368.421  
## 3 3 73034.629  
## 4 2 34924.396  
## 5 1 5948.719

# 9) Le salaire moyen des joueurs par ligue et par réputation internationale

aggregate(wage\_eur ~ league\_name + international\_reputation, data=fifa22, FUN = mean)

## league\_name international\_reputation wage\_eur  
## 1 Argentina Primera División 1 5883.0703  
## 2 Australian Hyundai A-League 1 1637.1377  
## 3 Austrian Football Bundesliga 1 5119.4190  
## 4 Belgian Jupiler Pro League 1 7104.4492  
## 5 Campeonato Brasileiro Série A 1 15263.8889  
## 6 Chilian Campeonato Nacional 1 2016.4516  
## 7 Chinese Super League 1 3761.8056  
## 8 Colombian Liga Postobón 1 951.0471  
## 9 Croatian Prva HNL 1 573.5849  
## 10 Cypriot First Division 1 600.0000  
## 11 Czech Republic Gambrinus Liga 1 732.4675  
## 12 Danish Superliga 1 4235.0158  
## 13 Ecuadorian Serie A 1 550.0000  
## 14 English League Championship 1 11311.3861  
## 15 English League One 1 2870.9135  
## 16 English League Two 1 2593.9542  
## 17 English National League 1 500.0000  
## 18 English Premier League 1 27107.5294  
## 19 Finnish Veikkausliiga 1 500.0000  
## 20 French Ligue 1 1 11806.2954  
## 21 French Ligue 2 1 2195.6480  
## 22 German 1, Bundesliga 1 12885.1389  
## 23 German 2, Bundesliga 1 5948.6948  
## 24 German 3, Bundesliga 1 1225.1391  
## 25 Greek Super League 1 758.5859  
## 26 Holland Eredivisie 1 4386.9612  
## 27 Hungarian Nemzeti Bajnokság I 1 612.0000  
## 28 Indian Super League 1 606.6308  
## 29 Italian Serie A 1 17530.3161  
## 30 Italian Serie B 1 4523.5537  
## 31 Japanese J, League Division 1 1 2573.1643  
## 32 Korean K League 1 1 2248.7654  
## 33 Liga de Fútbol Profesional Boliviano 1 516.8449  
## 34 Mexican Liga MX 1 12239.5062  
## 35 Norwegian Eliteserien 1 1415.7107  
## 36 Paraguayan Primera División 1 575.5747  
## 37 Peruvian Primera División 1 511.0429  
## 38 Polish T-Mobile Ekstraklasa 1 2020.6897  
## 39 Portuguese Liga ZON SAGRES 1 5616.3180  
## 40 Rep, Ireland Airtricity League 1 619.7674  
## 41 Romanian Liga I 1 2616.7421  
## 42 Russian Premier League 1 21711.2676  
## 43 Saudi Abdul L, Jameel League 1 6768.9773  
## 44 Scottish Premiership 1 6209.6154  
## 45 South African Premier Division 1 553.6364  
## 46 Spain Primera Division 1 18324.0042  
## 47 Spanish Segunda División 1 4627.5947  
## 48 Swedish Allsvenskan 1 1553.5623  
## 49 Swiss Super League 1 5145.9302  
## 50 Turkish Süper Lig 1 9923.1915  
## 51 UAE Arabian Gulf League 1 580.3571  
## 52 Ukrainian Premier League 1 681.0000  
## 53 Uruguayan Primera División 1 540.5303  
## 54 USA Major League Soccer 1 2965.3103  
## 55 Venezuelan Primera División 1 500.9317  
## 56 Argentina Primera División 2 12153.8462  
## 57 Australian Hyundai A-League 2 2000.0000  
## 58 Austrian Football Bundesliga 2 14666.6667  
## 59 Belgian Jupiler Pro League 2 14384.6154  
## 60 Chinese Super League 2 17727.2727  
## 61 Croatian Prva HNL 2 900.0000  
## 62 Cypriot First Division 2 650.0000  
## 63 Czech Republic Gambrinus Liga 2 1000.0000  
## 64 Danish Superliga 2 4000.0000  
## 65 English League Championship 2 20666.6667  
## 66 English League One 2 5500.0000  
## 67 English League Two 2 2000.0000  
## 68 English Premier League 2 68751.6779  
## 69 French Ligue 1 2 34359.3750  
## 70 French Ligue 2 2 4181.8182  
## 71 German 1, Bundesliga 2 36575.1634  
## 72 German 2, Bundesliga 2 13916.6667  
## 73 German 3, Bundesliga 2 2000.0000  
## 74 Greek Super League 2 1200.0000  
## 75 Holland Eredivisie 2 13772.7273  
## 76 Hungarian Nemzeti Bajnokság I 2 666.6667  
## 77 Italian Serie A 2 42734.6939  
## 78 Italian Serie B 2 14750.0000  
## 79 Japanese J, League Division 1 2 5700.0000  
## 80 Korean K League 1 2 4416.6667  
## 81 Mexican Liga MX 2 7500.0000  
## 82 Polish T-Mobile Ekstraklasa 2 3500.0000  
## 83 Portuguese Liga ZON SAGRES 2 20176.4706  
## 84 Russian Premier League 2 38750.0000  
## 85 Saudi Abdul L, Jameel League 2 31960.0000  
## 86 Scottish Premiership 2 29125.0000  
## 87 South African Premier Division 2 500.0000  
## 88 Spain Primera Division 2 43931.3725  
## 89 Spanish Segunda División 2 8333.3333  
## 90 Swiss Super League 2 13090.9091  
## 91 Turkish Süper Lig 2 20596.4912  
## 92 Ukrainian Premier League 2 1016.6667  
## 93 USA Major League Soccer 2 6818.1818  
## 94 Argentina Primera División 3 12000.0000  
## 95 Belgian Jupiler Pro League 3 27333.3333  
## 96 Chinese Super League 3 26000.0000  
## 97 English League Championship 3 20000.0000  
## 98 English Premier League 3 127306.4516  
## 99 French Ligue 1 3 65172.4138  
## 100 French Ligue 2 3 5000.0000  
## 101 German 1, Bundesliga 3 67428.5714  
## 102 Greek Super League 3 1000.0000  
## 103 Holland Eredivisie 3 24200.0000  
## 104 Italian Serie A 3 72672.1311  
## 105 Italian Serie B 3 8000.0000  
## 106 Japanese J, League Division 1 3 3900.0000  
## 107 Polish T-Mobile Ekstraklasa 3 6000.0000  
## 108 Portuguese Liga ZON SAGRES 3 17555.5556  
## 109 Russian Premier League 3 44666.6667  
## 110 Saudi Abdul L, Jameel League 3 48333.3333  
## 111 Spain Primera Division 3 82810.8108  
## 112 Swiss Super League 3 13000.0000  
## 113 Turkish Süper Lig 3 39785.7143  
## 114 USA Major League Soccer 3 9727.2727  
## 115 English Premier League 4 215000.0000  
## 116 French Ligue 1 4 126000.0000  
## 117 German 1, Bundesliga 4 121250.0000  
## 118 Holland Eredivisie 4 29000.0000  
## 119 Italian Serie A 4 98166.6667  
## 120 Italian Serie B 4 18000.0000  
## 121 Japanese J, League Division 1 4 10000.0000  
## 122 Portuguese Liga ZON SAGRES 4 14000.0000  
## 123 Spain Primera Division 4 210266.6667  
## 124 Turkish Süper Lig 4 37500.0000  
## 125 USA Major League Soccer 4 14000.0000  
## 126 English Premier League 5 270000.0000  
## 127 French Ligue 1 5 295000.0000  
## 128 German 1, Bundesliga 5 178000.0000  
## 129 Italian Serie A 5 51000.0000  
## 130 Spain Primera Division 5 132500.0000

# 10) Afficher les statistiques générales de toutes les données numériques

Dans un premier temps on va filtrer le jeu de données pour ne retenir que les colonnes numériques. La fonction select du package dplyr nous permet de sélectionner les colonnes d’un tableau de données (data.frame, tibble)

num\_cols <- select(fifa22, where(is.numeric))  
summary(num\_cols)

## overall potential value\_eur wage\_eur   
## Min. :47.00 Min. :49.00 Min. :9.00e+03 Min. : 500   
## 1st Qu.:61.00 1st Qu.:67.00 1st Qu.:4.75e+05 1st Qu.: 1000   
## Median :66.00 Median :71.00 Median :9.75e+05 Median : 3000   
## Mean :65.77 Mean :71.08 Mean :2.85e+06 Mean : 9018   
## 3rd Qu.:70.00 3rd Qu.:75.00 3rd Qu.:2.00e+06 3rd Qu.: 8000   
## Max. :93.00 Max. :95.00 Max. :1.94e+08 Max. :350000   
## NA's :74 NA's :61   
## age height\_cm weight\_kg club\_team\_id   
## Min. :16.00 Min. :155.0 Min. : 49.00 Min. : 1   
## 1st Qu.:21.00 1st Qu.:176.0 1st Qu.: 70.00 1st Qu.: 479   
## Median :25.00 Median :181.0 Median : 75.00 Median : 1938   
## Mean :25.21 Mean :181.3 Mean : 74.94 Mean : 50581   
## 3rd Qu.:29.00 3rd Qu.:186.0 3rd Qu.: 80.00 3rd Qu.:111139   
## Max. :54.00 Max. :206.0 Max. :110.00 Max. :115820   
## NA's :61   
## league\_level club\_jersey\_number club\_contract\_valid\_until nationality\_id   
## Min. :1.000 Min. : 1.00 Min. :2021 Min. : 1.0   
## 1st Qu.:1.000 1st Qu.: 9.00 1st Qu.:2022 1st Qu.: 21.0   
## Median :1.000 Median :18.00 Median :2022 Median : 45.0   
## Mean :1.354 Mean :20.95 Mean :2023 Mean : 58.6   
## 3rd Qu.:1.000 3rd Qu.:27.00 3rd Qu.:2024 3rd Qu.: 60.0   
## Max. :5.000 Max. :99.00 Max. :2031 Max. :219.0   
## NA's :61 NA's :61 NA's :61   
## nation\_team\_id nation\_jersey\_number weak\_foot skill\_moves   
## Min. : 1318 Min. : 1.00 Min. :1.000 Min. :1.000   
## 1st Qu.: 1338 1st Qu.: 7.00 1st Qu.:3.000 1st Qu.:2.000   
## Median : 1357 Median :12.00 Median :3.000 Median :2.000   
## Mean : 14481 Mean :12.57 Mean :2.946 Mean :2.352   
## 3rd Qu.: 1386 3rd Qu.:19.00 3rd Qu.:3.000 3rd Qu.:3.000   
## Max. :111473 Max. :28.00 Max. :5.000 Max. :5.000   
## NA's :18480 NA's :18480   
## international\_reputation release\_clause\_eur pace shooting   
## Min. :1.000 Min. : 16000 Min. :28.00 Min. :18.00   
## 1st Qu.:1.000 1st Qu.: 806000 1st Qu.:62.00 1st Qu.:42.00   
## Median :1.000 Median : 1600000 Median :69.00 Median :54.00   
## Mean :1.094 Mean : 5374044 Mean :68.21 Mean :52.35   
## 3rd Qu.:1.000 3rd Qu.: 3700000 3rd Qu.:76.00 3rd Qu.:63.00   
## Max. :5.000 Max. :373500000 Max. :97.00 Max. :94.00   
## NA's :1176 NA's :2132 NA's :2132   
## passing dribbling defending physic   
## Min. :25.00 Min. :27.00 Min. :14.0 Min. :29.00   
## 1st Qu.:51.00 1st Qu.:57.00 1st Qu.:37.0 1st Qu.:59.00   
## Median :58.00 Median :64.00 Median :56.0 Median :66.00   
## Mean :57.31 Mean :62.56 Mean :51.7 Mean :64.82   
## 3rd Qu.:64.00 3rd Qu.:69.00 3rd Qu.:64.0 3rd Qu.:72.00   
## Max. :93.00 Max. :95.00 Max. :91.0 Max. :90.00   
## NA's :2132 NA's :2132 NA's :2132 NA's :2132   
## attacking\_crossing attacking\_finishing attacking\_heading\_accuracy  
## Min. : 6.00 Min. : 2.00 Min. : 5.00   
## 1st Qu.:38.00 1st Qu.:30.00 1st Qu.:44.00   
## Median :54.00 Median :50.00 Median :55.00   
## Mean :49.58 Mean :45.89 Mean :51.78   
## 3rd Qu.:63.00 3rd Qu.:62.00 3rd Qu.:64.00   
## Max. :94.00 Max. :95.00 Max. :93.00   
##   
## attacking\_short\_passing attacking\_volleys skill\_dribbling skill\_curve   
## Min. : 7.00 Min. : 3.00 Min. : 4.00 Min. : 6.00   
## 1st Qu.:54.00 1st Qu.:30.00 1st Qu.:50.00 1st Qu.:35.00   
## Median :62.00 Median :43.00 Median :61.00 Median :49.00   
## Mean :58.87 Mean :42.46 Mean :55.66 Mean :47.27   
## 3rd Qu.:68.00 3rd Qu.:56.00 3rd Qu.:68.00 3rd Qu.:61.00   
## Max. :94.00 Max. :90.00 Max. :96.00 Max. :94.00   
##   
## skill\_fk\_accuracy skill\_long\_passing skill\_ball\_control movement\_acceleration  
## Min. : 4.00 Min. : 9.00 Min. : 8.00 Min. :14.00   
## 1st Qu.:31.00 1st Qu.:44.00 1st Qu.:55.00 1st Qu.:57.00   
## Median :41.00 Median :56.00 Median :63.00 Median :67.00   
## Mean :42.25 Mean :53.07 Mean :58.47 Mean :64.65   
## 3rd Qu.:55.00 3rd Qu.:64.00 3rd Qu.:69.00 3rd Qu.:75.00   
## Max. :94.00 Max. :93.00 Max. :96.00 Max. :97.00   
##   
## movement\_sprint\_speed movement\_agility movement\_reactions movement\_balance  
## Min. :15.00 Min. :18.0 Min. :25.00 Min. :15.00   
## 1st Qu.:58.00 1st Qu.:55.0 1st Qu.:56.00 1st Qu.:56.00   
## Median :68.00 Median :66.0 Median :62.00 Median :66.00   
## Mean :64.71 Mean :63.5 Mean :61.45 Mean :64.07   
## 3rd Qu.:75.00 3rd Qu.:74.0 3rd Qu.:67.00 3rd Qu.:74.00   
## Max. :97.00 Max. :96.0 Max. :94.00 Max. :96.00   
##   
## power\_shot\_power power\_jumping power\_stamina power\_strength   
## Min. :20.00 Min. :22.00 Min. :12.00 Min. :19.00   
## 1st Qu.:48.00 1st Qu.:57.00 1st Qu.:56.00 1st Qu.:57.00   
## Median :59.00 Median :65.00 Median :66.00 Median :66.00   
## Mean :57.78 Mean :64.81 Mean :63.08 Mean :65.01   
## 3rd Qu.:68.00 3rd Qu.:73.00 3rd Qu.:74.00 3rd Qu.:74.00   
## Max. :95.00 Max. :95.00 Max. :97.00 Max. :97.00   
##   
## power\_long\_shots mentality\_aggression mentality\_interceptions  
## Min. : 4.00 Min. :10.00 Min. : 3.00   
## 1st Qu.:32.00 1st Qu.:44.00 1st Qu.:26.00   
## Median :51.00 Median :58.00 Median :53.00   
## Mean :46.64 Mean :55.54 Mean :46.61   
## 3rd Qu.:62.00 3rd Qu.:68.00 3rd Qu.:64.00   
## Max. :94.00 Max. :95.00 Max. :91.00   
##   
## mentality\_positioning mentality\_vision mentality\_penalties mentality\_composure  
## Min. : 2.00 Min. :10.00 Min. : 7.00 Min. :12.00   
## 1st Qu.:40.00 1st Qu.:45.00 1st Qu.:38.00 1st Qu.:50.00   
## Median :56.00 Median :55.00 Median :49.00 Median :59.00   
## Mean :50.33 Mean :53.96 Mean :47.86 Mean :57.93   
## 3rd Qu.:64.00 3rd Qu.:64.00 3rd Qu.:60.00 3rd Qu.:66.00   
## Max. :96.00 Max. :95.00 Max. :93.00 Max. :96.00   
##   
## defending\_marking\_awareness defending\_standing\_tackle defending\_sliding\_tackle  
## Min. : 4.0 Min. : 5.00 Min. : 5.00   
## 1st Qu.:29.0 1st Qu.:28.00 1st Qu.:25.00   
## Median :52.0 Median :56.00 Median :53.00   
## Mean :46.6 Mean :48.05 Mean :45.91   
## 3rd Qu.:63.0 3rd Qu.:65.00 3rd Qu.:63.00   
## Max. :93.0 Max. :93.00 Max. :92.00   
##   
## goalkeeping\_diving goalkeeping\_handling goalkeeping\_kicking  
## Min. : 2.00 Min. : 2.00 Min. : 2.00   
## 1st Qu.: 8.00 1st Qu.: 8.00 1st Qu.: 8.00   
## Median :11.00 Median :11.00 Median :11.00   
## Mean :16.41 Mean :16.19 Mean :16.06   
## 3rd Qu.:14.00 3rd Qu.:14.00 3rd Qu.:14.00   
## Max. :91.00 Max. :92.00 Max. :93.00   
##   
## goalkeeping\_positioning goalkeeping\_reflexes goalkeeping\_speed  
## Min. : 2.00 Min. : 2.00 Min. :15.00   
## 1st Qu.: 8.00 1st Qu.: 8.00 1st Qu.:27.00   
## Median :11.00 Median :11.00 Median :36.00   
## Mean :16.23 Mean :16.49 Mean :36.44   
## 3rd Qu.:14.00 3rd Qu.:14.00 3rd Qu.:45.00   
## Max. :92.00 Max. :90.00 Max. :65.00   
## NA's :17107

La fonction summary() est utilisée pour résumer un jeu de donnée. Si les données sont numériques elle affiche le minimum (min), le premier quartile, la médiane, la moyenne, le 3e quartiel et le maximum de la variable.

fac\_cols <- select(fifa22, where(is.factor))  
summary(fac\_cols)

## club\_name nationality\_name work\_rate   
## Arsenal : 33 England : 1719 Medium/Medium:10015   
## Borussia Mönchengladbach: 33 Germany : 1214 High/Medium : 3661   
## Brentford : 33 Spain : 1086 Medium/High : 1880   
## Brighton & Hove Albion : 33 France : 980 High/High : 1099   
## Burnley : 33 Argentina: 960 Medium/Low : 810   
## (Other) :19013 Brazil : 897 High/Low : 808   
## NA's : 61 (Other) :12383 (Other) : 966

Si la colonne contient des données manquantes, la fonction affiche le nombre de données manquantes via la ligne NA’s.

# 11) Afficher la matrice de corrélation entre les variables suivantes

cor(fifa22$weight\_kg, fifa22$height\_cm)

## [1] 0.765465

La corrélation entre le poids d’un joueur de football et sa taille est de 0.76. Cette corrélation est positive, clea signifie que le poids et la taille vont dans le même sens. Cette même fonction cor() peut nous générer une matrice de corrélation entre plusieurs variables numériques.

cor(fifa22[, c("height\_cm", "weight\_kg", "wage\_eur", "potential", "skill\_dribbling")])

## height\_cm weight\_kg wage\_eur potential skill\_dribbling  
## height\_cm 1.000000000 0.76546497 NA 0.004403203 -0.4772269  
## weight\_kg 0.765464972 1.00000000 NA -0.016912415 -0.3973634  
## wage\_eur NA NA 1 NA NA  
## potential 0.004403203 -0.01691241 NA 1.000000000 0.3113960  
## skill\_dribbling -0.477226949 -0.39736339 NA 0.311395974 1.0000000

S’il y a des données manquantes, R ignore le calcul des corrélations. Dans ce cas il faut expliciter dire à la fonction cor quoi faire. On utilise l’argument use = “complete.obs” pour indiquer que R ne prenne les lignes qui n’ont pas de données manquantes.

cor(fifa22[, c("height\_cm", "weight\_kg", "wage\_eur", "potential", "skill\_dribbling")],   
 use = "complete.obs")

## height\_cm weight\_kg wage\_eur potential skill\_dribbling  
## height\_cm 1.00000000 0.76529382 0.02806235 0.00409139 -0.4769262  
## weight\_kg 0.76529382 1.00000000 0.06620795 -0.01730122 -0.3969755  
## wage\_eur 0.02806235 0.06620795 1.00000000 0.49761083 0.2552045  
## potential 0.00409139 -0.01730122 0.49761083 1.00000000 0.3119857  
## skill\_dribbling -0.47692620 -0.39697551 0.25520447 0.31198571 1.0000000

cor(fifa22[, c("movement\_acceleration", "movement\_agility", "power\_shot\_power",  
"power\_jumping", "power\_stamina", "power\_strength", "mentality\_aggression",  
"mentality\_interceptions", "defending\_marking\_awareness",  
"defending\_standing\_tackle", "defending\_sliding\_tackle", "goalkeeping\_diving",  
"goalkeeping\_handling", "goalkeeping\_kicking", "goalkeeping\_speed")],   
 use = "complete.obs")

## movement\_acceleration movement\_agility  
## movement\_acceleration 1.0000000 0.5944028  
## movement\_agility 0.5944028 1.0000000  
## power\_shot\_power 0.4127824 0.3486126  
## power\_jumping 0.4911378 0.4932555  
## power\_stamina 0.5385869 0.3911491  
## power\_strength 0.2751928 0.1922516  
## mentality\_aggression 0.2530558 0.2728768  
## mentality\_interceptions 0.5117559 0.3436096  
## defending\_marking\_awareness 0.3753938 0.3040647  
## defending\_standing\_tackle 0.2788265 0.1966700  
## defending\_sliding\_tackle 0.2961128 0.1800403  
## goalkeeping\_diving 0.4569491 0.3875869  
## goalkeeping\_handling 0.4057561 0.3449959  
## goalkeeping\_kicking 0.4129640 0.3504858  
## goalkeeping\_speed 0.9737030 0.6112578  
## power\_shot\_power power\_jumping power\_stamina  
## movement\_acceleration 0.4127824 0.4911378 0.5385869  
## movement\_agility 0.3486126 0.4932555 0.3911491  
## power\_shot\_power 1.0000000 0.3149410 0.3615151  
## power\_jumping 0.3149410 1.0000000 0.3714578  
## power\_stamina 0.3615151 0.3714578 1.0000000  
## power\_strength 0.3276616 0.1433713 0.2961986  
## mentality\_aggression 0.2446895 0.2344888 0.2754022  
## mentality\_interceptions 0.3984902 0.3361833 0.4763375  
## defending\_marking\_awareness 0.3876797 0.2919854 0.3672069  
## defending\_standing\_tackle 0.2186059 0.1616537 0.2584174  
## defending\_sliding\_tackle 0.2049565 0.1676720 0.2573681  
## goalkeeping\_diving 0.7600065 0.3730064 0.3863614  
## goalkeeping\_handling 0.7814859 0.3398169 0.3626059  
## goalkeeping\_kicking 0.9987449 0.3167773 0.3607233  
## goalkeeping\_speed 0.4285260 0.5039864 0.5580247  
## power\_strength mentality\_aggression  
## movement\_acceleration 0.2751928 0.2530558  
## movement\_agility 0.1922516 0.2728768  
## power\_shot\_power 0.3276616 0.2446895  
## power\_jumping 0.1433713 0.2344888  
## power\_stamina 0.2961986 0.2754022  
## power\_strength 1.0000000 0.2599792  
## mentality\_aggression 0.2599792 1.0000000  
## mentality\_interceptions 0.3008788 0.2994144  
## defending\_marking\_awareness 0.1924629 0.2083216  
## defending\_standing\_tackle 0.1212617 0.1425725  
## defending\_sliding\_tackle 0.1493052 0.1256170  
## goalkeeping\_diving 0.3642902 0.2358942  
## goalkeeping\_handling 0.3466732 0.2514927  
## goalkeeping\_kicking 0.3290788 0.2441848  
## goalkeeping\_speed 0.2951726 0.2565095  
## mentality\_interceptions defending\_marking\_awareness  
## movement\_acceleration 0.5117559 0.3753938  
## movement\_agility 0.3436096 0.3040647  
## power\_shot\_power 0.3984902 0.3876797  
## power\_jumping 0.3361833 0.2919854  
## power\_stamina 0.4763375 0.3672069  
## power\_strength 0.3008788 0.1924629  
## mentality\_aggression 0.2994144 0.2083216  
## mentality\_interceptions 1.0000000 0.4915747  
## defending\_marking\_awareness 0.4915747 1.0000000  
## defending\_standing\_tackle 0.3562060 0.2866507  
## defending\_sliding\_tackle 0.3416700 0.2940526  
## goalkeeping\_diving 0.4116444 0.3940371  
## goalkeeping\_handling 0.4112128 0.3932453  
## goalkeeping\_kicking 0.3986962 0.3882431  
## goalkeeping\_speed 0.5331306 0.3994995  
## defending\_standing\_tackle defending\_sliding\_tackle  
## movement\_acceleration 0.2788265 0.2961128  
## movement\_agility 0.1966700 0.1800403  
## power\_shot\_power 0.2186059 0.2049565  
## power\_jumping 0.1616537 0.1676720  
## power\_stamina 0.2584174 0.2573681  
## power\_strength 0.1212617 0.1493052  
## mentality\_aggression 0.1425725 0.1256170  
## mentality\_interceptions 0.3562060 0.3416700  
## defending\_marking\_awareness 0.2866507 0.2940526  
## defending\_standing\_tackle 1.0000000 0.5456108  
## defending\_sliding\_tackle 0.5456108 1.0000000  
## goalkeeping\_diving 0.2160768 0.1859613  
## goalkeeping\_handling 0.2052991 0.1857296  
## goalkeeping\_kicking 0.2194609 0.2054712  
## goalkeeping\_speed 0.2942353 0.3074139  
## goalkeeping\_diving goalkeeping\_handling  
## movement\_acceleration 0.4569491 0.4057561  
## movement\_agility 0.3875869 0.3449959  
## power\_shot\_power 0.7600065 0.7814859  
## power\_jumping 0.3730064 0.3398169  
## power\_stamina 0.3863614 0.3626059  
## power\_strength 0.3642902 0.3466732  
## mentality\_aggression 0.2358942 0.2514927  
## mentality\_interceptions 0.4116444 0.4112128  
## defending\_marking\_awareness 0.3940371 0.3932453  
## defending\_standing\_tackle 0.2160768 0.2052991  
## defending\_sliding\_tackle 0.1859613 0.1857296  
## goalkeeping\_diving 1.0000000 0.8671908  
## goalkeeping\_handling 0.8671908 1.0000000  
## goalkeeping\_kicking 0.7606520 0.7824772  
## goalkeeping\_speed 0.4768946 0.4238856  
## goalkeeping\_kicking goalkeeping\_speed  
## movement\_acceleration 0.4129640 0.9737030  
## movement\_agility 0.3504858 0.6112578  
## power\_shot\_power 0.9987449 0.4285260  
## power\_jumping 0.3167773 0.5039864  
## power\_stamina 0.3607233 0.5580247  
## power\_strength 0.3290788 0.2951726  
## mentality\_aggression 0.2441848 0.2565095  
## mentality\_interceptions 0.3986962 0.5331306  
## defending\_marking\_awareness 0.3882431 0.3994995  
## defending\_standing\_tackle 0.2194609 0.2942353  
## defending\_sliding\_tackle 0.2054712 0.3074139  
## goalkeeping\_diving 0.7606520 0.4768946  
## goalkeeping\_handling 0.7824772 0.4238856  
## goalkeeping\_kicking 1.0000000 0.4288311  
## goalkeeping\_speed 0.4288311 1.0000000

# PARTIE 3 : Réaliser une ACP sur le jeu de données

Réaliser une ACP sur les 100 premiers joueurs en prenant comme variables quantitatives actives les caractéristiques des joueurs suivant :

* “height\_cm”, “weight\_kg”, “skill\_dribbling”, “skill\_ball\_control”, “movement\_acceleration”, “movement\_agility”, “power\_shot\_power”, “power\_jumping”, “power\_stamina”, “power\_strength”, “mentality\_aggression”, “mentality\_interceptions”, “defending\_marking\_awareness”, “defending\_standing\_tackle”, “defending\_sliding\_tackle”, “goalkeeping\_diving”, “goalkeeping\_handling”, “goalkeeping\_kicking”, “goalkeeping\_speed”

## A quoi servent les variables quantitatives actives

Les variables quantitatives actives sont des variables qui entrent dans le calcul des composantes principales de l’ACP. Elles servent à calculer la matrice variance-covariance. Ce sont ces variables qui influent sur les composantes.

var\_quanti\_actives <- c("height\_cm", "weight\_kg", "skill\_dribbling", "skill\_ball\_control",  
"movement\_acceleration", "movement\_agility", "power\_shot\_power",  
"power\_jumping", "power\_stamina", "power\_strength", "mentality\_aggression",  
"mentality\_interceptions", "defending\_marking\_awareness",  
"defending\_standing\_tackle", "defending\_sliding\_tackle", "goalkeeping\_diving",  
"goalkeeping\_handling", "goalkeeping\_kicking", "goalkeeping\_speed")

comme variables quantitatives illustratives - “overall”, “potential”, “value\_eur”, “pace”, “shooting”, “passing”, “dribbling”, “defending”,“physic”

var\_quanti\_illustratives <- c("overall", "potential", "value\_eur", "pace", "shooting", "passing", "dribbling",  
"defending","physic")

## A quoi servent les variables quantitatives illustratives.

Elles nous servent essentiellement dans l’interprétation des résultats de l’ACP. Elles n’entrent pas dans le calcul des composantes. Par exemple la variable overall est un combiné de toutes les caractéristiques des joueurs, si on l’incluait dans les variables actives il y a un risque de redondance dans les informations.

et comme variable qualitative illustrative - “body\_type”.

var\_quali\_illustratives <- c("body\_type")

# Filtrer le jeu de données

On prend les 100 premières observations pour les variables d’intérêts de l’ACP

variables\_acp <- c(var\_quanti\_actives, var\_quanti\_illustratives, var\_quali\_illustratives)

J’utilise la fonction head pour ne prendre que les 100 premières observations.

fifa\_100 <- head(fifa22, 100)  
fifa\_100 <- fifa\_100[, variables\_acp]

# ACP avec FactoMineR

C’est dans le package FactoMineR que sont définies la plupart des fonctions qu’on va utiliser pour faire les analyses factorielles.

library(FactoMineR)

Pour réaliser une ACP avec FactoMineR on, utilise la fonction PCA.

* X : le jeu de données
* scale.unit : TRUE ou FALSE (pour réduire et centrer les données numériques)
* ncp : Le nombre de composantes principales
* quanti.sup : Les indices (position) des variables quantitatives illustratives
* quali.sup : Les indices des variables qualitatives illustratives
* ind.sup : Les indices des lignes contenant les individus illustratifs.
* graph : TRUE ou FALSE pour indiquer si R doit afficher les graphiques de l’ACP

Nous ne sommes pas obligés de renseigner tous les paramètres de la fonction PCA.

La condition minimale à satisfaire pour réaliser une ACP avec la fonction PCA c’est de fournir en argument au paramètre X un jeu de données ne contenant que des variables numériques et ne contenant aucune donnée manquante.

Dans ce cas une ACP sera réalisée en considérant toutes les colonnes comme variables actives.

Dans la pratique nous voulons avoir un contrôle sur les résultats de l’ACP donc nous allons donner plus de précisions à la fonction PCA.

Dans notre présent cas nous allons préciser les indices des variables actives, variables quantitatives illustratives et des variables qualitatives illustratives.

La fonction match retourne la position des variables dans une liste.

Les paramètres de la fonction PCA - quanti.sup : Les indices (position) des variables quantitatives illustratives - quali.sup : Les indices des variables qualitatives illustratives - ind.sup : Les indices des lignes contenant les individus illustratifs.

idx\_var\_quanti\_actives <- match(var\_quanti\_actives, colnames(fifa\_100))  
idx\_var\_quanti\_illustratives <- match(var\_quanti\_illustratives, colnames(fifa\_100))  
idx\_var\_quali\_illustratives <- match(var\_quali\_illustratives, colnames(fifa\_100))

premiere\_acp <- PCA(X = fifa\_100, quanti.sup = idx\_var\_quanti\_illustratives, quali.sup = idx\_var\_quali\_illustratives, graph = FALSE)

## Warning in PCA(X = fifa\_100, quanti.sup = idx\_var\_quanti\_illustratives, :  
## Missing values are imputed by the mean of the variable: you should use the  
## imputePCA function of the missMDA package

summary(premiere\_acp)

##   
## Call:  
## PCA(X = fifa\_100, quanti.sup = idx\_var\_quanti\_illustratives,   
## quali.sup = idx\_var\_quali\_illustratives, graph = FALSE)   
##   
##   
## Eigenvalues  
## Dim.1 Dim.2 Dim.3 Dim.4 Dim.5 Dim.6 Dim.7  
## Variance 9.547 4.167 1.873 1.047 0.767 0.369 0.285  
## % of var. 50.248 21.933 9.860 5.509 4.037 1.941 1.501  
## Cumulative % of var. 50.248 72.181 82.041 87.549 91.586 93.527 95.029  
## Dim.8 Dim.9 Dim.10 Dim.11 Dim.12 Dim.13 Dim.14  
## Variance 0.217 0.167 0.156 0.130 0.080 0.055 0.046  
## % of var. 1.141 0.882 0.819 0.685 0.423 0.288 0.240  
## Cumulative % of var. 96.170 97.051 97.871 98.555 98.978 99.266 99.506  
## Dim.15 Dim.16 Dim.17 Dim.18 Dim.19  
## Variance 0.028 0.022 0.019 0.016 0.009  
## % of var. 0.149 0.115 0.100 0.085 0.045  
## Cumulative % of var. 99.655 99.770 99.870 99.955 100.000  
##   
## Individuals (the 10 first)  
## Dist Dim.1 ctr cos2 Dim.2 ctr  
## 1 | 3.928 | 0.864 0.078 0.048 | -3.479 2.905  
## 2 | 2.857 | 0.591 0.037 0.043 | -0.374 0.033  
## 3 | 3.758 | 0.342 0.012 0.008 | -1.332 0.426  
## 4 | 3.933 | 1.321 0.183 0.113 | -3.624 3.151  
## 5 | 2.615 | 2.024 0.429 0.599 | -0.590 0.084  
## 6 | 7.200 | -6.992 5.121 0.943 | 0.261 0.016  
## 7 | 3.419 | 1.301 0.177 0.145 | -2.278 1.245  
## 8 | 7.505 | -6.894 4.978 0.844 | 0.310 0.023  
## 9 | 8.221 | -7.291 5.568 0.786 | 0.954 0.218  
## 10 | 2.843 | 0.364 0.014 0.016 | 0.467 0.052  
## cos2 Dim.3 ctr cos2   
## 1 0.784 | 0.758 0.306 0.037 |  
## 2 0.017 | 2.560 3.497 0.803 |  
## 3 0.126 | 3.179 5.395 0.716 |  
## 4 0.849 | -0.177 0.017 0.002 |  
## 5 0.051 | 0.008 0.000 0.000 |  
## 6 0.001 | -0.362 0.070 0.003 |  
## 7 0.444 | 1.812 1.753 0.281 |  
## 8 0.002 | 0.693 0.257 0.009 |  
## 9 0.013 | -0.068 0.002 0.000 |  
## 10 0.027 | 2.501 3.339 0.774 |  
##   
## Variables (the 10 first)  
## Dim.1 ctr cos2 Dim.2 ctr cos2   
## height\_cm | -0.532 2.970 0.284 | 0.623 9.302 0.388 |  
## weight\_kg | -0.579 3.517 0.336 | 0.544 7.111 0.296 |  
## skill\_dribbling | 0.910 8.677 0.828 | -0.304 2.215 0.092 |  
## skill\_ball\_control | 0.930 9.052 0.864 | -0.228 1.251 0.052 |  
## movement\_acceleration | 0.649 4.416 0.422 | -0.532 6.793 0.283 |  
## movement\_agility | 0.618 4.006 0.382 | -0.692 11.478 0.478 |  
## power\_shot\_power | 0.570 3.404 0.325 | -0.283 1.921 0.080 |  
## power\_jumping | 0.133 0.185 0.018 | 0.484 5.630 0.235 |  
## power\_stamina | 0.931 9.080 0.867 | -0.021 0.011 0.000 |  
## power\_strength | -0.003 0.000 0.000 | 0.717 12.345 0.514 |  
## Dim.3 ctr cos2   
## height\_cm 0.374 7.474 0.140 |  
## weight\_kg 0.490 12.823 0.240 |  
## skill\_dribbling 0.181 1.756 0.033 |  
## skill\_ball\_control 0.151 1.209 0.023 |  
## movement\_acceleration 0.180 1.726 0.032 |  
## movement\_agility 0.013 0.010 0.000 |  
## power\_shot\_power 0.597 18.995 0.356 |  
## power\_jumping 0.421 9.482 0.178 |  
## power\_stamina 0.027 0.038 0.001 |  
## power\_strength 0.590 18.551 0.348 |  
##   
## Supplementary continuous variables  
## Dim.1 cos2 Dim.2 cos2 Dim.3 cos2   
## overall | -0.175 0.031 | -0.150 0.022 | 0.245 0.060 |  
## potential | -0.120 0.014 | -0.034 0.001 | 0.247 0.061 |  
## value\_eur | 0.197 0.039 | -0.151 0.023 | 0.217 0.047 |  
## pace | 0.018 0.000 | -0.514 0.264 | 0.234 0.055 |  
## shooting | -0.062 0.004 | -0.675 0.456 | 0.475 0.226 |  
## passing | 0.065 0.004 | -0.584 0.341 | -0.213 0.045 |  
## dribbling | 0.031 0.001 | -0.860 0.740 | 0.006 0.000 |  
## defending | 0.138 0.019 | 0.827 0.683 | -0.501 0.251 |  
## physic | 0.021 0.000 | 0.783 0.613 | 0.368 0.135 |  
##   
## Supplementary categories  
## Dist Dim.1 cos2 v.test Dim.2 cos2  
## Lean (170-) | 4.652 | 2.626 0.319 0.850 | -1.551 0.111  
## Lean (170-185) | 1.998 | 1.800 0.811 1.932 | -0.285 0.020  
## Lean (185+) | 2.085 | -1.133 0.295 -0.745 | 1.406 0.455  
## Normal (170-) | 4.242 | 1.185 0.078 0.383 | -3.678 0.752  
## Normal (170-185) | 1.039 | 0.458 0.194 0.464 | 0.123 0.014  
## Normal (185+) | 2.883 | -1.948 0.457 -1.850 | 2.016 0.489  
## Stocky (185+) | 7.414 | -7.103 0.918 -2.299 | -0.482 0.004  
## Unique | 0.437 | 0.020 0.002 0.088 | -0.217 0.246  
## v.test Dim.3 cos2 v.test   
## Lean (170-) -0.760 | -3.182 0.468 -2.325 |  
## Lean (170-185) -0.462 | -0.714 0.128 -1.729 |  
## Lean (185+) 1.399 | -0.381 0.033 -0.565 |  
## Normal (170-) -1.802 | -1.022 0.058 -0.747 |  
## Normal (170-185) 0.189 | -0.690 0.441 -1.577 |  
## Normal (185+) 2.898 | -0.166 0.003 -0.356 |  
## Stocky (185+) -0.236 | -0.869 0.014 -0.635 |  
## Unique -1.472 | 0.322 0.544 3.264 |

## AJOUTER LES NOMS DES JOUEURS

fifa\_100 <- data.frame(fifa\_100)  
rownames(fifa\_100) <- head(fifa22$short\_name, 100)

## RÉALISER UNE ACP

premiere\_acp <- PCA(X = fifa\_100, quanti.sup = idx\_var\_quanti\_illustratives, quali.sup = idx\_var\_quali\_illustratives, graph = FALSE)

## Warning in PCA(X = fifa\_100, quanti.sup = idx\_var\_quanti\_illustratives, :  
## Missing values are imputed by the mean of the variable: you should use the  
## imputePCA function of the missMDA package

summary(premiere\_acp)

##   
## Call:  
## PCA(X = fifa\_100, quanti.sup = idx\_var\_quanti\_illustratives,   
## quali.sup = idx\_var\_quali\_illustratives, graph = FALSE)   
##   
##   
## Eigenvalues  
## Dim.1 Dim.2 Dim.3 Dim.4 Dim.5 Dim.6 Dim.7  
## Variance 9.547 4.167 1.873 1.047 0.767 0.369 0.285  
## % of var. 50.248 21.933 9.860 5.509 4.037 1.941 1.501  
## Cumulative % of var. 50.248 72.181 82.041 87.549 91.586 93.527 95.029  
## Dim.8 Dim.9 Dim.10 Dim.11 Dim.12 Dim.13 Dim.14  
## Variance 0.217 0.167 0.156 0.130 0.080 0.055 0.046  
## % of var. 1.141 0.882 0.819 0.685 0.423 0.288 0.240  
## Cumulative % of var. 96.170 97.051 97.871 98.555 98.978 99.266 99.506  
## Dim.15 Dim.16 Dim.17 Dim.18 Dim.19  
## Variance 0.028 0.022 0.019 0.016 0.009  
## % of var. 0.149 0.115 0.100 0.085 0.045  
## Cumulative % of var. 99.655 99.770 99.870 99.955 100.000  
##   
## Individuals (the 10 first)  
## Dist Dim.1 ctr cos2 Dim.2 ctr  
## L, Messi | 3.928 | 0.864 0.078 0.048 | -3.479 2.905  
## R, Lewandowski | 2.857 | 0.591 0.037 0.043 | -0.374 0.033  
## Cristiano Ronaldo | 3.758 | 0.342 0.012 0.008 | -1.332 0.426  
## Neymar Jr | 3.933 | 1.321 0.183 0.113 | -3.624 3.151  
## K, De Bruyne | 2.615 | 2.024 0.429 0.599 | -0.590 0.084  
## J, Oblak | 7.200 | -6.992 5.121 0.943 | 0.261 0.016  
## K, Mbappé | 3.419 | 1.301 0.177 0.145 | -2.278 1.245  
## M, Neuer | 7.505 | -6.894 4.978 0.844 | 0.310 0.023  
## M, ter Stegen | 8.221 | -7.291 5.568 0.786 | 0.954 0.218  
## H, Kane | 2.843 | 0.364 0.014 0.016 | 0.467 0.052  
## cos2 Dim.3 ctr cos2   
## L, Messi 0.784 | 0.758 0.306 0.037 |  
## R, Lewandowski 0.017 | 2.560 3.497 0.803 |  
## Cristiano Ronaldo 0.126 | 3.179 5.395 0.716 |  
## Neymar Jr 0.849 | -0.177 0.017 0.002 |  
## K, De Bruyne 0.051 | 0.008 0.000 0.000 |  
## J, Oblak 0.001 | -0.362 0.070 0.003 |  
## K, Mbappé 0.444 | 1.812 1.753 0.281 |  
## M, Neuer 0.002 | 0.693 0.257 0.009 |  
## M, ter Stegen 0.013 | -0.068 0.002 0.000 |  
## H, Kane 0.027 | 2.501 3.339 0.774 |  
##   
## Variables (the 10 first)  
## Dim.1 ctr cos2 Dim.2 ctr cos2   
## height\_cm | -0.532 2.970 0.284 | 0.623 9.302 0.388 |  
## weight\_kg | -0.579 3.517 0.336 | 0.544 7.111 0.296 |  
## skill\_dribbling | 0.910 8.677 0.828 | -0.304 2.215 0.092 |  
## skill\_ball\_control | 0.930 9.052 0.864 | -0.228 1.251 0.052 |  
## movement\_acceleration | 0.649 4.416 0.422 | -0.532 6.793 0.283 |  
## movement\_agility | 0.618 4.006 0.382 | -0.692 11.478 0.478 |  
## power\_shot\_power | 0.570 3.404 0.325 | -0.283 1.921 0.080 |  
## power\_jumping | 0.133 0.185 0.018 | 0.484 5.630 0.235 |  
## power\_stamina | 0.931 9.080 0.867 | -0.021 0.011 0.000 |  
## power\_strength | -0.003 0.000 0.000 | 0.717 12.345 0.514 |  
## Dim.3 ctr cos2   
## height\_cm 0.374 7.474 0.140 |  
## weight\_kg 0.490 12.823 0.240 |  
## skill\_dribbling 0.181 1.756 0.033 |  
## skill\_ball\_control 0.151 1.209 0.023 |  
## movement\_acceleration 0.180 1.726 0.032 |  
## movement\_agility 0.013 0.010 0.000 |  
## power\_shot\_power 0.597 18.995 0.356 |  
## power\_jumping 0.421 9.482 0.178 |  
## power\_stamina 0.027 0.038 0.001 |  
## power\_strength 0.590 18.551 0.348 |  
##   
## Supplementary continuous variables  
## Dim.1 cos2 Dim.2 cos2 Dim.3 cos2   
## overall | -0.175 0.031 | -0.150 0.022 | 0.245 0.060 |  
## potential | -0.120 0.014 | -0.034 0.001 | 0.247 0.061 |  
## value\_eur | 0.197 0.039 | -0.151 0.023 | 0.217 0.047 |  
## pace | 0.018 0.000 | -0.514 0.264 | 0.234 0.055 |  
## shooting | -0.062 0.004 | -0.675 0.456 | 0.475 0.226 |  
## passing | 0.065 0.004 | -0.584 0.341 | -0.213 0.045 |  
## dribbling | 0.031 0.001 | -0.860 0.740 | 0.006 0.000 |  
## defending | 0.138 0.019 | 0.827 0.683 | -0.501 0.251 |  
## physic | 0.021 0.000 | 0.783 0.613 | 0.368 0.135 |  
##   
## Supplementary categories  
## Dist Dim.1 cos2 v.test Dim.2 cos2  
## Lean (170-) | 4.652 | 2.626 0.319 0.850 | -1.551 0.111  
## Lean (170-185) | 1.998 | 1.800 0.811 1.932 | -0.285 0.020  
## Lean (185+) | 2.085 | -1.133 0.295 -0.745 | 1.406 0.455  
## Normal (170-) | 4.242 | 1.185 0.078 0.383 | -3.678 0.752  
## Normal (170-185) | 1.039 | 0.458 0.194 0.464 | 0.123 0.014  
## Normal (185+) | 2.883 | -1.948 0.457 -1.850 | 2.016 0.489  
## Stocky (185+) | 7.414 | -7.103 0.918 -2.299 | -0.482 0.004  
## Unique | 0.437 | 0.020 0.002 0.088 | -0.217 0.246  
## v.test Dim.3 cos2 v.test   
## Lean (170-) -0.760 | -3.182 0.468 -2.325 |  
## Lean (170-185) -0.462 | -0.714 0.128 -1.729 |  
## Lean (185+) 1.399 | -0.381 0.033 -0.565 |  
## Normal (170-) -1.802 | -1.022 0.058 -0.747 |  
## Normal (170-185) 0.189 | -0.690 0.441 -1.577 |  
## Normal (185+) 2.898 | -0.166 0.003 -0.356 |  
## Stocky (185+) -0.236 | -0.869 0.014 -0.635 |  
## Unique -1.472 | 0.322 0.544 3.264 |

1.3 COMBIEN DE COMPOSANTES RETENIR ?

premiere\_acp$eig

## eigenvalue percentage of variance cumulative percentage of variance  
## comp 1 9.547132954 50.24806818 50.24807  
## comp 2 4.167230676 21.93279303 72.18086  
## comp 3 1.873381812 9.85990427 82.04077  
## comp 4 1.046617990 5.50851574 87.54928  
## comp 5 0.766990307 4.03679109 91.58607  
## comp 6 0.368793722 1.94101959 93.52709  
## comp 7 0.285276150 1.50145342 95.02855  
## comp 8 0.216790603 1.14100317 96.16955  
## comp 9 0.167494930 0.88155226 97.05110  
## comp 10 0.155690900 0.81942579 97.87053  
## comp 11 0.130089532 0.68468175 98.55521  
## comp 12 0.080333643 0.42280865 98.97802  
## comp 13 0.054677465 0.28777613 99.26579  
## comp 14 0.045622429 0.24011805 99.50591  
## comp 15 0.028253459 0.14870242 99.65461  
## comp 16 0.021933695 0.11544050 99.77005  
## comp 17 0.019068931 0.10036279 99.87042  
## comp 18 0.016061778 0.08453567 99.95495  
## comp 19 0.008559024 0.04504750 100.00000

L’objet eig nous renvoie un tableau qui nous permet de déterminer combien de composantes retenir dans l’ACP.

La règle de Kaiser-Guttman “Le nombre des valeurs-propres supérieures à l’unité d’une matrice d’inter-corrélation est égal au nombre de facteur à extraire”.

On peut grâce à l’ACP réduire la dimensionalité d’un jeu de donnée. On avait 19 colonnes dans le jeu de donnée intial mais grâce à 4 variables synthétiques (dimensions, composantes) on arriver à extraire 87% de l’information totale.

## QUEL % DE VARIANCE ARRIVE-T-ON À EXPLIQUER AVEC CES COMPOSANTES ?

## CONTRIBUTIONS DES VARIABLES DANS LA FORMATION DES COMPOSANTES (AXES)

premiere\_acp$var$contrib

## Dim.1 Dim.2 Dim.3 Dim.4  
## height\_cm 2.969591e+00 9.302472868 7.473852821 0.108123414  
## weight\_kg 3.516685e+00 7.110995568 12.823066979 0.007952457  
## skill\_dribbling 8.677484e+00 2.214770462 1.756164292 0.377360497  
## skill\_ball\_control 9.052026e+00 1.251070798 1.209293715 0.011484800  
## movement\_acceleration 4.415650e+00 6.792658190 1.726025351 3.583497720  
## movement\_agility 4.006064e+00 11.478163213 0.009627637 0.188492260  
## power\_shot\_power 3.404378e+00 1.920696119 18.995014817 0.235283975  
## power\_jumping 1.849522e-01 5.630315993 9.481707592 5.582832612  
## power\_stamina 9.079943e+00 0.010931925 0.038070624 0.009826481  
## power\_strength 7.626043e-05 12.344572598 18.551245032 0.004499476  
## mentality\_aggression 6.428601e+00 5.146111689 0.689934536 0.064416197  
## mentality\_interceptions 4.880132e+00 9.069639435 6.093199122 0.179446596  
## defending\_marking\_awareness 4.707598e+00 8.959020938 6.853953060 0.166890907  
## defending\_standing\_tackle 5.370849e+00 8.937033856 4.447340033 0.006249153  
## defending\_sliding\_tackle 4.723933e+00 9.657857789 5.902802382 0.021477938  
## goalkeeping\_diving 9.538249e+00 0.003134013 1.379628587 0.628001599  
## goalkeeping\_handling 9.650193e+00 0.008674713 1.361391417 0.366608847  
## goalkeeping\_kicking 9.374658e+00 0.000713271 1.169287732 0.665735973  
## goalkeeping\_speed 1.893860e-02 0.161166562 0.038394271 87.791819100  
## Dim.5  
## height\_cm 5.752952e+00  
## weight\_kg 2.558978e+00  
## skill\_dribbling 1.291465e+00  
## skill\_ball\_control 1.999736e+00  
## movement\_acceleration 5.140964e+00  
## movement\_agility 2.745547e+00  
## power\_shot\_power 6.530748e+00  
## power\_jumping 6.140018e+01  
## power\_stamina 2.790492e-04  
## power\_strength 4.894286e-01  
## mentality\_aggression 1.261596e+00  
## mentality\_interceptions 3.811871e-01  
## defending\_marking\_awareness 1.833431e-01  
## defending\_standing\_tackle 2.376443e-01  
## defending\_sliding\_tackle 9.057635e-02  
## goalkeeping\_diving 5.268139e-01  
## goalkeeping\_handling 4.699497e-01  
## goalkeeping\_kicking 2.820795e-01  
## goalkeeping\_speed 8.656534e+00

La variable goalkeeping\_handling contribue à hauteur de 9.53% à former l’axe 1. La variable height\_cm contribue à hauteur de 2.9% à former l’axe 1, la variable goalkeeping\_diving contribue à hauteur de 9.53% à former l’axe 1.

Si on fait la somme de toutes les contributions des variables sur chaque axe on trouve 100.

sum(premiere\_acp$var$contrib[, 1])

## [1] 100

### Afficher les poids des variables qui forment l’axe 1

sort(premiere\_acp$var$contrib[, 1], decreasing = TRUE)

## goalkeeping\_handling goalkeeping\_diving   
## 9.650193e+00 9.538249e+00   
## goalkeeping\_kicking power\_stamina   
## 9.374658e+00 9.079943e+00   
## skill\_ball\_control skill\_dribbling   
## 9.052026e+00 8.677484e+00   
## mentality\_aggression defending\_standing\_tackle   
## 6.428601e+00 5.370849e+00   
## mentality\_interceptions defending\_sliding\_tackle   
## 4.880132e+00 4.723933e+00   
## defending\_marking\_awareness movement\_acceleration   
## 4.707598e+00 4.415650e+00   
## movement\_agility weight\_kg   
## 4.006064e+00 3.516685e+00   
## power\_shot\_power height\_cm   
## 3.404378e+00 2.969591e+00   
## power\_jumping goalkeeping\_speed   
## 1.849522e-01 1.893860e-02   
## power\_strength   
## 7.626043e-05

On voit par exemple que sur l’axe 1 la variable goalkeeping\_handling a une plus grande contribution que les autres.

### Afficher les poids des variables qui forment l’axe 2

sort(premiere\_acp$var$contrib[, 2], decreasing = TRUE)

## power\_strength movement\_agility   
## 12.344572598 11.478163213   
## defending\_sliding\_tackle height\_cm   
## 9.657857789 9.302472868   
## mentality\_interceptions defending\_marking\_awareness   
## 9.069639435 8.959020938   
## defending\_standing\_tackle weight\_kg   
## 8.937033856 7.110995568   
## movement\_acceleration power\_jumping   
## 6.792658190 5.630315993   
## mentality\_aggression skill\_dribbling   
## 5.146111689 2.214770462   
## power\_shot\_power skill\_ball\_control   
## 1.920696119 1.251070798   
## goalkeeping\_speed power\_stamina   
## 0.161166562 0.010931925   
## goalkeeping\_handling goalkeeping\_diving   
## 0.008674713 0.003134013   
## goalkeeping\_kicking   
## 0.000713271

## LES COORDONNÉES DES VARIABLES SUR LES AXES

On aimerait voir le signe de chaque variable dans la construction des axes.

premiere\_acp$var$coord

## Dim.1 Dim.2 Dim.3 Dim.4  
## height\_cm -0.532457323 0.62261987 0.37418418 -0.033639844  
## weight\_kg -0.579432980 0.54436347 0.49012754 0.009123149  
## skill\_dribbling 0.910192790 -0.30380025 0.18138264 -0.062845229  
## skill\_ball\_control 0.929628403 -0.22833091 0.15051475 -0.010963667  
## movement\_acceleration 0.649282662 -0.53203922 0.17981948 0.193663450  
## movement\_agility 0.618436925 -0.69160794 0.01342991 0.044416145  
## power\_shot\_power 0.570105693 -0.28291313 0.59653093 -0.049623829  
## power\_jumping 0.132882020 0.48438441 0.42146006 0.241724907  
## power\_stamina 0.931060794 -0.02134382 0.02670596 -0.010141288  
## power\_strength -0.002698274 0.71723554 0.58952154 0.006862385  
## mentality\_aggression 0.783420124 0.46308784 0.11368865 0.025965198  
## mentality\_interceptions 0.682577957 0.61477866 -0.33785927 0.043337286  
## defending\_marking\_awareness 0.670403311 0.61101806 -0.35833045 0.041793663  
## defending\_standing\_tackle 0.716074092 0.61026782 -0.28864452 -0.008087321  
## defending\_sliding\_tackle 0.671565475 0.63440146 -0.33253876 0.014993064  
## goalkeeping\_diving -0.954268987 0.01142810 -0.16076601 0.081072669  
## goalkeeping\_handling -0.959852468 0.01901303 -0.15969990 0.061943475  
## goalkeeping\_kicking -0.946050246 -0.00545194 -0.14800413 0.083472825  
## goalkeeping\_speed 0.042521682 -0.08195232 -0.02681923 0.958564016  
## Dim.5  
## height\_cm -0.21005853  
## weight\_kg -0.14009679  
## skill\_dribbling -0.09952592  
## skill\_ball\_control -0.12384579  
## movement\_acceleration 0.19857165  
## movement\_agility 0.14511403  
## power\_shot\_power -0.22380842  
## power\_jumping 0.68624589  
## power\_stamina 0.00146297  
## power\_strength -0.06126883  
## mentality\_aggression 0.09836829  
## mentality\_interceptions -0.05407096  
## defending\_marking\_awareness -0.03749965  
## defending\_standing\_tackle -0.04269319  
## defending\_sliding\_tackle 0.02635739  
## goalkeeping\_diving 0.06356580  
## goalkeeping\_handling 0.06003723  
## goalkeeping\_kicking 0.04651368  
## goalkeeping\_speed -0.25767183

Sur l’axe 1, les variables power\_stamina, skill\_ball\_control, skill\_dribbling ont des coordonnées positives. cela signifie que si un individu du jeu de données a une coordonnée positive c’est que cet individu est représenté par ces variables.

### Afficher les 5 premières coordonnées positives sur l’axe 1

sort(premiere\_acp$var$coord[, 1], decreasing = TRUE)[1:5]

## power\_stamina skill\_ball\_control skill\_dribbling   
## 0.9310608 0.9296284 0.9101928   
## mentality\_aggression defending\_standing\_tackle   
## 0.7834201 0.7160741

### Afficher les 5 premières coordonnées négatives sur l’axe 1

sort(premiere\_acp$var$coord[, 1], decreasing = FALSE)[1:5]

## goalkeeping\_handling goalkeeping\_diving goalkeeping\_kicking   
## -0.9598525 -0.9542690 -0.9460502   
## weight\_kg height\_cm   
## -0.5794330 -0.5324573

### Afficher les 5 premières coordonnées positives sur l’axe 2

Sur la dimension 2 (composante 2) les variables power\_strength et defending\_sliding\_tackle ont des coordonnées positives. Si on devrait les représenter sur un espace orhtonomé ces variables seraient à droite.

sort(premiere\_acp$var$coord[, 2], decreasing = TRUE)[1:5]

## power\_strength defending\_sliding\_tackle   
## 0.7172355 0.6344015   
## height\_cm mentality\_interceptions   
## 0.6226199 0.6147787   
## defending\_marking\_awareness   
## 0.6110181

### Afficher les 5 premières coordonnées négatives sur l’axe 2

sort(premiere\_acp$var$coord[, 2], decreasing = FALSE)[1:5]

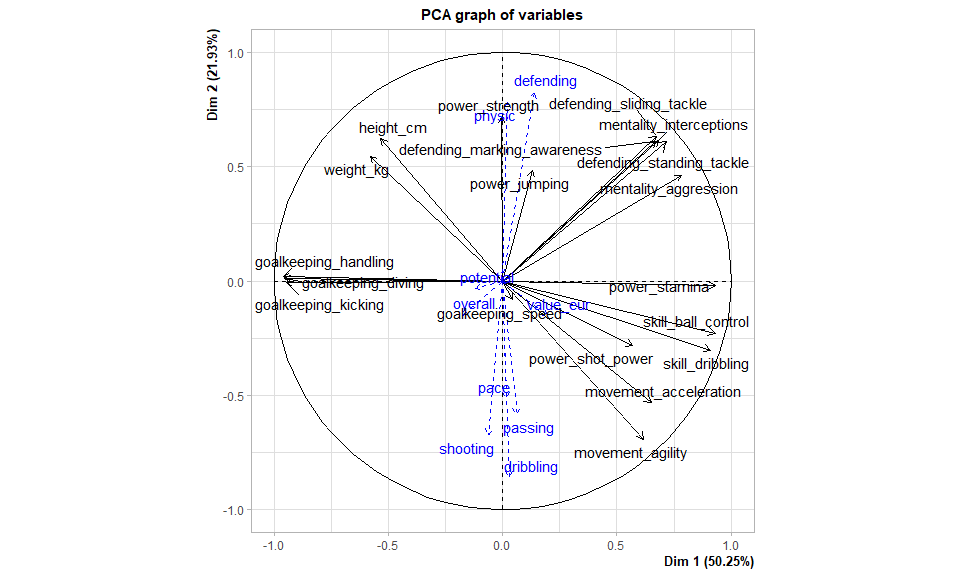
## movement\_agility movement\_acceleration skill\_dribbling   
## -0.6916079 -0.5320392 -0.3038003   
## power\_shot\_power skill\_ball\_control   
## -0.2829131 -0.2283309

## REPRÉSENTATION SIMULTANÉE DES DEUX PREMIERS AXES

On appelle ce graphique le cercle des corrélations. Il représente les corrélations de chaque

La fonction plot.PCA prend en argument l’objet ACP qu’on a créé, puis les axes qu’on veut représenter et l’argument choix prend en valeur : “var”, “ind”.

plot.PCA(premiere\_acp, axes = c(1, 2), choix = "var")



## LES INDIVIDUS

### Afficher les 10 premiers individus ayant des coordonnées positives sur l’axe 1

On explore toujours les propriétés de l’objet premiere\_acp et on regarde dans les individus. Si on affiche le tableau global on a les coordonnées de chaque joueur sur chaque axe (dimension , composante).

sort(premiere\_acp$ind$coord[, 1], decreasing = TRUE)[1:10]

## N, Kanté A, Robertson J, Kimmich M, Verratti Jesús Navas   
## 3.142613 2.788556 2.762806 2.734115 2.625595   
## Marcos Llorente A, Hakimi Jordi Alba João Cancelo L, Modrić   
## 2.512007 2.475914 2.436642 2.376720 2.372374

N, Kante est le plus à droite de l’axe 1, a la plus grande coordonnée positive.

### Afficher les 10 premiers individus ayant des coordonnées négatives sur l’axe 1

sort(premiere\_acp$ind$coord[, 1], decreasing = FALSE)[1:10]

## T, Courtois Alisson K, Casteels S, Handanovič P, Gulácsi   
## -7.868107 -7.561502 -7.535096 -7.521622 -7.422478   
## W, Szczęsny M, ter Stegen G, Donnarumma K, Schmeichel J, Oblak   
## -7.294438 -7.290675 -7.129166 -7.103182 -6.991911

### Afficher les 10 premiers individus ayant des coordonnées positives sur l’axe 2

sort(premiere\_acp$ind$coord[, 2], decreasing = TRUE)[1:10]

## V, van Dijk G, Chiellini K, Koulibaly M, Hummels Piqué M, de Ligt   
## 3.910655 3.882630 3.829444 3.609459 3.567733 3.427404   
## A, Laporte L, Bonucci M, Škriniar Rúben Dias   
## 3.396182 3.384239 3.297347 3.295519

### Afficher les 10 premiers individus ayant des coordonnées négatives sur l’axe 2

sort(premiere\_acp$ind$coord[, 2], decreasing = FALSE)[1:10]

## L, Insigne A, Gómez Neymar Jr L, Messi R, Mahrez   
## -5.338622 -3.677799 -3.623794 -3.479110 -3.359841   
## E, Hazard K, Coman R, Sterling Bernardo Silva J, Sancho   
## -3.209868 -3.203207 -3.036538 -2.791582 -2.749813

## PRODUIRE DES GRAPHIQUES AVEC FACTOEXTRA

library(factoextra)

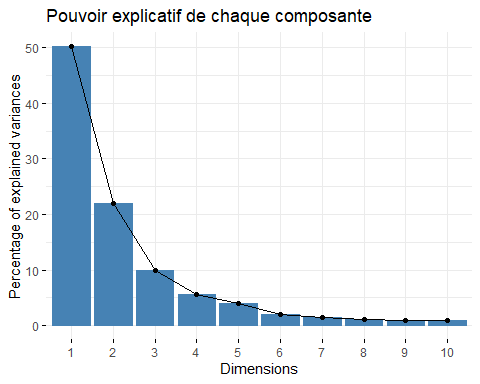
## Le chargement a nécessité le package : ggplot2

## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa

### Représenter le % des variances expliquées par chaque axe

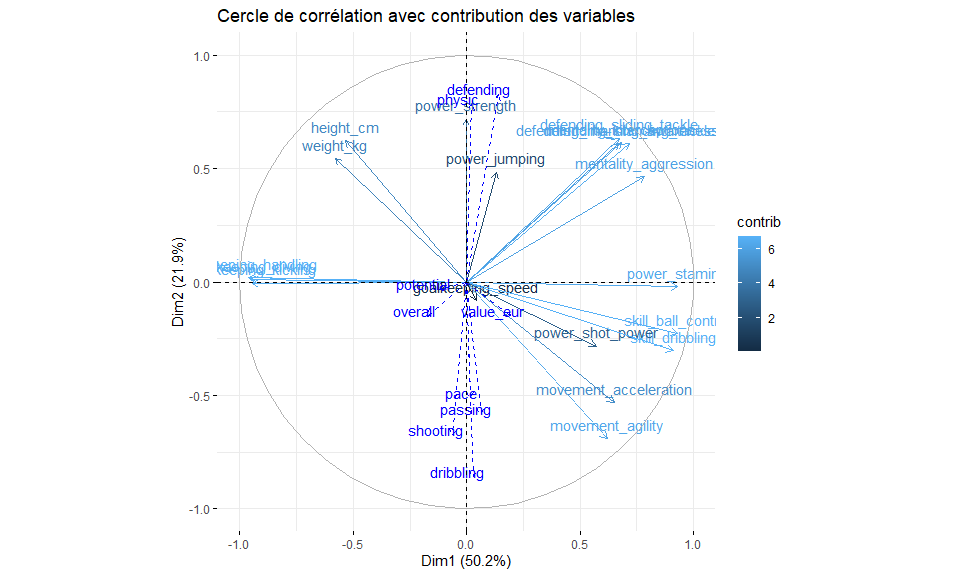
Ce graphique représente le pouvoir explicatif de chaque axe. La dimension 1 explique 50% de la variance du jeu de données.

factoextra::fviz\_screeplot(premiere\_acp, title = "Pouvoir explicatif de chaque composante")



### Représenter le cercle de corrélation puis mettre en avant la contribution de chaque variable

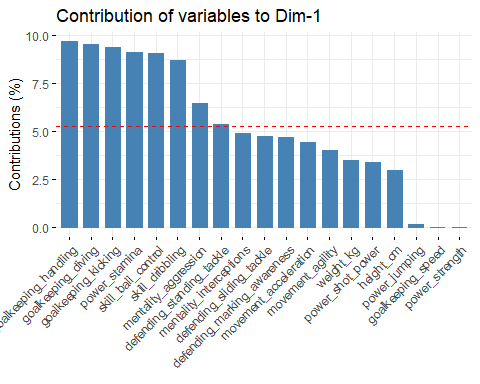
factoextra::fviz\_pca\_var(premiere\_acp, axes = c(1,2), col.var="contrib", title= "Cercle de corrélation avec contribution des variables")



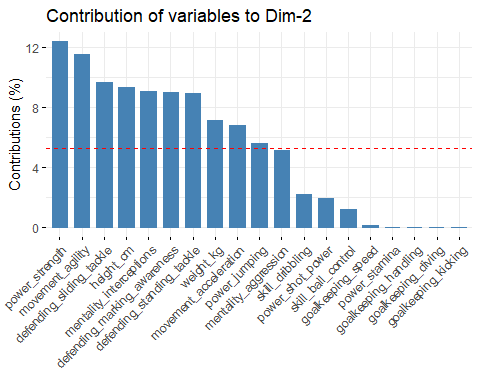
### Représenter le % de contribution de chaque variable sur les axes 1 et 2

la fonction fviz\_contrib montre le pourcentage de contribution de chaque variable sur l’axe choisi.

fviz\_contrib(premiere\_acp, axes = 1, choice = "var")



fviz\_contrib(premiere\_acp, axes = 2, choice = "var")



# CLASSIFICATION ASCENDANTE HIERARCHIQUE

La classification ascendante hiérarchique nous permet de regrouper des joueurs qui ont des caractéristiques similaires dans un même groupe qu’on appelle cluster. On choisit un nombre de cluster à produire et la CAH fait en sorte que chaque cluster soit différent d’un autre mais qu’au sein de chaque cluster les individus se ressemblent.

### CLASSER LES JOUEURS DANS 5 CLUSTERS

On va utiliser une fonction qui vient du package FactoMineR qui s’appelle HCPC on lui donne en agument l’objet ACP qu’on a créé puis on précise le nombre de cluster qu’on veut créer.

library(FactoMineR)  
classsif <- HCPC(premiere\_acp, nb.clust = 5, graph = FALSE)

## QUELLE(S) COMPOSANTE(S) CARACTÉRISENT LES PLUS LES INDIVIDUS DE CHAQUE CLUSTER ?

Il nous crée un objet classif qui contient tous les résultats de la classification.

La propriété desc.axes renvoie une description de chaque cluster

Pour la classification, l’algrithme a déterminé qu’il est optimal de ne retenir que 3 dimensions pour différencier les joueurs et les classer dans chaque cluster.

classsif$desc.axes

##   
## Link between the cluster variable and the quantitative variables  
## ================================================================  
## Eta2 P-value  
## Dim.1 0.9710515 3.985630e-72  
## Dim.2 0.8407178 5.188359e-37  
## Dim.3 0.6293291 1.039484e-19  
##   
## Description of each cluster by quantitative variables  
## =====================================================  
## $`1`  
## v.test Mean in category Overall mean sd in category Overall sd  
## Dim.1 -9.61829 -7.11018 6.931955e-16 0.4384079 3.089844  
## p.value  
## Dim.1 6.693444e-22  
##   
## $`2`  
## v.test Mean in category Overall mean sd in category Overall sd  
## Dim.3 6.592117 2.455677 4.058733e-16 0.7196673 1.368715  
## p.value  
## Dim.3 4.335989e-11  
##   
## $`3`  
## v.test Mean in category Overall mean sd in category Overall sd  
## Dim.1 2.063221 1.002524 6.931955e-16 0.4640714 3.089844  
## Dim.2 -7.184421 -2.306366 -1.407095e-15 1.0784803 2.041380  
## p.value  
## Dim.1 3.909160e-02  
## Dim.2 6.749241e-13  
##   
## $`4`  
## v.test Mean in category Overall mean sd in category Overall sd  
## Dim.2 7.771677 2.917443 -1.407095e-15 0.6613834 2.041380  
## Dim.1 2.214230 1.258123 6.931955e-16 0.5733747 3.089844  
## p.value  
## Dim.2 7.745396e-15  
## Dim.1 2.681297e-02  
##   
## $`5`  
## v.test Mean in category Overall mean sd in category Overall sd  
## Dim.1 3.673506 2.212603 6.931955e-16 0.5615386 3.089844  
## Dim.3 -5.086956 -1.357242 4.058733e-16 0.8761187 1.368715  
## p.value  
## Dim.1 2.39245e-04  
## Dim.3 3.63857e-07

Pour la classification, l’algrithme a déterminé qu’il est optimal de ne retenir que 3 dimensions pour différencier les joueurs et les classer dans chaque cluster.

2.3 QUELS INDIVIDUS SONT LES PLUS REPRÉSENTATIFS DANS CHAQUE CLUSTER ? 2.4 QUELS INDIVIDUS SONT LES MOINS REPRÉSENTATIFS DANS CHAQUE CLUSTER ?