Daanish Ahsan
Working with Data MTHM501
Candidate Number: 124070

Working with Data Project

Introduction

Fifa is a well known and popular sports game that has been around since the late nineties. As football and Fifa have grown vastly in popularity over the years, so too has the gameplay within the game, which relies more and more heavily every year on player data. As such, player data has become a vital component, and so with each iteration of Fifa, player data statistics are becoming increasingly detailed and accurate. This report will focus on 'what interesting insights can be generated from an analysis of a Fifa 20 dataset?', and examines a multitude of variables, but also focuses generally on the top clubs and players.

This analysis was a broad one, and so several topic areas were engaged with. These were high quality graphics, handling missing data, mapping data geographically, and random forest modelling.

Objectives

Overall ratings, position, value, wages, nationality and attributes and the relationships between them will be the main variables examined in this report. As the given question is a broad one, several topic areas will be used to try and gain an understanding of how different methods return differing insights. Thus the main objective was to simply create interesting findings that were not easily observable in the dataset. Wrangling the data before each analysis stage is vital to ensure that the data is in the correct format in order to prevent errors or incorrect result. It is also important to examine the dataset for missing data, and impute values for these with the correct method, to try and come up with the most accurate results possible. Visualisation of the data is a helpful step in analysis, and mapping and graphics were used to aid this. Random forest modelling was used as a last step to examine how the number of predictors affects the error.

Data

The dataset was downloaded from a Kaggle page, before which it had been scraped by a Kaggle from the Sofifa website. The dataset had 18,278 rows- a row per player, and 104 columns for each feature for the player. These variables were a combination of numerical, categorical and descriptive, featuring interesting variables such as player traits and player wages. For every stage of analysis, varying amounts of data wrangling were required. For handling missing data, a package was simply used to delete 10% of values from a duplicate of the original data set; little actual wrangling was needed other than completing the imputed values into the empty dataset. Geographical mapping required much more wrangling, including recoding the dataset country names to match those in the world map dataset, making data types compatible and joining dataframes. Graphics generally did not need much more wrangling beyond using aggregate functions, although calculating BMI with imputed values was slightly more complicated. Melting of data was sometimes required and the correlation plot was very challenging to scale correctly. Random forest consisted of creating a training and test set in terms of wrangling. Evidence of data wrangling can be seen in the R code appendix.

Daanish Ahsan
Working with Data MTHM501
Candidate Number: 124070

Results

Briefly looking at some interesting summary statistics for particular variables from the Fifa 20 dataset, it is observed that the overall rating for Fifa 20 players ranges from 48 to 94, with the average being 66. The average player is 25 years old, which reflects the short career time of football players; however, of more interest is the vast range in release clause, which ranges from €13,000 to a massive €195,800,000. This huge release clause is that of Lionel Messi, widely regarded as the world's best player (*BBC, 2019, 'Lionel Messi: Barcelona forward wins Ballon d'or for record sixth time*).

Figure 1

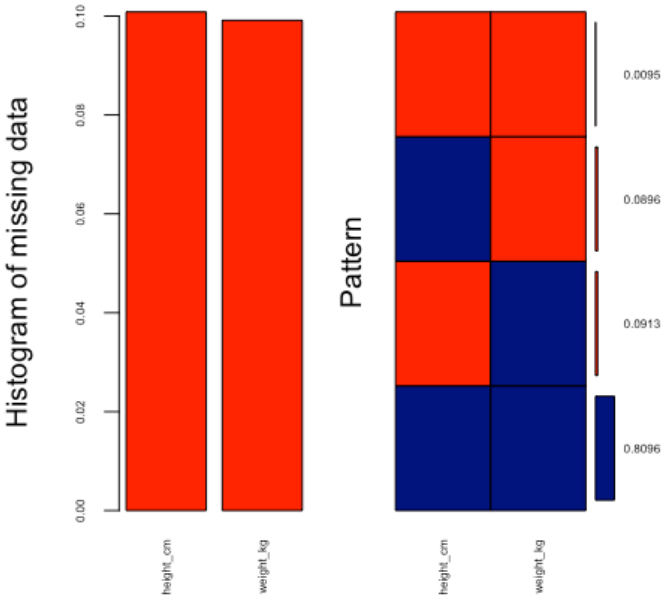
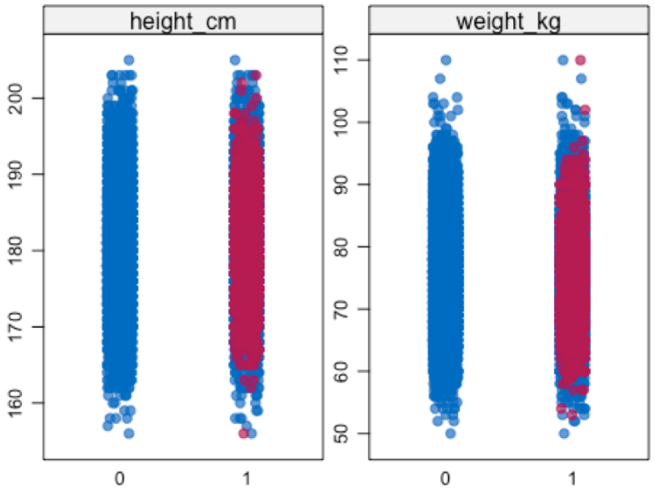


Figure 2



Handling Missing Data

This dataset was free of missing values, thus the Mice package was used to randomly delete 10% of all values in the height and weight columns of the dataset. Examining Figure 1, the visual representation of the missing data shows that about 81% of the data in both columns are not missing any data, with 9% missing height data. The histogram confirms that 10% of the data from each column is missing. Imputed values were generated to replace these missing values, using a random forest model to generate these values. The plot in Figure 2 shows how accurate these imputed height and weight values are in relation to the actual observed values. The red is the imputed, whilst the blue is the observed values; it is clear the the imputed values are on the whole very accurate, with only some outliers. However, this may be because with only 10% of the data for these columns deleted, the random forest model which generated these imputed values was able to be very accurate; with a lower level of data left, the imputed values would invariably become more inaccurate. Figure 3 shows the actual BMI against overall rating by using observed weight and height values, whilst Figure 4 shows the same but uses the imputed values of weight and height. Figure 3 shows that generally, the vast majority of players are in the healthy BMI range, although there is a very large outlier with a BMI near 35- this is likely Akinfenwa, a player renowned for his strength and over 100 kg weight. Indeed, it makes sense that most players have a BMI of less than 25 as they are athletes. Figure 4 shows more spread and outliers in the data, because 10% of the values of height and weight have been imputed using random forest and are not actual observations. This suggests that imputed data, whilst very helpful in replacing missing data,

increases inaccuracy compared to using actual variables. Indeed, this inaccuracy problem would be worse if larger amount of data were missing.

Figure 3

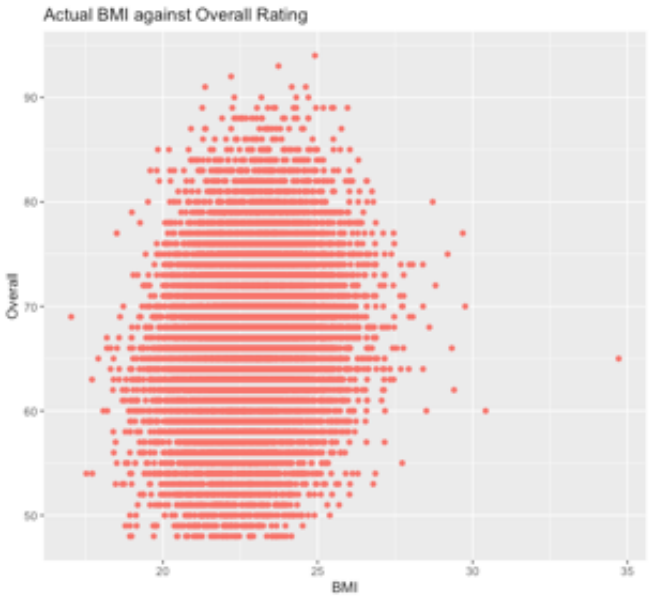
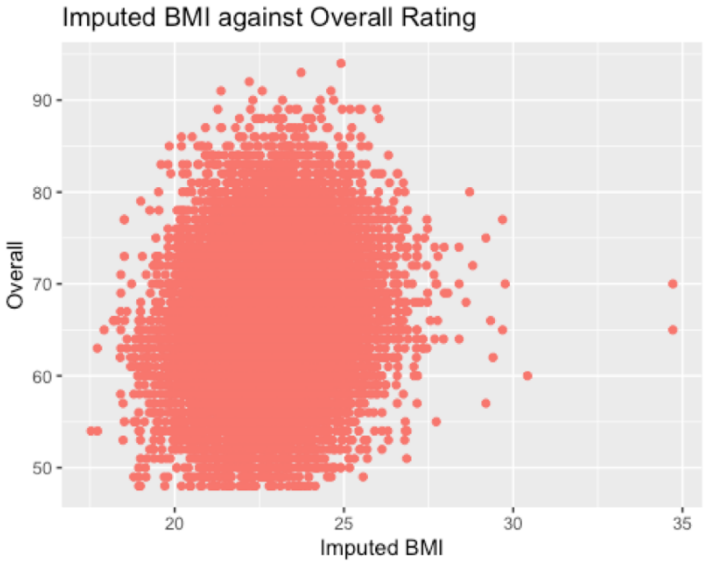


Figure 4



Mapping

The map graphic was used as it is a particularly useful visual aid in seeing how many players are from each country. Figure 5 is clear here in showing that the vast majority of players are from the UK, France, Germany, Spain and Brazil and Argentina. The highest concentration of player nationality is the UK, with about 2500 players from there, and the vast majority from England. Indeed, the UK is well known for the popularity of football, the wealth of the domestic premier league, and is where football as a sport originated from (*Football.History.org, 2019*). About 1600 players come from Germany, with around 1200 from Spain and France; Western Europe and Germany are well known for having the most famous football leagues in the world, such as LaLiga and the Bundesliga. Unsurprisingly, many football players also come from Brazil and Argentina, at around 1100 each. In international competitions too, these countries are well known for winning trophies (except perhaps England), and so the fact that most players are from these countries makes sense. Figure 5 also shows that much of Scandinavia contributes around 700 players, as do part of Eastern Europe. Chile and Colombia contribute close to a thousand players each, whilst the US and Mexico have close to 800 players- a surprisingly high statistic for the US. Africa is by far the lowest contributor of players, likely due to lack of funding in schools and clubs there to make football a grassroots sport. Other interesting findings are that Saudi Arabia and China contribute a similar number of players to Northern and Eastern Europe, perhaps reflecting the huge amount of money these countries are pouring into football (*Bloomberg, 2017, 'How China is Spending Billions to conquer World Soccer'*).

Daanish Ahsan
Working with Data MTHM501
Candidate Number: 124070

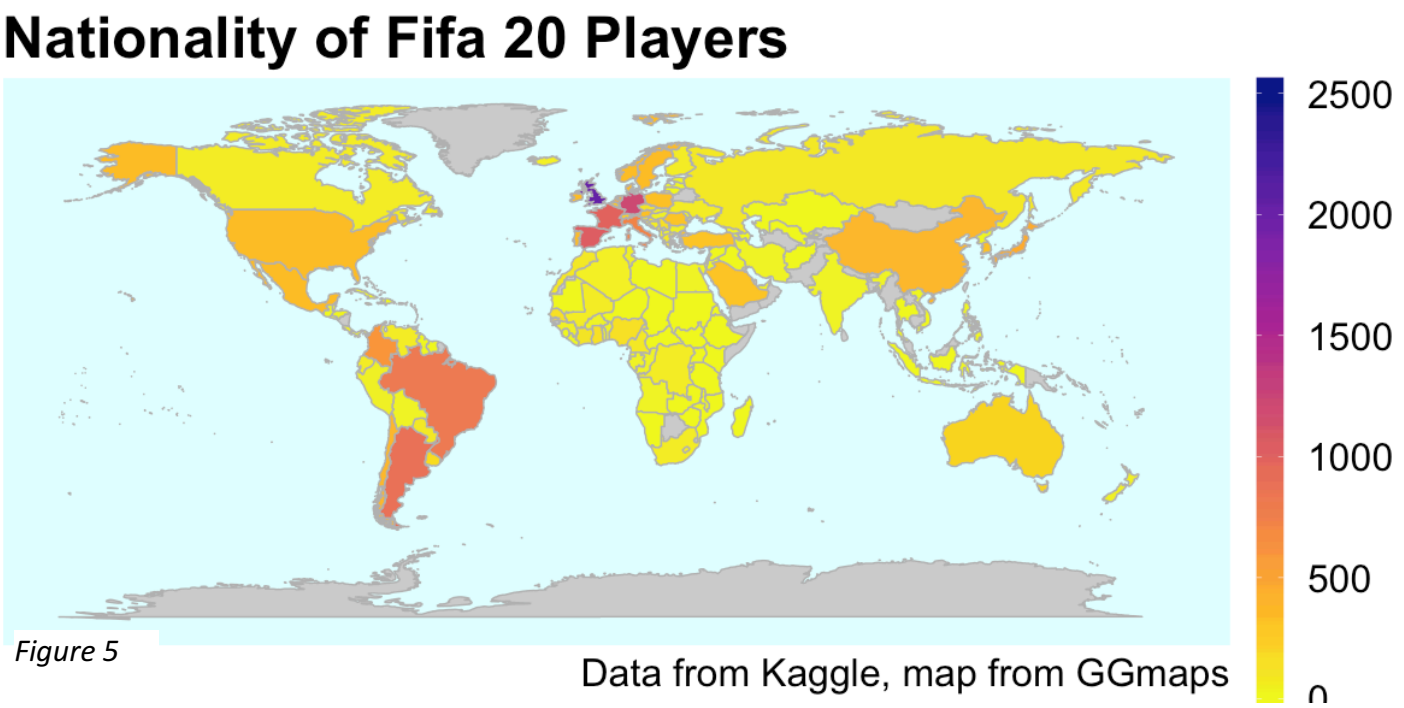


Figure 5

Graphics

Analysing the dataset further, individual player positions were categorised by position group, and reserves, substitutes and no position players were excluded. The result is figure 6, which shows that the vast majority of players are midfielders at around 2300 players, followed by defenders are 1300, goalkeepers at 600 and forwards at 500. However, it should be noted that midfield consists of by far the largest number of individual positions, followed by defenders. This means that as there are more playing positions in these groups, it is not unexpected that there are so many midfielders and defenders. Forwards are the most competitive position to play, and so the lower number of them may reflect the higher standards that are required for them. After all, it is goals that win games; the best forwards, not other position groups, are regarded as the best players of all time. This view of forwards may also explain the much higher wages they receive in comparison to other position groups (*Forbes, 2019, The World's Highest-Paid Soccer Players 2019'*). Fig 7 again looks at positional group, but examines how average attributes differ depending on positional group. As expected goalkeepers have good goalkeeping attributes, but as they cannot be compared to outfield players, comparing between outfield player attributes is of more interest. As expected, defenders are best at defending, forwards are best at shooting and midfielders are best at passing. There are also some very interesting results here; forwards have the lowest pace, and also seem to have better physical attributes than defenders,

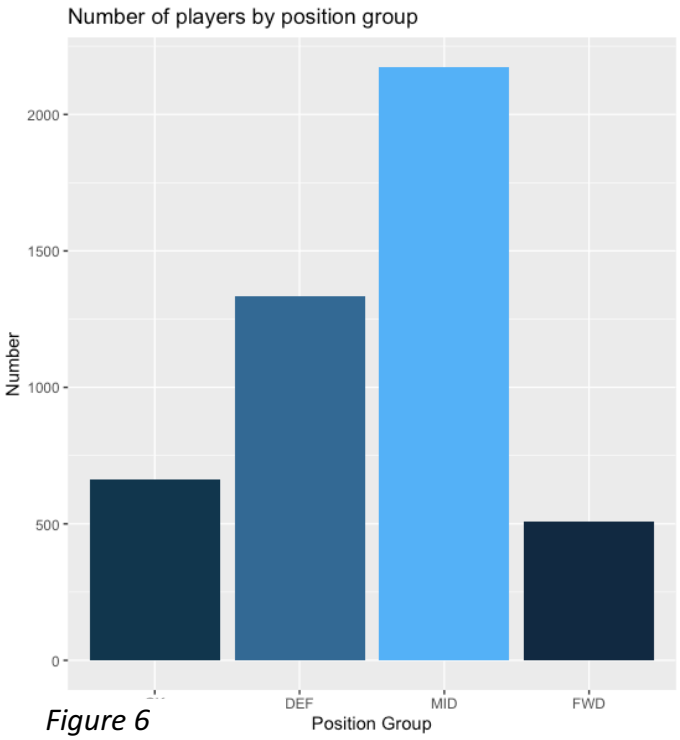
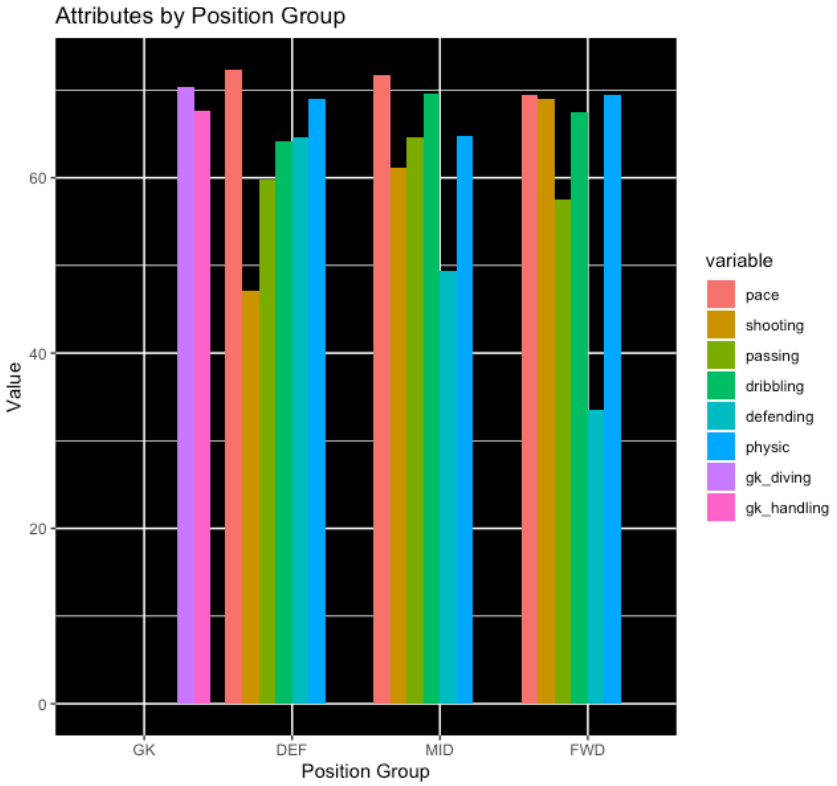


Figure 6

Figure 7



whilst midfielders seem to be marginally better at dribbling then forwards too. One explanation is that over time forwards have become more "complete players", in that they are more physical and able to engage in hold up play, whilst defenders have become more speed and play rather than strength oriented.

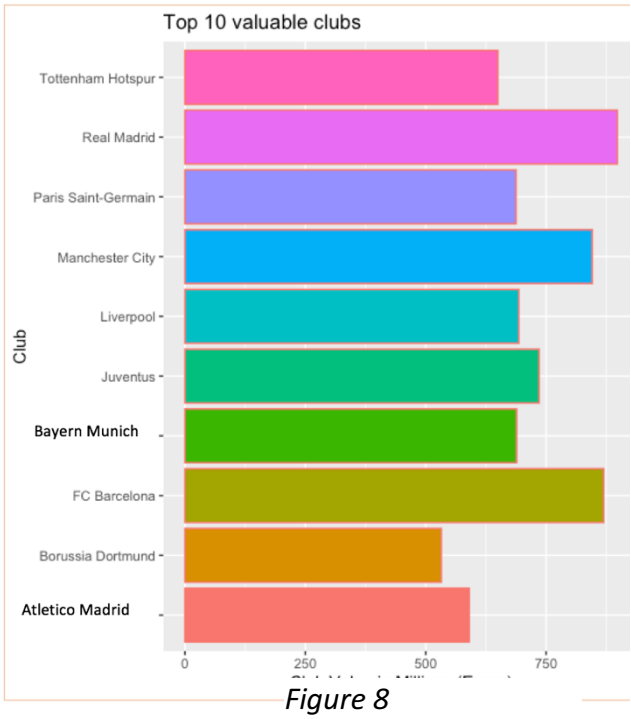


Figure 8

Player value is based on many interesting metrics (*Sportkeeda, 2019, 'Transfer Values-Calculation explained*) including age, position, wage, the strength of the league they play in and brand value of the player. Figure 8 shows the top 10 most valuable clubs in millions (euros), based not on the traditionally used metric of revenue, but on the total value of players at each club. Despite Cristiano Ronaldo leaving, Real Madrid is valued at around €895 million, with Barcelona close behind at €875 million. Manchester City are not far off, at around €840 million, but virtually all other top 10 clubs are below the €750 million mark, indicating that these 3 clubs have the current best, or potential best players. These clubs are well known for their lavish spending however, and the fact that Spurs and Dortmund are on this chart indicate they are scouting or growing players very effectively. Table 1 is a table of the top clubs by average player potential, and contains every team in the 10 most valuable clubs by player value except Dortmund. The higher average player potentials of Real Madrid, Barcelona and Manchester City indicate a correlation between player potential and club value.

Table 1

club	Average Player Potential
Atletico Madrid	83.06061
Colombia	83.00000
FC Barcelona	85.72727
FC Bayern Munchen	85.82609
Juventus	83.24242
Liverpool	82.69697
Manchester City	84.30303
Paris Saint-Germain	82.93939
Real Madrid	85.72727
Tottenham Hotspur	83.24242

Daanish Ahsan
Working with Data MTHM501
Candidate Number: 124070

Figure 9 examines the relationship between player rating and value, whilst Figure 10 examines the link between rating and wages. It is clear from Figure 9 that there is an exponential relationship between rating and value, which especially when the rating begins to exceed value. The values of between €75 and €100 million seem to be reserved for only the very best players with ratings of 90 and above. A similar relationship is evident in Figure 10, but arguably weekly wages are a little more inelastic in response to rating. It seems that generally, most players earn below €200,000 a week, but the very best with rating above 90 can earn double this. Examining wage a little more closely, Figure 11 shows that generally, wages are lower in the early and late stages of a football career. However, wages seem to peak when players are between 27 and 32, which is widely suggested to be when players are at their peak performance career wise (*BBC, 2014, 'When do footballers reach their peak')*. Again using overall as a variable, Figure 12 examines the link between player rating and shirt number. It seems the majority of players have a shirt number below 50, but there is a clear scatter across other numbers. However, there is also an observable trend showing that as player rating rises from 80 and above, shirt number falls quite dramatically, to about 30 and below. Looking closer at the players rated 85 and above, the shirt numbers do not generally exceed 10; this is an interesting finding as the number 10, 7 and 9 shirts are generally associated with only the best attacking players (*firsttouchonline, 2018, 'The most Notorious Shirt Numbers in Football History)*. That there are such a large amount of players of all ratings who wear lower numbered shirts, suggests that lower rated players also want the association with such esteemed shirt numbers.

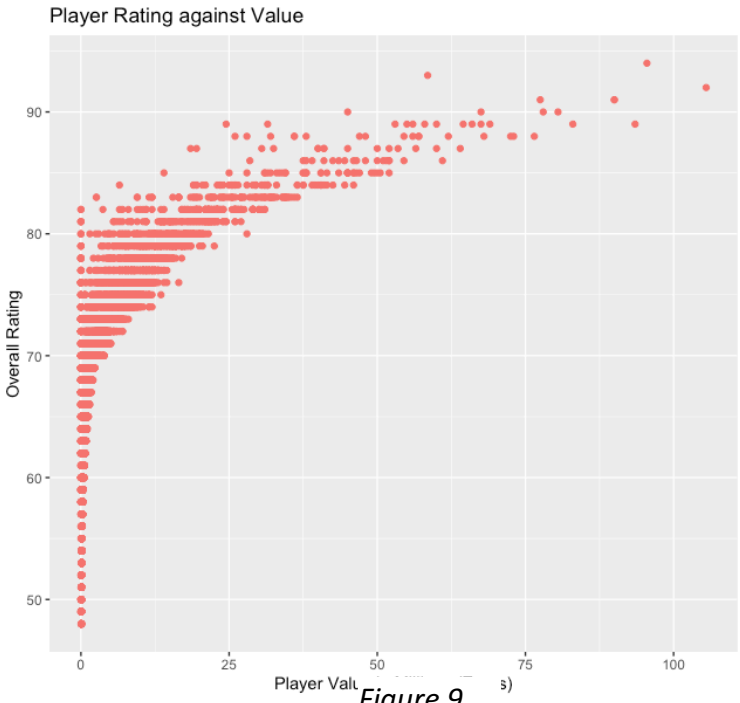


Figure 9

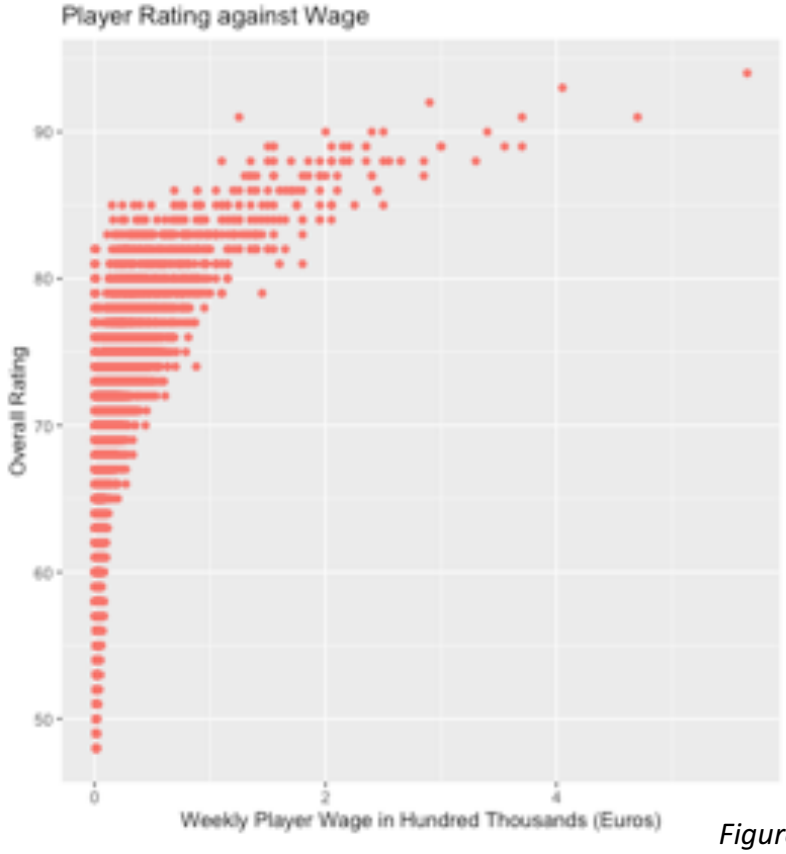


Figure 10

Daanish Ahsan
Working with Data MTHM501
Candidate Number: 124070

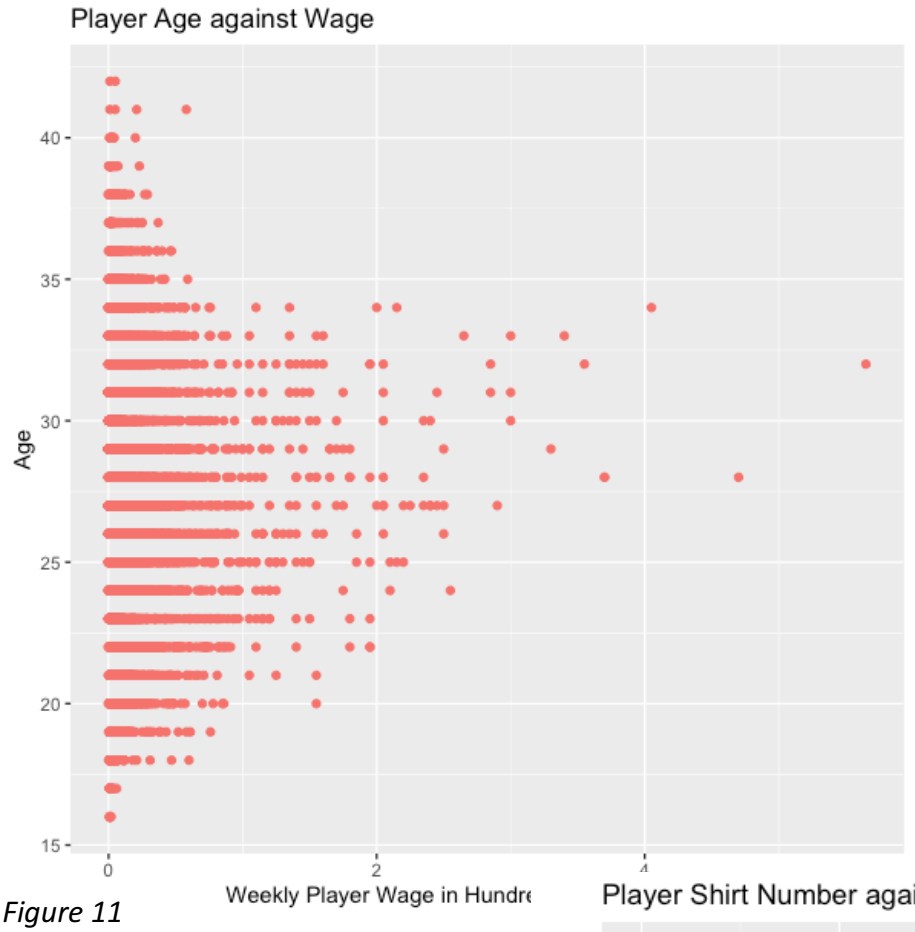


Figure 11

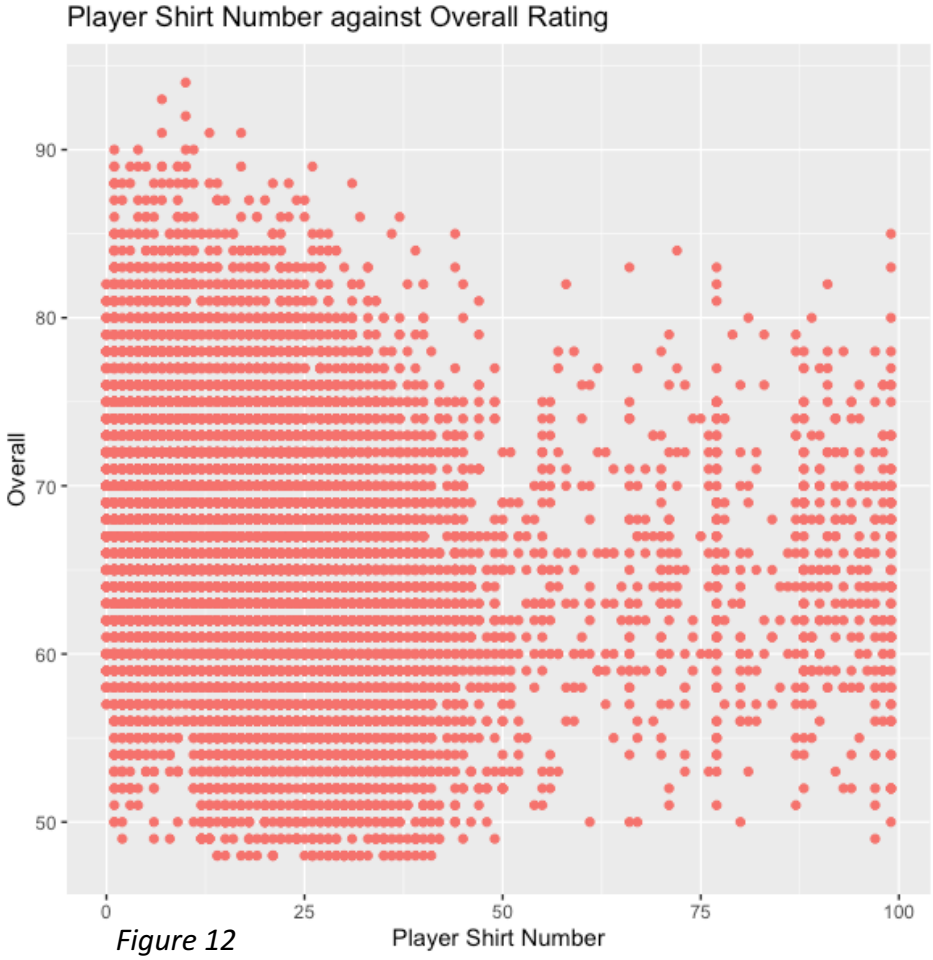


Figure 12

Daanish Ahsan
Working with Data MTHM501
Candidate Number: 124070

Correlation Heatmap

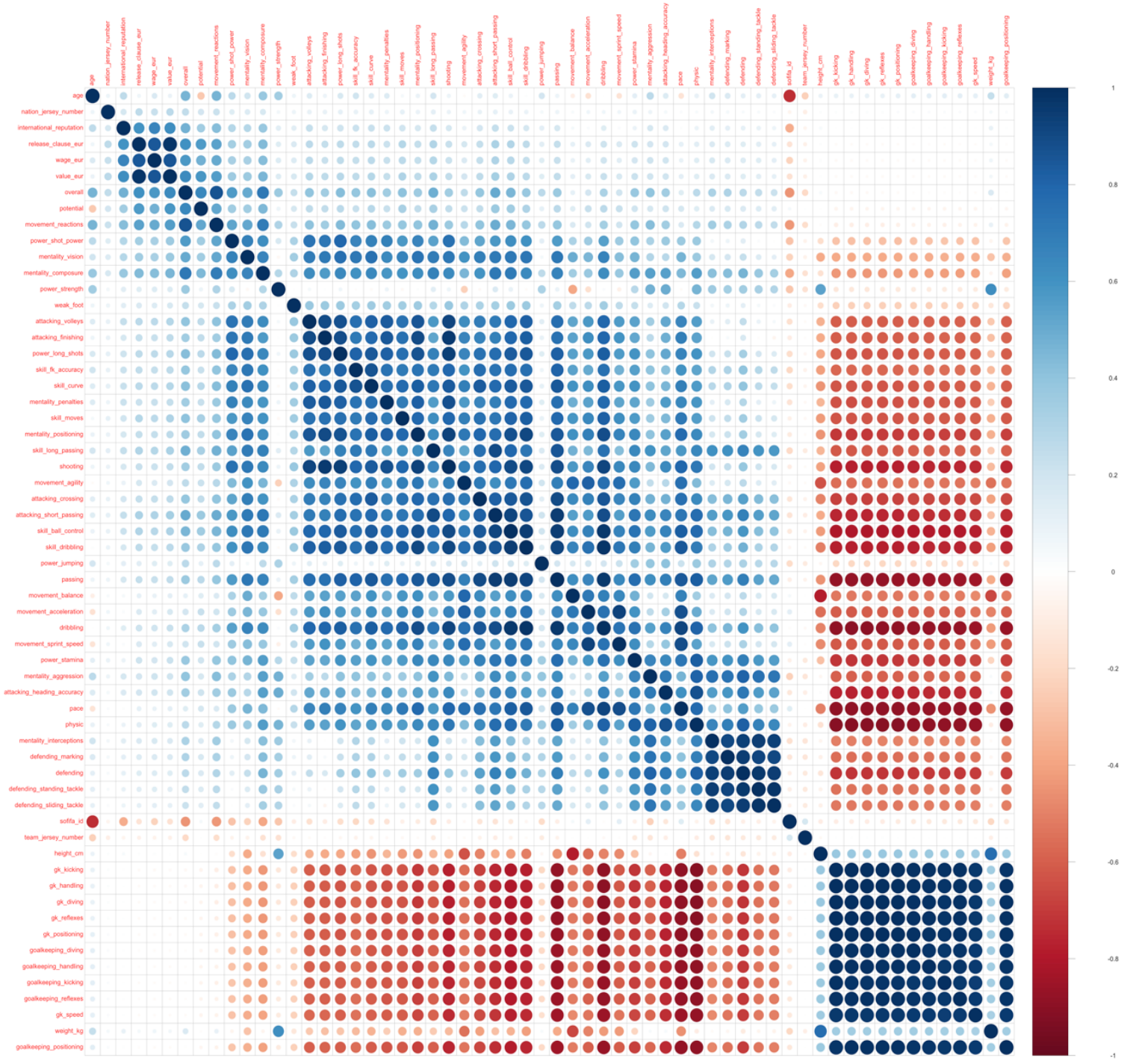


Figure 13

Figure 13 is a heat map of the correlation between all numeric variables in the dataset and offers many interesting insights. It is clear that all goalkeeper attributes are fully negatively associated with non-goalkeeper attributes, and vice versa as outfield players do not have goalkeeper attributes and the opposite also holds. Interestingly, movement reaction and mentality composure have the strongest positive correlations with overall rating, with a correlation coefficient of 0.8 and 0.7 respectively. This suggests that these two attributes are by far the most important in how good overall a footballer is, regardless of position. Strength

is positively correlated with height and to a greater extent weight at a coefficient of 0.8. However, height and weight are both quite strongly negatively correlated with a multitude of attacking attributes, such as balance and agility (coefficients of -0.9 and 0.7 respectively), and to a lesser extent, dribbling, sprint speed and passing. This suggests taller and/or heavier players are more suited to defence or goalkeeping. Pace and dribbling are also very strongly positively correlated with a coefficient of 0.9, perhaps reflecting that most players who are fast or good at dribbling play as attackers, which is the only position that requires both.

Random Forest

Ensemble learning is a form of supervised learning which generates multiple models on a training dataset, and then averages them to create a good model which is does not suffer from overfitting. Random Forest is a type of ensemble learning which improved the performance of decision trees by averaging them to reduce the variance, hence avoiding model overfit. The number of variables randomly selected at each fit was 20 when fitting random forest to the numerical variables in the Fifa20 dataset. Figure 14 clearly shows the power of random

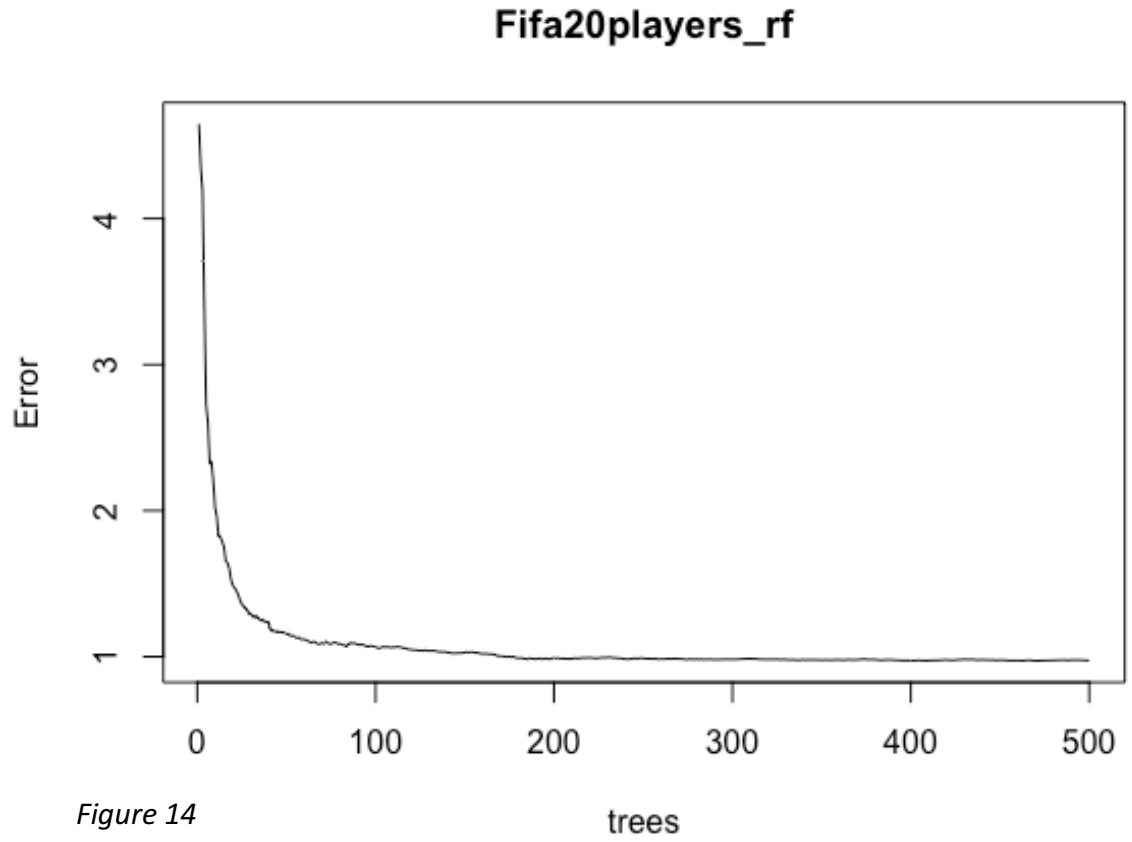


Figure 14

forest, as the error falls as more trees are added and averaged. Finally, trying all numerical variables in the dataset to be randomly selected at each split, yields fig 12. Figure 15 shows the out of bag error and test error, against the mean squared error. The test error and out of bag error appear highly correlated, and the MSE seems to be minimised where the number of variables at each split is 56.

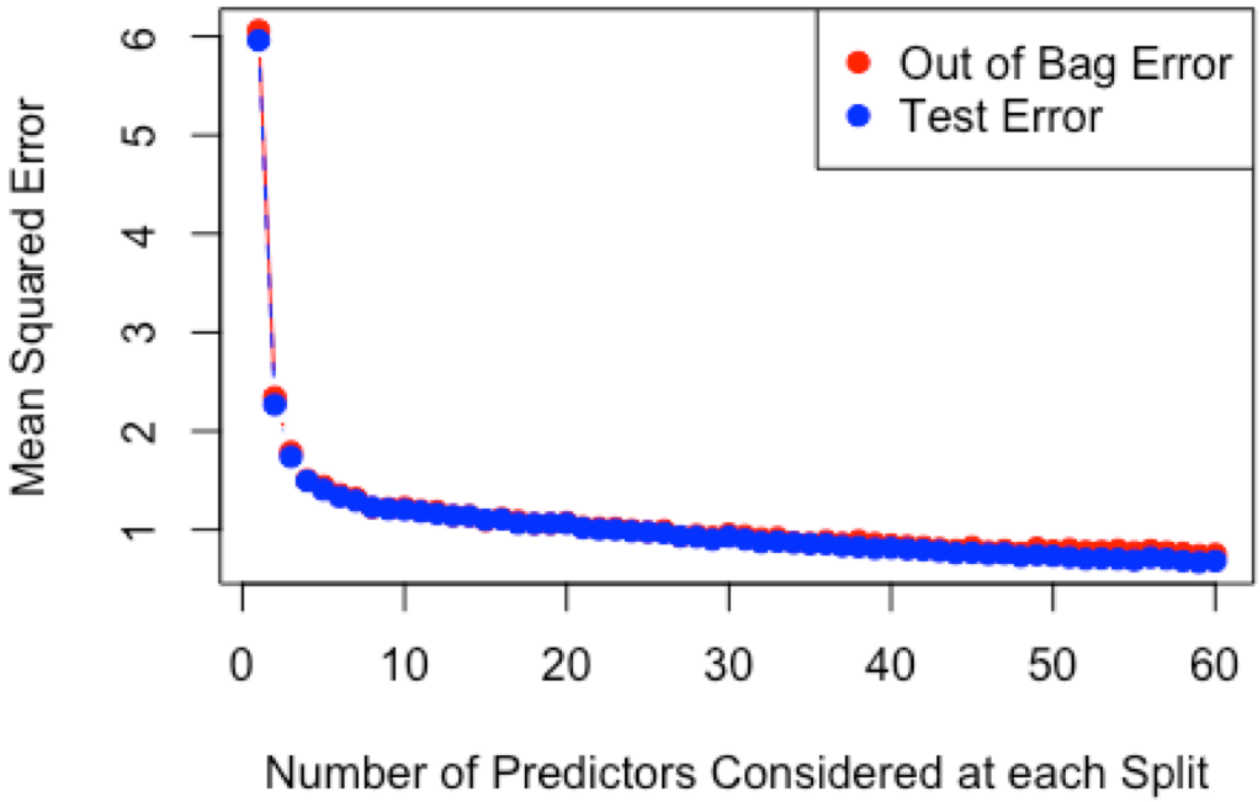


Figure 15

Limitations

In terms of limitations of the data, it would have been very insightful to have been able to produce a plot of how many players play outside their own country and where. However, this dataset did not contain country codes for club unfortunately, so this was not possible. A minor aesthetic issue was also that names of clubs and players that contained accents, were not printed when using ggplot, which meant that several of these would be left off of plots all together. Also, Fifa data is continually being updated every week, so the latest version would be more accurate to use. In hindsight it would have been sensible to try web-scraping the latest data, as it also contains the country codes which would have allowed a very helpful visualisation. The visualisation of the number and location of where players play in relation to their home country, would have been helpful also in seeing which domestic league is biggest and most attractive, and seeing the nationality breakdown within. The data also did not contain information on several foreign leagues, which may have been interesting to analyse. This dataset was very interesting for myself which led me to starting with analysis straight away. However, next time it may be more prudent to come up with a focused and specific question first, before beginning analyses. It would also have been interesting to analyse the link between positional group and wage.

Conclusion

Overall, many interesting findings have been gleaned by analysing this dataset. Firstly, football players generally have a BMI of below 25, although there are some very large outliers. The vast majority of players are from the UK, followed by Germany, Spain, France

and to a lesser extent, Brazil and Argentina; a surprisingly high number are from China and Saudi Arabia. Midfielders are overwhelmingly the most common positional group, but within midfield there are also by far the most positions available. Surprisingly, averaging attributes by position group reveals that forwards are the slowest and the most physical and defenders are marginally the fastest position group. Club value based off of player value indicates that Real Madrid, Barcelona and Manchester City are the leaders, and the top clubs by average potential is nearly identical. This suggests player potential is strongly correlated to player value. Overall ratings against player value and wage are found to have an exponential pattern, with exceptionally high wages and values corresponding to only the very best players. The heatmap shows that movement reaction and mentality composure have by far the strongest positive correlation with overall rating. Finally, the random forest modelling indicates that the mean squared error is minimised where the number of variables randomly selected at each split is 56.

Daanish Ahsan
Working with Data MTHM501
Candidate Number: 124070

References

All in Website form

2019
https://www.bbc.co.uk/sport/football/50634076

https://www.footballhistory.org/

Panja, Tariq, 2017
https://www.bloomberg.com/news/features/2017-07-13/soccer-balls-and-china-s-billions

Settimi, Christina 2019,
https://www.forbes.com/sites/christinasettimi/2019/06/18/the-worlds-highest-paid-soccer-players-2019-messi-ronaldo-and-neymar-dominate-the-sporting-world/#6ec813ab55ef

Dash, Kirankumar 2019,
https://www.sportskeeda.com/football/transfer-values-calculation-explained

https://www.firsttouchonline.com/the-most-notorious-shirt-numbers-in-football-history/

Daanish Ahsan
Working with Data MTHM501
Candidate Number: 124070

R Code Appendix

```
knitr::opts_chunk$set(include = FALSE)
#Working w data project

#This is an analysis of Fifa20 dataset

#Loading packages

library(arm)

library(norm)
library(tidyverse)

library(caTools)
library(randomForest

library(spdep)

library(sf)
library(CARBayes)

library(rgdal)

library(rgeos)

library(RColorBrewer)
library(ggplot2)
library(tidyverse)
library(broom)
library(cclust)
library(GGally)

library(LearnBayes)
library(memisc)

library(MASS)
library(sjstats)

library(sjPlot)

library(tidyr)
library(caTools)
library(dplyr)
library(raster)

library(tidyverse)
library(rvest)

library(magrittr)

library(ggmap)

library(stringr)
library(maps)
```

```r
library(mice)

library(missForest)

library(VIM)

library(rnaturalearth)
library(rnaturalearthdata)
library(viridis)

library(plyr)

library(corrplot)

library(ggcorrplot)
library(scico)
library(formattable)

library(kableExtra)

#Reading in the datasets

Fifa20players <- read.csv('fifa20data.csv')

set.seed(500)
options(scipen = 999999)

#Some summary statistics

summary(Fifa20players$release_clause_eur)

##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.      NA's
##     13000    563000   1200000   4740717   3700000 195800000      1298

summary(Fifa20players$overall)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   48.00   62.00   66.00   66.24   71.00   94.00

summary(Fifa20players$potential)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   49.00   67.00   71.00   71.55   75.00   95.00

summary(Fifa20players$age)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   16.00   22.00   25.00   25.28   29.00   42.00

summary(Fifa20players$height_cm)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   156.0   177.0   181.0   181.4   186.0   205.0

summary(Fifa20players$weight_kg)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   50.00   70.00   75.00   75.28   80.00  110.00

#removing uneeded columns that are not helpful in showing relationships to
player attributes

Fifa20players <- subset(Fifa20players, select = -c(loaned_from,long_name,
player_url, real_face, nation_position, contract_valid_until))

#Making NA values 0

Fifa20players[is.na(Fifa20players)] <- 0

#Creating data frames of variables for required plots

#Handling Missing Data with Mice

#Removing 10% of values at random from height and weight columns
Fifa20missingweightandheight <- prodNA(Fifa20players[,5:6], noNA = 0.1)

#Using mice to identify pattern of missing data
md.pattern(Fifa20missingweightandheight)

#Visual representation of missing data
#Comment on it
vis_plot_missing <- aggr(Fifa20missingweightandheight, col=c('navyblue','r
ed'), numbers=TRUE, sortVars=TRUE, labels=names(data), cex.axis=0.5, cex.n
umbers=0.5, gap=3, ylab=c("Histogram of missing data","Pattern"))

#Using random forests to impute values
imputeData <- mice(Fifa20missingweightandheight,m=1,maxit=50,meth='rf',see
d=500)

imputeData$imp$height_cm

imputeData$imp$weight_kg

#Constructing completed dataset w imputed values
completefifa20data <- complete(imputeData)

#comment on graphs and the colour
stripplot(imputeData, pch = 20, cex = 1.2)

#MAP
#MAP of world by number of players in each country

#Wrangling data of players by nation

nation <- table(Fifa20players$nationality)

nation <- as.data.frame(nation)

names(nation)[1]<-"region"
```

```r
#Removing England, Scotland and NI rows
nation <- nation[-c(46, 114, 130),]

#Need to combine NI, England, Scotland into UK

#UK = 2025

#Adding a UK row
UK <- data.frame(region='UK',Freq='2025')
nation <- rbind(nation, UK)

#Loading in world data
world0 <- map_data("world")

#Seeing if country names in world map and my data frame match
matchcountries <- nation$region %in%
  world0$region

nation <- add_column(nation, matchcountries)

#Renaming countries to match those in the worldmap dataset
nation$region <- as.factor(nation$region)

nation$region <- plyr::revalue(nation$region, c('United States'='USA', 'An
tigua & Barbuda'='Antigua', 'Bosnia Herzegovina'='Bosnia and Herzegovina',
                                                'Central African Rep.'='Ce
ntral African Republic', 'China PR'='China',
                                                'Congo'='Republic of Congo
', 'DR Congo'='Democratic Republic of the Congo', 'FYR Macedonia'='Macedon
ia',
                                                'Guinea Bissau'='Guinea',
'Korea DPR'='North Korea', 'Korea Republic'='South Korea',
                                                'Republic of Ireland'='Ire
land', 'St Kitts Nevis', 'St Lucia', 'Wales', 'Trinidad & Tobago'='Trinida
d'))

str(nation)

str(world0)

#Getting rid of match countries column
nation$matchcountries<- NULL
nation

#Creating a new dataframe with same numbers of rows as regions in world ma
p dataset
tojoin <- as.data.frame(matrix(
  nrow = length(table(world0$region)),
  ncol = 2,
  NA,
  dimnames = list(names(table(world0$region)), colnames(nation))
))
tojoin$region <- rownames(tojoin)
```

```
#Trying to get compatible types for nation
nation <- data.frame(lapply(nation, as.character), stringsAsFactors=FALSE)

tojoin <- data.frame(lapply(tojoin, as.character), stringsAsFactors=FALSE)

#Joining my dataframe with worldmap data
all <- full_join(nation, tojoin)

all <- all[order(all$region), ]

mapbig <- inner_join(world0, all, by = "region")

str(mapbig)

mapbig <- data.frame(lapply(mapbig, as.numeric), stringsAsFactors=FALSE)

str(mapbig)

#Map of background for map
worldmapempty <- ggplot() + theme(
  panel.background = element_rect(fill = "lightcyan1",
                                  color = NA),
  panel.grid = element_blank(),
  axis.text.x = element_blank(),
  axis.text.y = element_blank(),
  axis.ticks = element_blank(),
  axis.title.x = element_blank(),
  axis.title.y = element_blank()
)

worldmapempty

str(worldmapempty)

#Plotting map
Freqmap <- worldmapempty +  geom_polygon(data = mapbig,
                            aes(fill = Freq,
                            x = long,
                            y = lat,
                            group = group),
                            color = "grey70") +
                            labs(title = "Nationality of Fifa 20 Players "
,
                            caption = "Data from Kaggle, map from GGmaps")
+
                            theme(text = element_text(size = 30),
                            plot.title = element_text(face = "bold")) + sc
ale_fill_viridis(option = "plasma",
                            direction = -1,

breaks = c(0, 500, 1000, 1500, 2000, 2500), limits= c(0, 2500),
                            name = "",
                            na.value = "grey80",
                            guide = guide_colorbar(
                            barheight = unit(140, units = "mm"),
```

```
                              barwidth = unit(6, units = "mm")))

Freqmap

#GRAPH section

#first plot
#number of players by age group

#18,278 players in total.

AGE <- Fifa20players$age

summary(AGE)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   16.00   22.00   25.00   25.28   29.00   42.00

hist(AGE, breaks = 10, main = 'Histogram of player ages', freq = TRUE, col
= 'LIGHTBLUE', ylim = c(0, 3000))

#Average age of 25 years and 4 months
#majority of players between 19 and 30 years old, reflecting the average p
layer career
#much fewer players between 15 and 18, and 33 and above
#reflects that young players feature less in general, and that playing car
eers generally do not last past early 30s

#Players by positional group
#adding a column for position groups
x <- as.factor(Fifa20players$team_position)
levels(x) <- list(GK = c("GK"), DEF=c("LWB", "LB", "CB", "RB", "RWB"), MID
= c("LW", "LM", "CDM", 'LDM', 'RDM', "CM", "CAM", "RM", "RW"), FWD = c("CF
", "ST", 'LF', "RF"))

Fifa20playerspositions <- mutate(Fifa20players, PositionGroup = x)

head(Fifa20playerspositions)

grouppos <- as.data.frame(table(Fifa20playerspositions$PositionGroup))

ggplot(grouppos, aes(x=Var1, y=Freq)) + geom_bar(stat='identity',aes(fill
= Freq)) + labs(title="Number of players by position group", x='Position G
roup', y='Number') +
theme(legend.position = 'none')

#Team position data
pos <- as.data.frame(table(Fifa20players$team_position))

#Excluding subs, reserves and no position

pos <- pos[-c(1, 30, 23),]

pos
```

```
#Bar plot by every position
ggplot(pos, aes(x=Var1, y=Freq, col='red')) + geom_bar(stat = "identity")
+ labs(title = 'Number of players by club position', x='Position' ,y='Numb
er') +
  theme(legend.position = 'none')

#Top 10 most valuable clubs based on total Player value in Eur

valueclub <- aggregate(Fifa20players$value_eur, by=list(club=Fifa20players
$club), FUN=sum)

top10valueclubs <- top_n(valueclub, 10, x)

top10valueclubs$club <-as.factor(top10valueclubs$club)

ggplot(top10valueclubs, aes(x = club, y = x/1000000, col='RED')) + geom_ba
r(stat = "identity", aes(fill=club)) + coord_flip() + ggtitle("Top 10 valu
able clubs") +
  labs(y='Club Value in Millions (Euros)', x='Club') +theme(legend.positio
n = 'none')

#Top 10 clubs based on average player potential

potclub <- aggregate(Fifa20players$potential, by=list(club=Fifa20players$c
lub), FUN=mean)

top10potclubs <- top_n(potclub, 10, x)

top10potclubs <- rename(top10potclubs,c('x'='Average Player Potential'))

kable(top10potclubs, 'html') %>%
  cat(., file = 'top10potclubs.html') #table

#Top clubs based on average rating

clubsoverall <- aggregate(Fifa20players$overall, by=list(club=Fifa20player
s$club), FUN=mean)

top10clubsoverall <- top_n(clubsoverall, 10, x)

top10clubsoverall <- rename(top10clubsoverall,c('x'='Average Player Overal
l'))

kable(top10clubsoverall, 'html') %>%
  cat(., file = 'top10clubsoverall.html') #table

#Comparing top players by overall to top players by potential who aren't a
lready in top 10 overall
#which potential players could overtake the top overall ones

top10playersrating <- top_n(Fifa20players, n=10, overall)

top10playersrating <- top10playersrating[,c('short_name', 'overall')]
```

```
kable(top10playersrating, 'html') %>%
  cat(., file = 'top10playersrating.html')#table

toppotential <- top_n(Fifa20players, n=10, potential)

#removing those who are already in top 10, from top potential

toppotential <- toppotential[-c(1, 2, 3, 4, 5),]

toppotential <- toppotential[,c("short_name",'potential')]

kable(toppotential, 'html') %>%
  cat(., file = 'toppotential.html')#table

#K.Mbappe, L.Sane, Joao Felix, Vinicius Jr
#Mbappe could potentially become greatest player of all time according to
potential

#Value vs overall dot plot

valuevsoverall <- Fifa20players[,c('value_eur', 'overall')]

ggplot(valuevsoverall,aes(x=value_eur/1000000, y=overall, col='RED')) +
  geom_point() + labs(title='Player Rating against Value', y='Overall Rati
ng', x='Player Value in Millions (Euros)') +theme(legend.position = 'none'
)

#Clear that as rating goes above 85, value increases exponentially due to
fewer players at higher ratings, so they are highly valued

#Wages vs overall rating

wagevsoverall <- Fifa20players[,c('wage_eur', 'overall')]

ggplot(wagevsoverall,aes(x=wage_eur/100000, y=overall, color='red')) +
  geom_point() + labs(title='Player Rating against Wage ', y='Overall Rati
ng', x='Weekly Player Wage in Hundred Thousands (Euros)') +theme(legend.po
sition = 'none')

#Clear from rating against wage that as rating exceeds 85, wages increase
exponentially.

#Both of the above diagrams show that rating is strongly correlated w/ age
and value

#Age against wage

agevswage <- Fifa20players[,c('age', 'wage_eur')]

ggplot(agevswage,aes(x=wage_eur/100000, y=age, colour='lightblue')) +
  geom_point() + labs(title='Player Age against Wage ', y='Age', x='Weekly
Player Wage in Hundred Thousands (Euros)') +theme(legend.position = 'none'
)
```

```
#team jersey number and rating correlated?

#Looking at all players first
jerseynovsoverall <- Fifa20players[,c('team_jersey_number', 'overall')]

ggplot(jerseynovsoverall,aes(x=team_jersey_number, y=overall, colour='ligh
tblue')) +
  geom_point() + labs(title='Player Shirt Number against Overall Rating',
y='Overall', x='Player Shirt Number') +theme(legend.position = 'none')

#Looking at the top players a bit more closely to examine their kit number
s

jerseynovsoveralltop200 <- jerseynovsoverall[c(1:200),]

ggplot(jerseynovsoveralltop200,aes(x=team_jersey_number, y=overall, colour
='lightblue')) +
  geom_point() + labs(title='Player Shirt Number against Overall Rating',
y='Overall', x='Player Shirt Number') +theme(legend.position = 'none')

#We can see that the top 200 players generally have shirt numbers below 35
, with a few exceptions- these may be keepers or subs or reserves
#Or very young rotational squad players
#Looking at the top players in the world, about 88 ranking and above
#clear that kit numbers become lower numbers- reflects starting keeper num
bers
#Number 10, 7 and 9 are highly sought after numbers that the best attacker
s wear
#indeed the top rated players are Messi, Ronaldo, Neymar etc- Messi and Ro
naldo can be seen on this chart
#as the two points that are the only two to stand above a rating of 92.5.

#BMI

#Converting cm to metres
Fifa20metresheight <- Fifa20players[,5]/100

Fifa20playerswheightinm <- add_column(Fifa20players, Fifa20metresheight)

Fifa20playerswheightinm$Fifa20metresheight



Fifa20BMI <- mutate(Fifa20playerswheightinm, BMI= Fifa20playerswheightinm$
weight/Fifa20playerswheightinm$Fifa20metresheight^2)

Fifa20BMIfinal <- Fifa20BMI[,c('BMI', 'overall')]

#BMI and overall rating link
ggplot(Fifa20BMIfinal,aes(x=BMI, y=overall, colour='lightblue')) +
  geom_point() + labs(title='Actual BMI against Overall Rating', y='Overal
l', x='BMI') +theme(legend.position = 'none')
```

```
#Clear that majority of players are in healthy range
#Big outlier is Akinfenwa
#The very highest rated players have a BMI very close to 25
#The very highest rated player- Messi has BMI of virtually 25- but BMI doe
sn't take into account muscle properly
#But even outliers or underweight or overweight by BMI is no big deal
#as BMI is not a great measure of fitness- weight could be due to muscle
#underweight could be due to tall height.

#Using Random Forest imputed values for BMI and rating link

completemetresfifa <- completefifa20data$height_cm/100

completefifa20metres <- add_column(completefifa20data, completemetresfifa)

Fifa20BMIimpute <- mutate(Fifa20BMIfinal, imputeBMI= completefifa20metres$
weight_kg/completefifa20metres$completemetresfifa^2)

FifaimputedBMI <- Fifa20BMIimpute[,c('imputeBMI', 'overall')]

ggplot(FifaimputedBMI,aes(x=imputeBMI, y=overall, colour='lightblue')) +
  geom_point(aes()) + labs(title='Imputed BMI against Overall Rating', y='
Overall', x='Imputed BMI') +theme(legend.position = 'none')

#Imputed BMI has more spread as height and weight have been predicted usin
g random forest
#and are not actual results.
#Also a few more outliers upper and lower ends of BMI

#Clubs by average BMI- helps avoid problem of number of players at clubs

BMIclub <- aggregate(Fifa20BMI$BMI, by=list(club=Fifa20BMI$club), FUN=mean
)

BMIclub <- rename(BMIclub,c('x'='BMI'))

top10BMIclubs <- top_n(BMIclub, 10, BMI)

kable(top10BMIclubs, 'html') %>%
  cat(., file = 'top10BMIclubs.html')#table

lowest10BMI <- top_n(BMIclub, 10, -BMI)

kable(lowest10BMI, 'html') %>%
  cat(., file = 'lowest10BMIpls.html')#table

#BMIs dont differ hugely between top 10 and lowest 10
#Reflects that most football players have to be fit- no avg is 25 or great
er
#which is considered overweight

#Age distribution in the top 10 most valuable clubs
```

```
ageclubs <- aggregate(Fifa20players$age, by=list(club=Fifa20players$club),
FUN=mean)

ageclubs <- add_column(ageclubs, valueclub$x)

ageclubstop10 <- top_n(ageclubs, 10, valueclub$x)

ageclubstop10 <- rename(ageclubstop10,c('x'='Average_age'))
ageclubstop10 <- rename(ageclubstop10,c('valueclub$x'='Club Value'))

kable(ageclubstop10, 'html') %>%
  cat(., file =ageclubstop10.html')#table

  ggplot(ageclubstop10, aes(x = club, y = Average_age, fill=club)) +
    geom_bar(stat = 'identity') +
    theme(axis.text.x = element_text(angle = 90), legend.position = "none"
) +
    ylim(0, 40) +
  labs(title="Average Age at top 10 clubs", y="Average Age", x='Club')

#Juventus have oldest squad- known for buying veteran players for cheap
#their average squad age is 27
#Youngest average squad age is Borussia Dortmund's at 23 years and 8 month
s- known for their academy
#and letting young players play regularly

#Attributes by position
e <- aggregate(Fifa20playerspositions[,26:33], by=list(PositionGroup=Fifa2
0playerspositions$PositionGroup), FUN = mean)

e

#Multibar chart of attributes by position using mean of each attribute acr
oss position groups

#Melting data to reshape it
data.e <- melt(e, id.vars='PositionGroup')

#Plot
ggplot(data.e, aes(PositionGroup, value)) +
  geom_bar(aes(fill = variable), position = "dodge", stat="identity") +
  labs(title = 'Attributes by Position', x= 'Position Group', y= 'Value')
+
  theme(panel.background = element_rect(fill = 'black'))

##Correlation of heatmap of attributes and overall and value and potential

Corr <- Fifa20players

#Checking which columns are numeric
sapply(Corr, is.numeric)

#Dropping all non-numeric columns
Corr <- Corr[, sapply(Corr, is.numeric)]
```

```
#Note that NA values need to be removed here for this cor to work
yay <- cor(Corr, use = "complete.obs", method = "pearson")

bfg <- corrplot(yay, method = 'circle')

#Printing the corrplot to scale and saving the file
col4 <- scico(100, palette = 'vik')
filetag <- "bfgpls.png"

png(filetag, height = 4500, width = 4500)

corrplot(yay, order = "AOE", upper = "ellipse", lower = "number",
               upper.col = col4, lower.col = col4,
               tl.cex = 2, cl.cex = 2, number.cex = 2)

dev.off()

#AOE is angular order of eigenvectors

#Modelling
#Random Forest w/ just numerical values used in heatmap

dim(Fifa20players)

#Seperating training and test set
#Training Sample with 1300 observations
trainFifa20 <- sample(1:nrow(Fifa20players),1300)

Fifa20players_rf <- randomForest(overall~., data = Corr, subset = trainFif
a20)

#Number of variables randomly selected at each split is 20.

plot(Fifa20players_rf)

#Clear from plot that error falls as more trees are added and averaged.

dim(Corr)

oob.err <- double(60)
test.err <- double(60)

#mtry is no of Variables randomly chosen at each split. #We are growing 40
0 trees for all 60 predictors
for(mtry in 1:60)
{
  rfy <- randomForest(overall ~ . , data = Corr , subset = trainFifa20,mtr
y=mtry,ntree=400)
  oob.err[mtry] <- rfy$mse[400] #Error of all Trees fitted

  pred<-predict(rfy,Fifa20players[-trainFifa20,]) #Predictions on Test Set
for each Tree
  test.err[mtry]= with(Fifa20players[-trainFifa20,], mean( (overall - pred
```

```
)^2)) #Mean Squared Test Error

  cat(mtry," ") #printing the output to the console

matplot(1:mtry , cbind(oob.err,test.err), pch=19 , col=c("red","blue"),type="b",ylab="Mean Squared Error",xlab="Number of Predictors Considered at each Split")
legend("topright",legend=c("Out of Bag Error","Test Error"),pch=19, col=c("red","blue"))
```