

PSY6422_Project

230170379

2024

Contents

Introduction	2
Background	2
Requirements	2
Importing the Data	2
Data Origins	3
Research Questions	4
Plot One: How have UK stock values changed over time?	5
Data Preperation	5
Creation	5
The Visualisation	6
Discussion	7
Plot Two: How do the four UK national football teams differ?	8
Data Preperation	8
Creation	11
The Visualisation	12
Discussion	13
Plot 3: Can national football results really have an impact on UK stock market values?	14
Data Preperation	14
Creation	16
The Visualisation	18
Discussion	19
Summary	19

Introduction

Background

Edmans et. al., 2007 performed an analysis on the influence of international sporting results on countries' stock returns, finding that **national team losses predicted lower stock returns**. The researchers hypothesized that this effect was due to decreased investor mood.

Here's a link to their study if you're interested...

https://onlinelibrary.wiley.com/doi/epdf/10.1111/j.1540-6261.2007.01262.x?saml_referrer

I would like to perform a similar analysis of each UK country's national football (soccer) results and the returns of FTSE 100 stocks, beginning in 2002 after the start of the Japan World Cup.

Requirements

The code used loads of functions, and therefore required many libraries!

These included: `library(tidyverse)`, `library(here)`, `library(tidyr)`, `library(ggplot2)`, `library(dplyr)`, `library(plotly)`, `library(gganimate)`, `library(gifski)`, `library(forcats)`, `library(ggtext)`, `library(scales)`, `library(prismatic)`,

Importing the Data

I used the `here` function to import the data, allowing this code to run on any computer which had downloaded the "projectfolder" folder

```
#Import Datasets
ftse100pathway <- here("projectfolder", "data", "FTSE100_Data_2002_2024.csv")
ftse100 <- read.csv(ftse100pathway)

resultspathway <- here("projectfolder", "data", "results.csv")
results <- read.csv(resultspathway)

shootoutspathway <-
  here("projectfolder", "data", "shootouts.csv")
shootouts <- read.csv(shootoutspathway)
```

Data Origins

We first needed a record of UK stock data...

The Financial Times Stock Exchange 100 Index (ftse100) is a stock market formed from a collective of 100 of the UK's most valuable companies. It is renowned as a solid reflection of UK stocks and UK investor behaviours in general.

The ftse100 dataset contains a huge collection of daily stock returns, showing the date of data collection (*Date*) as well as the highest (*High*), lowest (*Low*), and average (*Price*) stock values of that day. The dataset also presents the percentage at which stocks changed from the day before (*Change..*).

This data was sourced from: <https://uk.investing.com/indices/uk-100-historical-data> Investing.com is a financial website which provides free real-time stock data across 250 exchanges worldwide. They employ over 250 people to publish trustworthy data each day. <https://uk.investing.com/about-us/>

```
head(ftse100)
```

```
##           Date   Price   Open   High   Low   Vol. Change..
## 1 21/02/2024 7,662.51 7,719.21 7,719.21 7,642.75 1.19B -0.73%
## 2 20/02/2024 7,719.21 7,728.50 7,748.73 7,705.98 819.03M -0.12%
## 3 19/02/2024 7,728.50 7,711.71 7,733.54 7,692.48 575.04M 0.22%
## 4 16/02/2024 7,711.71 7,597.53 7,720.72 7,597.53 996.23M 1.50%
## 5 15/02/2024 7,597.53 7,568.40 7,612.34 7,562.10 651.78M 0.38%
## 6 14/02/2024 7,568.40 7,512.28 7,590.13 7,512.28 1.14B 0.75%
```

Next, we needed a record of international football matches since 2002...

The results dataset contains all international football results since 1872, detailing the date of each match (*date*), the current names of countries that played (*home_team* and *away_team*), the number of goals scored by each team (*home_score* and *away_score*), and the competition that each game was played in (*tournament*). Information was also provided regarding where the match was played and whether this was a neutral location or not (*neutral*).

This data was sourced from: <https://www.kaggle.com/datasets/martj42/international-football-results-from-1872-to-2017>

Unlike what the link itself suggests, this kaggle user collected data from international football matches starting from the very first official match in 1872 up to 2024. The dataset contains only men's first team matches, and does not include matches from the Olympic Games. In the words of the user: "The data is gathered from several sources including but not limited to Wikipedia, rsssf.com, and individual football associations' websites."

```
head(results)
```

```
##           date home_team away_team home_score away_score tournament   city
## 1 1872-11-30  Scotland   England         0         0  Friendly Glasgow
## 2 1873-03-08   England  Scotland         4         2  Friendly  London
## 3 1874-03-07  Scotland   England         2         1  Friendly Glasgow
## 4 1875-03-06   England  Scotland         2         2  Friendly  London
## 5 1876-03-04  Scotland   England         3         0  Friendly Glasgow
## 6 1876-03-25  Scotland    Wales         4         0  Friendly Glasgow
##      country neutral
```

```
## 1 Scotland FALSE
## 2 England FALSE
## 3 Scotland FALSE
## 4 England FALSE
## 5 Scotland FALSE
## 6 Scotland FALSE
```

Finally, we needed data for any football matches that went to penalty shootouts...

```
head(shootouts)
```

```
##      date   home_team   away_team   winner first_shooter
## 1 22/08/1967      India      Taiwan      Taiwan
## 2 14/11/1971 South Korea Vietnam Republic South Korea
## 3 07/05/1972 South Korea      Iraq      Iraq
## 4 17/05/1972  Thailand  South Korea South Korea
## 5 19/05/1972  Thailand    Cambodia  Thailand
## 6 21/04/1973   Senegal    Ghana      Ghana
```

The shootouts dataset contains data for every international penalty shootout since 1967. Once again, there is information showing the two teams that played (*home_team* and *away_team*) and the date of each shootout (*date*). Each row contains the winner of the shootout (*winner*), and some rows even show which team took the first penalty in the shootout (*first_shooter*).

This data was also sourced from: <https://www.kaggle.com/datasets/martj42/international-football-results-from-1872-to-2017>

Research Questions

Three research questions were created in this research project.

1. *How have UK stock values changed over time?*
2. *How do the four UK national football teams differ?*
3. *Can national football results really have an impact on UK stock market values?*

Three visualisations were created in attempt to answer these questions.

Plot One: How have UK stock values changed over time?

I wanted to create a line graph showing how daily ftse100 stock values have increased/decreased since January 2002. I then planned to animate the graph to better visualise how the line moves through time.

This Plot only required data from the ftse100 dataset

Data Preperation

Before creating the plot, I needed to ensure that my variables of interest were in the correct format.

```
#Convert date values to a date variable on R
ftse100$Date <- as.Date(ftse100$Date, format = "%d/%m/%Y")

#Convert Price to a numerical variable
ftse100$Price <- as.numeric(gsub(",", "", ftse100$Price)) #Remove confusing commas
ftse100$Price <- as.numeric(ftse100$Price)
```

Creation

Before creating an animated plot, I needed to create one which wasn't animated

```
#Create the initial (not animated) plot
#Line graph
stockplot <- ggplot(ftse100,
  aes(x = Date, y = Price, color = Price)) +
  geom_line() +

#Add heading and axis titles
labs(title = "UK Stocks Since 2002",
  subtitle = paste("Change in FTSE100 Daily Stock Values from",
    format(min(ftse100$Date), "%B %Y"),
    "to",
    format(max(ftse100$Date), "%B %Y")),
  x = "Year", y = "UK Stock Price (£)") +

#Choose Theme
theme_classic() +

#Add Colour Scale
colour_scale +

#Determine X axis Labels every 2 years
scale_x_date(date_breaks = "2 years", date_labels = "%Y") +

#Add vertical dashed lines coming from the X axis labels
geom_vline(xintercept = as.numeric(seq(min(ftse100$Date), max(ftse100$Date), by = "2 years")),
  color = "gray", linetype = "dashed", alpha = 0.5) +

#Add data source below plot
annotate("text", x = as.Date("2014-01-01"), y = min(ftse100$Price),
```

```

    label = "Data Source: https://uk.investing.com", hjust = 0, vjust = -1,
    color = "darkgray", size = 3) +

#Colour code, add fonts, determine size of writing
theme(
  panel.background = element_rect(fill = "#fbf9f4", color = "#fbf9f4"),
  plot.background = element_rect(fill = "#fbf9f4", color = "#fbf9f4"),
  panel.border = element_blank(),
  panel.grid.major.x = element_blank(),
  axis.title.x = element_text(color = "#1b2860"),
  axis.title.y = element_text(color = "#1b2860"),
  plot.title = element_text(
    family = "sans",
    size = 20,
    face = "bold",
    color = "#1b2860"),
  plot.title.position = "plot", # slightly different from default
  axis.text = element_text(
    family = "sans",
    size = 10,
    color = "#1b2860"),
  axis.title = element_text(
    family = "sans",
    size = 12))

```

Only then could I animate the line using the `transition_reveal` function

```

stocks_animated <- stockplot + transition_reveal(Date) +
  enter_fade() +
  exit_fade()

```

The Visualisation

Animated gifs are unfortunately not supported by all PDF outputs. Here is a visualisation of the linegraph before animation.

```
print(stockplot)
```

UK Stocks Since 2002

Change in FTSE100 Daily Stock Values from January 2002 to February 2024



Discussion

This is a line graph showing the ups and downs of ftse100 stock prices since the 1st of January, 2002. Higher stock values are coloured in a positive green, whilst drops in values are coloured in a negative red. Dashed vertical lines also give reference for the 2-year breaks labelled on the x axis.

There is a general increase in stock values, but this has been paired with some harsh dips. One dip that is interesting to note occurs just after the start of the year 2020. The COVID pandemic really affected investor behaviour!

Plot Two: How do the four UK national football teams differ?

For my next plot, I wanted to create a dumbbell chart showing how each UK nation's football team differ in their individual results.

Data Preperation

Before beginning the second (and third!) analyses, I needed to merge the stocks, results, and shootouts datasets. Merging the results and shootouts datasets was necessary for the second plot as i wanted to present a statistical visualisation of how each UK nation has performed in all of their football matches.

I first ensured that the values from all date variables were characterised as dates, and merged the datasets using the dates as well as the teams which played for the results and shootouts datasets.

```
#Convert all date values to a date variable on R
```

```
ftse100$Date <-  
  as.Date(ftse100$Date, format = "%d/%m/%Y")
```

```
results$date <-  
  as.Date(results$date, format = "%Y-%m-%d")
```

```
shootouts$date <-  
  as.Date(shootouts$date, format = "%d/%m/%Y")
```

```
#Merge Datasets using this Date variable
```

```
data <-  
  merge(ftse100, results,  
        by.x = "Date",  
        by.y = "date",  
        all = TRUE)  
data <-  
  merge(data, shootouts,  
        by.x = c("Date", "home_team", "away_team"),  
        by.y = c("date", "home_team", "away_team"),  
        all.x = TRUE)
```

The ftse100 dataset did not contain values for every day, so information from missing days was filled in using the next known values.

```
#Fill in missing Stock data
```

```
#Some days do not have stock data
```

```
#This code adds to missing info using the next known stock values
```

```
data <- data[order(data$Date), ]  
data <- data %>%  
  fill(Price, Open, High, Low, Vol., Change..., .direction = "down")
```

We then needed to create a new variable showing who won football matches. So far we only had the winners of games that went to penalties. We needed to add to the already created winner variable without changing its present values


```

#Determine which rows need filling in the winner variable
newwinners <- is.na(data$winner)

#If home_score > away_score, put the home team in winner variable
#If away_score > home_score, put the away team in winner variable
#If else (scores are even), put "draw"
data$winner[newwinners] <-

  ifelse(data$home_score[newwinners] > data$away_score[newwinners],
    data$home_team[newwinners],

    ifelse(data$home_score[newwinners] < data$away_score[newwinners],
      data$away_team[newwinners],

      "draw"))

```

Rows without any football matches allocated to them were then removed, and a sanity check was performed to see if this all went through ok.

```

#Remove Rows with no Football Match
data <- data[complete.cases(data$home_team), ]

#Sanity check
head(data)

```

```

##      Date home_team away_team Price Open High Low Vol. Change.. home_score
## 1 1872-11-30  Scotland   England    NA <NA> <NA> <NA> <NA>    <NA>         0
## 2 1873-03-08   England  Scotland    NA <NA> <NA> <NA> <NA>    <NA>         4
## 3 1874-03-07  Scotland   England    NA <NA> <NA> <NA> <NA>    <NA>         2
## 4 1875-03-06   England  Scotland    NA <NA> <NA> <NA> <NA>    <NA>         2
## 5 1876-03-04  Scotland   England    NA <NA> <NA> <NA> <NA>    <NA>         3
## 6 1876-03-25  Scotland    Wales    NA <NA> <NA> <NA> <NA>    <NA>         4
##  away_score tournament   city  country neutral  winner first_shooter
## 1          0   Friendly Glasgow Scotland  FALSE    draw    <NA>
## 2          2   Friendly London  England  FALSE  England    <NA>
## 3          1   Friendly Glasgow Scotland  FALSE Scotland    <NA>
## 4          2   Friendly London  England  FALSE    draw    <NA>
## 5          0   Friendly Glasgow Scotland  FALSE Scotland    <NA>
## 6          0   Friendly Glasgow Scotland  FALSE Scotland    <NA>

```

A new dataset specific for the creation of plot 2 was made, and all unneeded columns were removed.

```

#Create new dataset for plot 2
df_footy <- data

#Remove Unneeded Columns
df_footy <-
  subset(df_footy, select = -c(Price, Open, High, Low, Vol., Change.., city, country, first_shooter))

```

I wanted to see how nations had differed in their wins, losses, draws. I therefore created a new dataset showing these variables, called nationstats.

We first needed the stats of when each team was the home team, and then performed the same methods on when each team was the away team

```
nationstats <- df_footy %>%
  group_by(team = home_team) %>% #Create variable named "team" for each national football team
  summarise(
    games_played = n(), #Total games played by how many times they are in home_team
    games_won = sum(winner == home_team), #If team is in winner, they won the game
    games_lost = sum(winner != home_team & winner != "draw") #If team is not winner and the game was :
  ) %>%
  ungroup() %>% # Remove grouping
  bind_rows( #Bind rows to this dataset using the same process for away teams
    df_footy %>%
      group_by(team = away_team) %>%
      summarise(
        games_played = n(),
        games_won = sum(winner == away_team),
        games_lost = sum(winner != away_team & winner != "draw")
      ) %>%
    arrange(desc(games_played)) # Arrange by total games played, descending
```

Two rows for each team were created by this code. One showed the statistics for when a team was the home team, and the other showed statistics for when the team was the away team. The next job was to bind these rows together to show total matches/wins/losses

```
nationstats <- nationstats %>%
  group_by(team) %>%
  summarise(
    games_played_total = sum(games_played),
    games_won_total = sum(games_won),
    games_lost_total = sum(games_lost),
    .groups = "drop" # Ensure that grouped data is returned as a flat data frame
  )

#Check the new dataset looks ok
head(nationstats)
```

```
## # A tibble: 6 x 4
##   team          games_played_total games_won_total games_lost_total
##   <chr>                <int>          <int>          <int>
## 1 Abkhazia             28             14             8
## 2 Afghanistan         129             34            65
## 3 Albania             371             100           191
## 4 Alderney             19              3            16
## 5 Algeria             569             263           165
## 6 American Samoa      48              4            42
```

I was only interested in data from the four UK countries. These are: England, Scotland, Wales, and Northern Ireland. I therefore filtered the dataset to only show the values of these teams.

```
#Determine which nations i want to keep
countries_to_keep <-
  c("England", "Scotland", "Wales", "Northern Ireland")
```

```
#Create new dataset with only values of those teams
```

```
UKfooty <- nationstats %>%  
  filter(team %in% countries_to_keep)
```

```
#View new dataset
```

```
head(UKfooty)
```

```
## # A tibble: 4 x 4  
##   team          games_played_total games_won_total games_lost_total  
##   <chr>                <int>          <int>          <int>  
## 1 England              1059            608            209  
## 2 Northern Ireland      683            176            353  
## 3 Scotland              824            393            255  
## 4 Wales                 702            225            321
```

Creation

Before visualising the dataset, I arranged the rows in descending order of total matches played, calculated percentages to show the frequency of wins and losses for each team, and set colour palettes that I thought represented the win and loss categories well.

I also set the themes for the visualisation in advance, alongside the title and subtitles that I wished to show.

After performing these measures in advance, here is the code I used to create the actual visualisation:

```
#Create Plot
```

```
UKFootballPlot = ggplot(UKfooty, aes(x = share, y = team)) +
```

```
#dumbbell segments
```

```
stat_summary(  
  geom = "linerange", fun.min = "min", fun.max = "max",  
  linewidth = c(rep(.8, n), 1.2), color = c(rep(blue_base, n), blue_dark)  
) +
```

```
#dumbbell points
```

```
#white point to go over line endings
```

```
geom_point(  
  aes(x = share), size = 6, shape = 21, stroke = 1, color = "white", fill = "white"  
) +
```

```
#semi-transparent point fill
```

```
geom_point(  
  aes(x = share, fill = type), size = 6, shape = 21, stroke = 1, color = "white", alpha = .7  
) +
```

```
#point outline
```

```
geom_point(  
  aes(x = share), size = 6, shape = 21, stroke = 1, color = "white", fill = NA  
) +
```

```
#result labels
```

```
geom_text(  
  aes(label = percent(share, accuracy = 1, prefix = " ", suffix = "% " ),  
    x = share -0.02, hjust = is_smaller-0.32, color = type),
```

```

fontface = c(rep("plain", n*2), rep("bold", 2)),
family = "sans", size = 4.2
) +

#legend labels
annotate(
  geom = "text", x = c(.18, .60), y = n + 1.8,
  label = c("matches lost", "matches won"), family = "sans",
  fontface = "bold", color = pal_base, size = 5, hjust = .5
) +

coord_cartesian(clip = "off") + #Prevent clipping of points
scale_x_continuous(expand = expansion(add = c(.035, .05)), guide = "none") + #Expand x-axis and rem
scale_y_discrete(expand = expansion(add = c(.35, 1))) + #Expand y axis
scale_color_manual(values = pal_dark) + #Set colour scale for lines
scale_fill_manual(values = pal_base) + #Set fill scale for points
labs(title = title, caption = caption) + #Set plot title and caption
theme(axis.text.y = element_text(face = "bold", size=14)) #Set y axis text to bold and adjust size

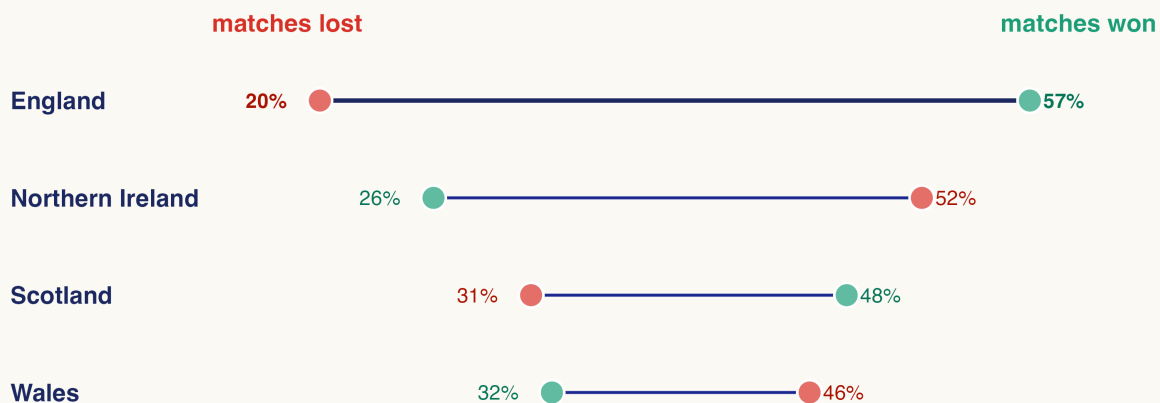
```

The plot was then saved as a png file in the correct dimensions so all information could be shown.

The Visualisation

```
knitr::include_graphics("projectfolder/figs/ukfootballplot.png")
```

Out of the Four UK National Football Teams,
England have the **lowest loss** and **highest win** rates



Teams are ranked according to the number of International Football matches played so far | Source: Mart Jürisoo

Discussion

This dumbbell plot clearly presents the difference in football results across the four UK teams. England are shown to achieve the highest win rates and the lowest percentage of losses, and so could be argued to have fared the best since international football competitions began. On the other hand, Wales show the lowest win rates and the highest percentage of losses.

Plot 3: Can national football results really have an impact on UK stock market values?

The final analysis was the one I was most looking forward to! Will positive or negative football results influence the UK's most reliable stock market in the ftse100?

Data Preperation

To start, I once again had to determine my countries of interest and filter out any matches not containing UK countries. I then removed any rows that contained values occurring before the first of January, 2002. The dataset was then cleaned of any unneeded variables.

```
#Set countries of interest
countries_to_keep <-
  c("England", "Scotland", "Wales", "Northern Ireland")

#Keep rows if UK country is a home or away team
data <-
  data[data$home_team %in% countries_to_keep |
        data$away_team %in% countries_to_keep, ]

#Remove data before 2002
remove_date <- as.Date("2002-01-01")
data <- data[data$Date >= remove_date, ]

#Remove Unneeded Columns
data <-
  subset(data, select = -c(Open, High, Low, Vol., city, country, first_shooter))

#Check data looks ok
head(data)
```

```
##           Date      home_team      away_team  Price Change.. home_score
## 25043 2002-02-13    Netherlands      England 5153.9   0.35%           1
## 25044 2002-02-13 Northern Ireland      Poland 5153.9   0.35%           1
## 25051 2002-02-13         Wales      Argentina 5153.9   0.35%           1
## 25111 2002-03-27         England        Italy 5214.7   0.37%           1
## 25113 2002-03-27         France      Scotland 5214.7   0.37%           5
## 25116 2002-03-27 Liechtenstein Northern Ireland 5214.7   0.37%           0
##           away_score tournament neutral winner
## 25043           1   Friendly   FALSE   draw
## 25044           4   Friendly   TRUE Poland
## 25051           1   Friendly   FALSE   draw
## 25111           2   Friendly   FALSE Italy
## 25113           0   Friendly   FALSE France
## 25116           0   Friendly   FALSE   draw
```

Using the same process as before, match winners were added to the “winner” variable without overriding any values there already from penalty shootouts.

I then needed to create a variable to show how each UK team had fared in their match. In an ideal world this would be easy, as i could just see if a UK team was playing and check if they are in the winner variable. However, UK teams often play against each other, so this process would allocate every one of these games as a UK win (unless it was a draw of course).

I therefore decided to prioritise the result of a country with highest population. Using this logic, if winner of a match was England, the result would show win. If the winner was Scotland, show win if England didn't play. If winner is Wales, show win if England or Scotland didn't play. If winner is Northern Ireland, only show win if England, Scotland, or Wales didn't play. If the winner variable showed draw, obviously I would want the result variable to say draw. Finally, if anything else was the case (the winner was not a UK country or draw), the result variabe would show loss.

Here's the function I created to do this:

```
get_result <- function(winner, home_team, away_team) {
  if (winner == "draw") {
    return("draw")
  } else if (winner == "England") {
    return("win")
  } else if (winner == "Scotland") {
    if ("England" %in% c(home_team, away_team)) {
      return("loss")
    } else {
      return("win")
    }
  } else if (winner == "Wales") {
    if ("England" %in% c(home_team, away_team) || "Scotland" %in% c(home_team, away_team)) {
      return("loss")
    } else {
      return("win")
    }
  } else if (winner == "Northern Ireland") {
    if ("England" %in% c(home_team, away_team) || "Scotland" %in% c(home_team, away_team) || "Wales" %in% c(home_team, away_team)) {
      return("loss")
    } else {
      return("win")
    }
  } else {
    return("loss")
  }
}

# Apply the function to dataset
data$result <- mapply(get_result, data$winner, data$home_team, data$away_team)
```

I then ensured that my variable showing percentage changes in stock prices was a numerical variable, before ordering my results columns in the order: win, draw, loss so that it looks good on the plot.

Using the same techniques as before, I filtered out games from any competitions that were not international tournaments. Friendly games, qualifiers, and the recently created “nations league” very rarely generate as much attention as the much larger World Cup or EUROS tournament games, and so I decided to focus on the impact of these games on the stock market.

Finally, I calculated the total number of matches that fitted into the win, draw, and loss conditions. The aim was to add this information to the plot using the ggplotly function.

```
#Calculate number of each condition
result_n <- tournament_data %>%
  group_by(result) %>%
  summarize(count = n())
```

Creation

Here is the code I used to create the template for the third visualisation.

```
#Create Plot 3
```

```
tournaments_plot <- ggplot(tournament_data %>% # Start with tournament data
  group_by(result) %>% # Group by tournament result
  summarize(Mean_Change = mean(Change..)), # Summarize mean change for each
  aes(x = result, y = Mean_Change, fill = result, # Define aesthetics
  text = paste("Result:", result, # Tooltip text
    "<br>Mean UK Stock Change (%):", round(Mean_Change, 3),
    "<br>Number of Values:", result_n$count[result]))) + # Inc

geom_bar( # Add bar plot
  stat = "identity", # Use identity statistics
  position = "identity", # Use identity position
  width = 0.8, # Set bar width
  colour = "#1b2860", # Set border color
  size = 0.3) + # Set border size

# Set plot labels
labs(title = "Football and The Stock Market",
  x = "UK International Football Result (WC and EUROs only)",
  y = "Mean Stock Value Change (%)",
  subtitle = "Can UK National Teams' Football Results Influence Daily FTSE100 Stock Prices?") +

# Set fill colors manually
scale_fill_manual(values = c("win" = "#1b9e77", "draw" = "#fdae61", "loss" = "#d73027"),
  name = NULL) +

# Set plot theme
theme_light() + # Set light theme
theme(
  panel.background = element_rect(fill = "#fbf9f4", color = "#fbf9f4"), # Set panel background
  plot.background = element_rect(fill = "#fbf9f4", color = "#fbf9f4"), # Set plot background
  panel.border = element_blank(), # Remove panel border
  panel.grid.major.x = element_blank(), # Remove major gridlines on x-axis
  axis.title.x = element_text(color = "#1b2860"), # Set x-axis title color
  axis.title.y = element_text(color = "#1b2860"), # Set y-axis title color
  plot.title = element_text( # Set plot title text properties
    family = "sans", # Set font family
    size = 20, # Set font size
    face = "bold", # Set font face
    color = "#1b2860"), # Set font color
  axis.text = element_text( # Set axis text properties
    family = "sans", # Set font family
```



```

    size = 10, # Set font size
    color = "#1b2860"), # Set font color
axis.title = element_text( # Set axis title properties
    family = "sans", # Set font family
    size = 12)) + # Set font size

geom_hline(yintercept = 0, linetype = "solid", color = "#1b2860") + # Add horizontal line at y = 0
guides(fill = "none") # Remove fill legend

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

```

I then converted this plot into an interactive plot using ggplotly. One thing I had to note was that the code to make subtitles needed to be specifically tailored to the ggplotly function, meaning that these had to be written in after I had made the plot interactive.

```

#Make the plot interactive
tournaments_animated <- ggplotly(tournaments_plot, tooltip = "text")

#A seperate process was needed to include a subtitle to the interactive plot
tournaments_animated <- tournaments_animated %>%
  layout(
    annotations = list(
      text = "Can UK National Teams' Football Results Influence Daily FTSE100 Stock Prices?",
      x = 0, y = 1.04,
      font = list(size = 14, color = "black", family = "sans"),
      xref = "paper", yref = "paper",
      showarrow = FALSE
    )
  )

#Add Data Sources
#Add a text annotation for the stock data source
tournaments_animated <- tournaments_animated %>%
  layout(
    annotations = list(
      text = "Stock Data Source: https://uk.investing.com",
      x = 1,
      y = 0.85, # Adjust the y-coordinate to position the text at the bottom
      xref = "paper",
      yref = "paper",
      showarrow = FALSE,
      font = list(
        family = "sans",
        size = 10, # Adjust the font size as needed
        color = "darkgrey"
      )
    )
  )

```

```

)

# Add a text annotation for the football data source
tournaments_animated <- tournaments_animated %>%
  layout(
    annotations = list(
      text = "Football Data Source: Mart Jürisoo",
      x = 1,
      y = 0.9, # Adjust the y-coordinate to position the text at the bottom
      xref = "paper",
      yref = "paper",
      showarrow = FALSE,
      font = list(
        family = "sans",
        size = 10, # Adjust the font size as needed
        color = "darkgrey"
      )
    )
  )
)

```

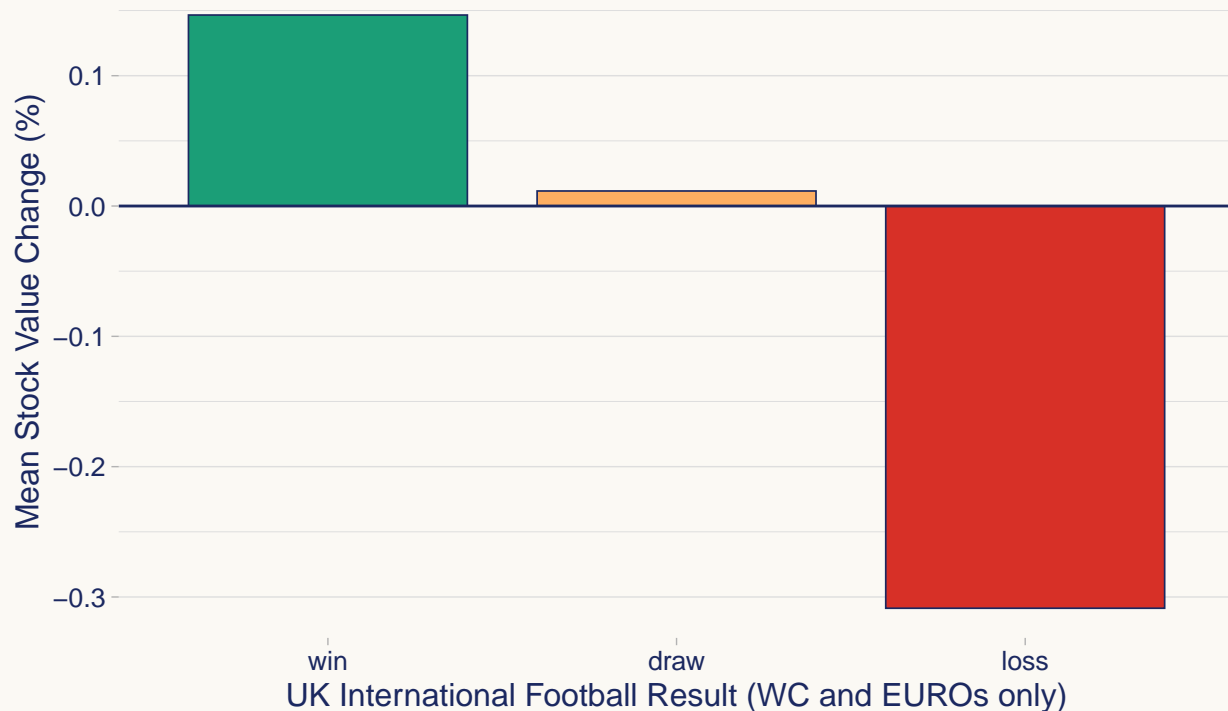
The Visualisation

Unfortunately the ggplotly version of this plot did not seem to work on a PDF. Here is the non-interactive version.

```
print(tournaments_plot)
```

Football and The Stock Market

Can UK National Teams' Football Results Influence Daily FTSE100 Stock Prices?



Discussion

At a first glance, there seems to be some interaction between football results and the UK stock market value changes. When UK nations win football matches, there is an associated increase in Stock Values, and when they lose there is an associated decrease. However, it is worth noting that these mean changes are extremely small, being less than one percent. Furthermore, a statistical analysis of these results found absolutely no significance, so it is important to take these findings with a pinch of salt.

Perhaps a larger analysis (such as Edmans' study) would find greater significance in the small differences between the win/loss conditions, due to the greater sample of football matches to go through. Only analysing the data of four football teams from a relatively short time frame would not have allowed me to determine whether these small football-related stock value changes were truly meaningful.

Summary

Throughout this module I have had the opportunity to really develop my understanding of data science as a whole. I feel much more confident handling and analysing large datasets, and I believe I have really improved upon my ability to create attractive but also informative visualisations.

This module has set me up with skills that I am sure could take me down a number of career pathways. Thank you to Tom and Hazel for teaching and supporting this module in an encouraging and enthusiastic manner. It really was a pleasure to take part in the lab classes each week.