Final

Angel Fernandes

2023-05-22

Let me introduce you to the wonderful world of tennis! Tennis is a captivating sport that has captured the hearts of millions around the globe. It’s a game of skill, strategy, and athleticism, played on a court with a net in the middle. Two players (or teams) use rackets to hit a ball back and forth, aiming to outsmart and outmaneuver their opponents. From the intense rallies to the electrifying atmosphere of major tournaments, tennis offers a thrilling and engaging experience for both players and spectators. So, whether you’re a fan or aspiring player, get ready to dive into the excitement of this incredible sport!

Why do I love tennis you ask? Two Words - Rafael Nadal. He is undeniably one of the most remarkable players in the sport. Nadal’s passion, intensity, and unmatched skills on the court have earned him a massive fan base worldwide. His fierce competitiveness and incredible athleticism make every match he plays a spectacle to behold. Whether it’s his powerful forehand, relentless defense, or never-give-up attitude, Nadal’s style of play has won the hearts of many tennis enthusiasts. It’s no wonder that his exciting performances have inspired countless people to develop a love for the game.

Why do I still love tennis? Two Words - Maria Sharapova. Sharapova is a renowned tennis player who has left a significant impact on the sport. Known for her exceptional skills and fierce determination, she has garnered a large following of admirers around the world. Sharapova’s powerful groundstrokes, aggressive style of play, and mental fortitude on the court have made her a formidable opponent for any challenger. Off the court, she has also made a mark as a role model and a successful entrepreneur. Sharapova’s contributions to tennis have undoubtedly captured the attention and admiration of many fans.

**Getting the Data**

I acquired the data with the help of Github package I found, created by Stephanie Kovalchik (skoval), called ‘deuce’. The package is a great resource for Analysis of Professional Tennis Data. I chose this data because it contains an exclusive collection of data sources and tools for extracting data for professional men’s and women’s tennis ranging from shot-level to match-level data.

**Data Tables**

Below is a list of the data I collected and what they contain: atp\_elo - ATP Elo Ratings atp\_odds - ATP Match Odds atp\_rankings - Rankings of ATP Players atp\_matches - ATP Playing Activity

**Setup** I began by installing and loading the required github package and libraries. Following that, I loaded the datasets as directed by the Github author.

#remotes::install\_github("skoval/deuce")  
#install.packages("deuce")  
library(deuce)

## Loading required package: jsonlite

## Loading required package: dplyr

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

## Loading required package: stringr

## Loading required package: lubridate

##   
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

## Loading required package: rvest

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ forcats 1.0.0 ✔ readr 2.1.4  
## ✔ ggplot2 3.4.2 ✔ tibble 3.2.1  
## ✔ purrr 1.0.1 ✔ tidyr 1.3.0

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ purrr::flatten() masks jsonlite::flatten()  
## ✖ readr::guess\_encoding() masks rvest::guess\_encoding()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(dplyr)  
library(ggplot2)  
library(lubridate)

#For more info:  
help(package = "deuce")

**atp\_elo - Winning Predictions**

I loaded the dataset called atp\_elo by filtering for matches played in the “Wimbledon” tournament, matches occurring on or after January 1, 2000, and involving specific players: “Rafael Nadal,” “Roger Federer,” “Andy Murray,” “Jo-Wilfried Tsonga,” and “Novak Djokovic.” The resulting dataset is then arranged in ascending order first by player name and then by tournament start date. This code allows for the extraction of a subset of data relevant to Wimbledon matches involving specific players from a larger dataset.

data(atp\_elo)  
  
eloWim <- atp\_elo %>%  
 filter(tourney\_name == "Wimbledon") %>%  
 filter(tourney\_start\_date >= "2000-01-01") %>%  
 filter(player\_name %in% c("Rafael Nadal","Roger Federer", "Andy Murray", "Jo-Wilfried Tsonga", "Novak Djokovic")) %>%  
 arrange(player\_name, tourney\_start\_date )

The provided code fits a logistic regression model using the glm() function in R. The response variable is “win,” and the predictors are the differences in “player\_after\_elo” and “player\_before\_elo,” as well as the differences in “opponent\_after\_elo” and “opponent\_before\_elo.” The model assumes a binomial family, indicating that the response variable follows a binomial distribution.

# Fit the logistic regression model  
player\_diff = eloWim$player\_after\_elo - eloWim$player\_before\_elo  
opp\_diff = eloWim$opponent\_after\_elo - eloWim$opponent\_before\_elo  
model <- glm(win ~ player\_diff + opp\_diff, data = eloWim, family = binomial)  
summary(model)

##   
## Call:  
## glm(formula = win ~ player\_diff + opp\_diff, family = binomial,   
## data = eloWim)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.84083 0.17359 10.604 < 2e-16 \*\*\*  
## player\_diff -0.06992 0.01867 -3.745 0.00018 \*\*\*  
## opp\_diff 0.02404 0.01541 1.560 0.11867   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 316.40 on 386 degrees of freedom  
## Residual deviance: 299.22 on 384 degrees of freedom  
## AIC: 305.22  
##   
## Number of Fisher Scoring iterations: 5

# Make predictions  
predictions <- predict(model, newdata = eloWim)  
summary(predictions)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -2.368 1.726 1.899 1.907 2.073 6.880

In this analysis, a logistic regression model was applied to examine the relationship between various Elo ratings and the probability of winning a tennis match. The model used the formula win ~ (player\_after\_elo - player\_before\_elo) + (opponent\_after\_elo - opponent\_before\_elo) and assumed a binomial distribution for the response variable. The coefficients of the model indicate the estimated impact of Elo rating differences on the odds of winning. The intercept term suggests a baseline winning advantage, while the coefficients for the differences in player and opponent Elo ratings provide insights into their individual effects. Additionally, the summary provides descriptive statistics of the response variable, giving an overview of its distribution.

The predictors are the differences in “player\_after\_elo” and “player\_before\_elo” and the differences in “opponent\_after\_elo” and “opponent\_before\_elo.”

The coefficient estimates are as follows:

The intercept is 1.84083. The coefficient for “player\_diff” is -0.06992 The coefficient for “opp\_diff” is 0.02404. The significance of the coefficients is indicated by the p-values. The intercept is significant at the 0.05 level. Both “player\_diff” and “opp\_diff” are not statistically significant.

Graph showing the actual predictions vs the GLM predictions with actual and predicted wins / losses:

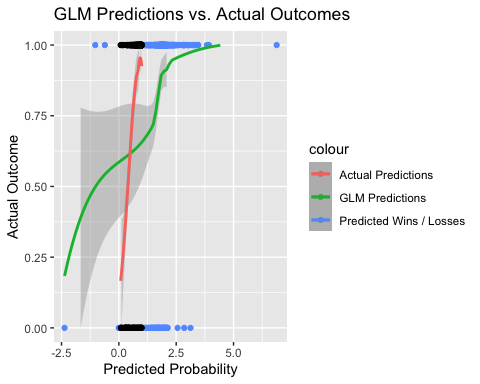
# Create a data frame with actual outcomes and predicted probabilities  
predictions\_df <- data.frame(win = eloWim$win, predictions = eloWim$prediction, glm\_predictions = predictions)  
  
# Create a scatter plot  
predictions\_df %>%  
 ggplot() +  
 geom\_point(aes(x = glm\_predictions, y = win, color = "Predicted Wins / Losses")) +  
 geom\_point(aes(x = predictions, y = win)) +  
 geom\_smooth(aes(x = glm\_predictions, y = win, method = "glm", color = "GLM Predictions")) +  
 geom\_smooth(aes(x = predictions, y = win, method = "glm", color = "Actual Predictions")) +  
 labs(x = "Predicted Probability", y = "Actual Outcome") +  
 ggtitle("GLM Predictions vs. Actual Outcomes") +  
 scale\_y\_continuous(limits = c(0, 1))

## Warning in geom\_smooth(aes(x = glm\_predictions, y = win, method = "glm", :  
## Ignoring unknown aesthetics: method

## Warning in geom\_smooth(aes(x = predictions, y = win, method = "glm", color =  
## "Actual Predictions")): Ignoring unknown aesthetics: method

## `geom\_smooth()` using method = 'loess' and formula = 'y ~ x'  
## `geom\_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 21 rows containing missing values (`geom\_smooth()`).

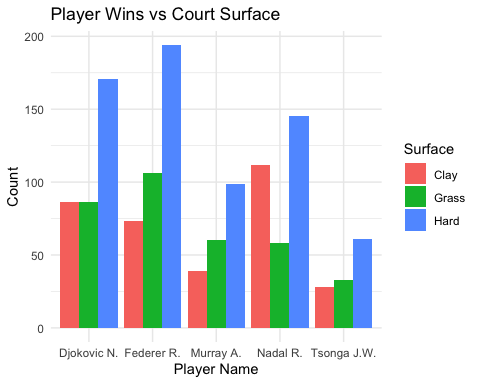


**atp\_odds - Players vs. Courts**

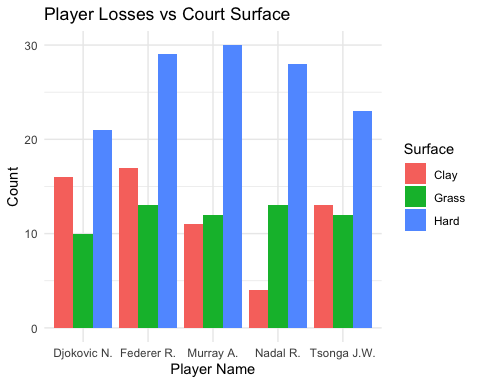
I loaded the dataset called atp\_odds by filtering for matches occurring on or after January 1, 2000, and involving specific players: “Rafael Nadal,” “Roger Federer,” “Andy Murray,” “Jo-Wilfried Tsonga,” and “Novak Djokovic.” Then I created two datasets filtering for winning players and losers and counted the number of times they won / loss grouped by the court surface played on.

Grouped Bar Chart showing number of games won or loss and Court Surface played on:

data(atp\_odds)  
  
atp\_odds <- atp\_odds %>%  
 filter(Date >= "2000-01-01") %>%  
 filter(Winner %in% c("Nadal R.","Federer R.", "Murray A.", "Tsonga J.W.", "Djokovic N.") | Loser %in% c("Nadal R.","Federer R.", "Murray A.", "Tsonga J.W.", "Djokovic N.")) %>%  
 select("ATP", "Location", "Tournament", "Date", "Series", "Court", "Surface", "Round", "Best.of",  
 "Winner", "Loser", "WRank", "LRank", "WPts", "LPts", "W1", "L1", "W2", "L2",  
 "W3", "L3", "W4", "L4", "W5", "L5", "Wsets", "Lsets", "Comment",  
 "B365W", "B365L", "PSW", "PSL", "MaxW", "MaxL", "AvgW", "AvgL")  
  
atp\_odds1 <- atp\_odds %>%  
 group\_by(Surface, player\_name = Winner) %>%  
 count() %>%  
 rename(count\_player = n) %>%  
 filter(player\_name %in% c("Nadal R.","Federer R.", "Murray A.", "Tsonga J.W.", "Djokovic N."))  
   
atp\_odds2 <- atp\_odds %>%  
 group\_by(Surface, player\_name = Loser) %>%  
 count() %>%  
 rename(count\_player = n) %>%  
 filter(player\_name %in% c("Nadal R.","Federer R.", "Murray A.", "Tsonga J.W.", "Djokovic N."))  
  
merged\_data <- rbind(transform(atp\_odds1, dataset = "Winner"),  
 transform(atp\_odds2, dataset = "Loser"))  
  
# Plot the clustered stacked bar chart  
atp\_odds1 %>%  
 ggplot()+  
 geom\_bar(aes(x = player\_name, y = count\_player, fill = Surface), position="dodge", stat = "identity") +  
 labs(x = "Player Name", y = "Count", fill = "Surface") +  
 ggtitle("Player Wins vs Court Surface") +  
 theme\_minimal()



atp\_odds2 %>%  
 ggplot()+  
 geom\_bar(aes(x = player\_name, y = count\_player, fill = Surface), position="dodge", stat = "identity") +  
 labs(x = "Player Name", y = "Count", fill = "Surface") +  
 ggtitle("Player Losses vs Court Surface") +  
 theme\_minimal()

 Observations: 1. All players seem to perform very well and also lost the most games on the Hard Court. 2. Federer is the best on Hard Court and Grass Court but Nadal is the King of the Clay Court. 3. Murray loses the most on Hard Court, followed by Federer and Nadal on the Grass Court and Federer on the Clay Court. 4. Overall, Federer wins more games as compared to Tsonga and Djokovic wins more games as compared to Nadal. 4. Overall, Tsongs loses more games as compared to Federer and Nadal loses more games as compared to Djokovic.

**atp\_rankings - Seeds over the Years**

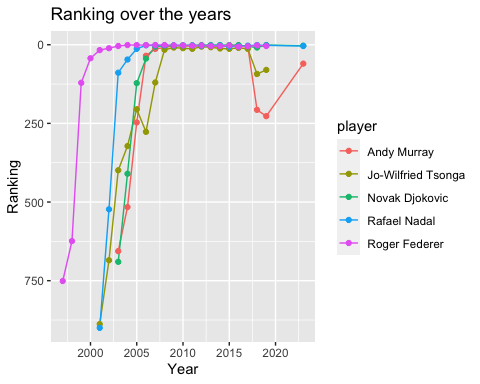
I loaded the dataset called atp\_rankings by filtering for rankings occurring on or after January 1, 2000, and involving specific players: “Rafael Nadal,” “Roger Federer,” “Andy Murray,” “Jo-Wilfried Tsonga,” and “Novak Djokovic”, found by their player\_id numbers.

Line Graph showing ranking over the years:

data(atp\_rankings)  
  
rankings <- atp\_rankings %>%  
 filter(player\_id %in% c(104745, 103819, 104918, 104542, 104925)) %>%  
 arrange(player\_id) %>%  
 mutate(  
 year = as.integer(substr(date, 1, 4)),  
 ranking\_int = as.integer(ranking),  
 ranking\_points\_int = as.integer(ranking\_points),  
 player = case\_when(  
 player\_id == 104745 ~ "Rafael Nadal",  
 player\_id == 103819 ~ "Roger Federer",  
 player\_id == 104918 ~ "Andy Murray",  
 player\_id == 104542 ~ "Jo-Wilfried Tsonga",  
 player\_id == 104925 ~ "Novak Djokovic",  
 TRUE ~ "Unknown"  
 )  
 ) %>%  
 select(player, year, ranking\_int, ranking\_points\_int) %>%  
 group\_by(player, year) %>%  
 summarise(  
 avg\_ranking = as.integer(round(mean(ranking\_int))),  
 avg\_ranking\_points = as.integer(round(mean(ranking\_points\_int)))  
 )

## `summarise()` has grouped output by 'player'. You can override using the  
## `.groups` argument.

rankings %>%  
 ggplot() +  
 geom\_line(aes(x = year, y = avg\_ranking, color = player)) +  
 geom\_point(aes(x = year, y = avg\_ranking, color = player)) +  
 xlab("Year") +  
 ylab("Ranking") +  
 ggtitle("Ranking over the years") +  
 scale\_y\_reverse()

 Observations: 1. Jo-Wilfried Tsonga has the shortest history of ranking 1 with some minor downs along the way. 2. Rodger Federer would have had the longest first seed if he didnt retire in 2022 3. Rafael has held the first seed for 23 years now. 4. Tsonga’s reign lasted about 8 years.

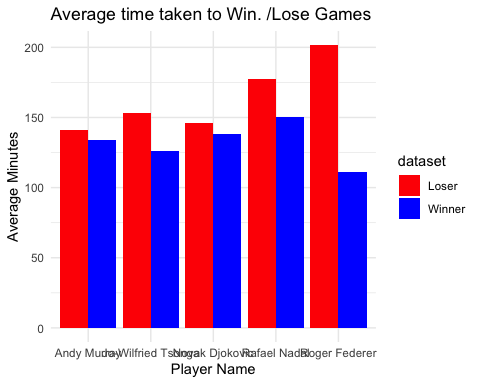
**atp\_matches - Wins / Loses vs Time**

I loaded the dataset called atp\_matches by filtering for matches played in the “Wimbledon” tournament and involving specific players: “Rafael Nadal,” “Roger Federer,” “Andy Murray,” “Jo-Wilfried Tsonga,” and “Novak Djokovic.” Then I created two datasets filtering for winning players and losers and calculated the Average time taken to either win or lose the game.

Grouped Bar Chart showing the Average time taken to either win or lose the game:

data(atp\_matches)  
  
MatchesNadal <- atp\_matches %>%  
 tibble()%>%  
 filter(tourney\_name == "Wimbledon") %>%  
 filter(winner\_name == "Rafael Nadal" | loser\_name == "Rafael Nadal")   
  
MatchesFederer <- atp\_matches %>%  
 filter(tourney\_name == "Wimbledon") %>%  
 filter(winner\_name == "Roger Federer" | loser\_name == "Roger Federer")   
  
MatchesMurray <- atp\_matches %>%  
 filter(tourney\_name == "Wimbledon") %>%  
 filter(winner\_name == "Andy Murray" | loser\_name == "Andy Murray")   
  
MatchesTsonga <- atp\_matches %>%  
 filter(tourney\_name == "Wimbledon") %>%  
 filter(winner\_name == "Jo-Wilfried Tsonga" | loser\_name == "Jo-Wilfried Tsonga")   
  
MatchesDjokovic <- atp\_matches %>%  
 filter(tourney\_name == "Wimbledon") %>%  
 filter(winner\_name == "Novak Djokovic" | loser\_name == "Novak Djokovic")

merged\_matches <- bind\_rows(MatchesNadal, MatchesFederer, MatchesMurray, MatchesTsonga, MatchesDjokovic)  
  
merged\_matches1 <- merged\_matches %>%  
 mutate(minutes = ifelse(is.na(minutes), 0, minutes)) %>%  
 group\_by(player\_name = winner\_name) %>%  
 summarize(average\_minutes = mean(minutes)) %>%  
 filter(player\_name %in% c("Rafael Nadal", "Roger Federer", "Andy Murray", "Jo-Wilfried Tsonga", "Novak Djokovic"))  
  
merged\_matches2 <- merged\_matches %>%  
 mutate(minutes = ifelse(is.na(minutes), 0, minutes)) %>%  
 group\_by(player\_name = loser\_name) %>%  
 summarize(average\_minutes = mean(minutes)) %>%  
 filter(player\_name %in% c("Rafael Nadal", "Roger Federer", "Andy Murray", "Jo-Wilfried Tsonga", "Novak Djokovic"))  
  
merged\_data <- rbind(transform(merged\_matches1, dataset = "Winner"),  
 transform(merged\_matches2, dataset = "Loser"))  
  
# Plot the clustered stacked bar chart  
merged\_data %>%  
 ggplot()+  
 geom\_bar(aes(x = player\_name, y = average\_minutes, fill = dataset), position="dodge", stat = "identity") +  
 labs(x = "Player Name", y = "Average Minutes", fill = "dataset") +  
 ggtitle("Average time taken to Win. /Lose Games")+  
 scale\_fill\_manual(values = c("Winner" = "blue", "Loser" = "red")) +  
 theme\_minimal()



Observations: 1. Tennis matches last anywhere between 2 to 3 hours. 2. The longest tennis match lasted about 11 hours over a period of 3 days between John Isner and Nicolas Mahut in Wimbledon in 2010. 3. Rodger Federer spends the most time trying to salvage his game and ultimately losing but spends less time playing when he does win it. 4. Rafael Nadal spends the most time winning his game and a good chunk of time trying to salvage it.

**Conclusion**

The logistic regression model showed that the baseline winning advantage, represented by the intercept, is statistically significant. However, the differences in player and opponent Elo ratings (player\_diff and opp\_diff) were not found to be statistically significant in pridicting the odds of winning.

In terms of court performance, it was observed that players generally perform well on Hard Courts, while Federer excels on both Hard and Grass Courts, and Nadal dominates on Clay Courts.

Looking at the rankings, Tsonga had a relatively short period as the top-ranked player, while Federer would have held the first seed for the longest duration if he hadn’t retired in 2022. Rafael Nadal has maintained the first seed for 23 years.

The duration of tennis matches typically ranges from 2 to 3 hours, but the longest recorded match lasted an astonishing 11 hours over three days. Federer tends to spend more time salvaging his game in losses but less time playing when he wins, while Nadal invests more time in winning matches and attempting comebacks.