

Main

Wendy

2023-10-30

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
# load in libraries
package_list <- c("tidyverse", "vroom", "stringr", "dplyr", "ggsci")

# load in packages
for (package_name in package_list) {
  if (!requireNamespace(package_name, quietly = TRUE)) {
    install.packages(package_name)
  }
  library(package_name, character.only = TRUE)
}

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.3      v readr      2.1.4
## v forcats    1.0.0      v stringr   1.5.0
## v ggplot2    3.4.3      v tibble    3.2.1
## v lubridate  1.9.2      v tidyr     1.3.0
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
##
## Attaching package: 'vroom'
##
##
## The following objects are masked from 'package:readr':
##
##   as.col_spec, col_character, col_date, col_datetime, col_double,
##   col_factor, col_guess, col_integer, col_logical, col_number,
##   col_skip, col_time, cols, cols_condense, cols_only, date_names,
##   date_names_lang, date_names_langs, default_locale, fwf_cols,
##   fwf_empty, fwf_positions, fwf_widths, locale, output_column,
##   problems, spec

# load in data
df <- vroom("cleandata/data_2022_2023.csv")
```

```

## Rows: 24387 Columns: 14
## -- Column specification -----
## Delimiter: ","
## chr (9): LastName, FirstName, Gender, Country, Date, Competition, Round, Loc...
## dbl (5): Rank, D_Score, E_Score, Penalty, Score
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
# Clean up names to make sure there is unique name for each athlete.
# taking the first 'word' of the names to simplify
df <- df%>%
  mutate(first_name_only = word(FirstName, 1)) %>%
  mutate(last_name_only = word(LastName, 1))

# standardizing countries. Ex know Scotland and Wales all compete with England as Great Britain.
df_c <- df %>%
  mutate(
    clean_country = case_when(Country == "SCO" ~ "GBR",
                              Country == "ENG" ~ "GBR",
                              Country == "WAL" ~ "GBR",
                              Country == "SIN" ~ "SIG",
                              Country == "GE1" ~ "GER",
                              Country == "GE2" ~ "GER",
                              Country == "NIR" ~ "GBR",
                              Country == "IOM" ~ "GBR",
                              Country == "PUR" ~ "USA",
                              TRUE ~ Country

  )
)

# Simplifying accents in names and dashes
replace_letters <- function(name){
  name <- gsub("Ö", "OE", name)
  name <- gsub("Ü", "U", name)
  name <- gsub("Ø", "O", name)
  name <- sub("-.*", "", name)
  name <- gsub("Ä", "AE", name)
  return(name)
}

standardize <- function(name){
  name <- iconv(name, "UTF-8", "ASCII//TRANSLIT")
  return (name)
}

# Applying accent cleaning function to dataframe's name columns.
df_c$first_name_only <- lapply(df_c$first_name_only, replace_letters)
df_c$last_name_only <- lapply(df_c$last_name_only, replace_letters)

df_c$first_name_only <- lapply(df_c$first_name_only, standardize)
df_c$last_name_only <- lapply(df_c$last_name_only, standardize)

```

```

# mutate to create name variable comprising of both first and last name
df_c <- df_c %>%
  mutate(name = paste(first_name_only, last_name_only))

# Separating into two dataset of men and women's gymnastics
df_men_c <- df_c %>%
  filter(Gender == "m")

df_women_c <- df_c %>%
  filter(Gender == "w")

# Manual cleaning done on name for women's data
df_women_c <- df_women_c %>%
  mutate(
    clean_name = case_when(
      name == "Anna KONIG" ~ "Anna KOENIG",
      name == "Aberdeen O" ~ "Aberdeen O'DRISCOL",
      name == "Christina KIOSO" ~ "Christina VASILOPOULOU",
      name == "Korkem YEROBSSYNKYZY" ~ "Korkem YERBOSSYNKYZY",
      name == "Larasati REGGANIS" ~ "Larasati RENGGANIS",
      name == "Liu MAI" ~ "Mai LIU",
      name == "Melanie DE" ~ "Melanie DE JESUS",
      name == "Melanie JESUS" ~ "Melanie DE JESUS",
      name == "Milka ELPITIYA" ~ "Milka GEHANI",
      name == "NA ELPITIYA" ~ "Milka GEHANI",
      name == "Nika KUKULJAN" ~ "Nika FRLETA",
      name == "Olivia MAFES" ~ "Olivia Kelly",
      name == "Ruby EVAN" ~ "Ruby EVANS",
      name == "Sasiwimion MUEANGPHUAN" ~ "Sasiwimon MUEANGPHUAN",
      name == "Shade OORSCHOT" ~ "Shade VAN OORSCHOT",
      name == "Shade VAN" ~ "Shade VAN OORSCHOT",
      name == "Vera VAN" ~ "Vera VAN POL",
      name == "Vera POL" ~ "Vera VAN POL",
      name == "Wong VANESSA" ~ "Vanessa WONG",
      name == "Zeina SHARAF" ~ "Zeina IBRAHIM",
      name == "NA JESUS" ~ "Melanie DE JESUS",
      TRUE ~ name
    )
  )

# Manual cleaning done on country specifications for women's data
df_women_c <- df_women_c %>%
  mutate(
    clean_country = ifelse(clean_name == "Teja BELAK", "SLO", clean_country),
    clean_country = ifelse(clean_name == "Naveen DARIES", "RSA", clean_country),
    clean_country = ifelse(clean_name == "Madeleine MARSHALL", "NZL", clean_country)
  )

df_women_c <- df_women_c %>%
  filter(!is.na(first_name_only))

numeric_columns <- sapply(df_women_c, is.numeric)
df_women_c[numeric_columns][is.na(df_women_c[numeric_columns])] <- 0

```

```

# Manual cleaning done on name for men's data
df_men_c <- df_men_c %>%
  mutate(
    clean_name = case_when(
      name == "Mc ATEER" ~ "Ewan MCATEER",
      name == "Ewan MC" ~ "Ewan MCATEER",
      name == "Mc CLENAGHAN" ~ "Rhys MCCLENAGHAN",
      name == "Rhys MC" ~ "Rhys MCCLENAGHAN",
      name == "Samual DICK" ~ "Samuel DICK",
      name == "William FU" ~ "William FUALLEN",
      name == "Jermain GRUNBERG" ~ "Jermain GRUENBERG",
      name == "Rakah HARITHI" ~ "Rakan HARITHI",
      name == "Vinzenn HOECK" ~ "Vinzenn HOECK",
      name == "Vinzenn HOECK" ~ "Vinzenn HOECK",
      name == "Mohamad KHALIL" ~ "Mohamed KHALIL",
      name == "Severin KRANZLMULLER" ~ "Severin KRANZLMUELLER",
      name == "NA KUMARASINGHEGE" ~ "Hansha KUMARASINGHEGE",
      name == "Manuel MARTINEZ" ~ "Jose MARTINEZ",
      name == "Felipe MARTINEZ" ~ "Andres MARTINEZ",
      name == "Clay MASON" ~ "Clay MASONSTEPHENS",
      name == "Dimitrijs MICKEVICS" ~ "Dmitrijs MICKEVICS",
      name == "Dmitry PATANIN" ~ "Dmitriy PATANIN",
      name == "NA SOUZA" ~ "Caio SOUZA",
      name == "Jann TIMBANG" ~ "Jan TIMBANG",
      name == "Kim VANSTROM" ~ "Kim VANSTROEM",
      name == "Sam ZAKUTNEY" ~ "Samuel ZAKUTNEY",
      TRUE ~ name
    )
  )

# Manual cleaning done on country specifications for men's data
df_men_c <- df_men_c %>%
  mutate(
    clean_country = ifelse(clean_name == "Ewan MCATEER", "IRL", clean_country),
    clean_country = ifelse(clean_name == "Rhys MCCLENAGHAN", "IRL", clean_country),
    clean_country = ifelse(clean_name == "Dominick CUNNINGHAM", "IRL", clean_country),
    clean_country = ifelse(clean_name == "Eamon MONTGOMERY", "IRL", clean_country),
    clean_country = ifelse(clean_name == "NIkita SIMONOV", "AZE", clean_country)
  )

df_men_c <- df_men_c %>%
  filter(!is.na(first_name_only))

numeric_columns <- sapply(df_men_c, is.numeric)
df_men_c[numeric_columns][is.na(df_men_c[numeric_columns])] <- 0

df_men_names <- df_men_c %>%
  distinct(last_name_only, first_name_only, clean_country, name)

unique(df_women_c$Competition)

## [1] "2022 51st FIG Artistic Gymnastics World Championships"
## [2] "2022 Cairo World Cup"
## [3] "2022 Paris World Challenge Cup"

```

```

## [4] "2023 52nd FIG Artistic Gymnastics World Championships"
## [5] "2023 Baku World Cup"
## [6] "2023 Cairo World Cup"
## [7] "2023 Doha World Cup"
## [8] "2022 British Gymnastics Championships"
## [9] "BIRMINGHAM 2022 Commonwealth Games"
## [10] "2023 Artistic Gymnastics Senior Pan American Championships"
## [11] "2022 Senior European Championships"
## [12] "2023 British Gymnastics Championships"
## [13] "2023 Senior European Championships"
## [14] "2022 Osijek World Challenge Cup"
## [15] "SANTIAGO 2023 XIX Pan American Games"
## [16] "2022 U.S. Championships"
## [17] "2022 U.S. Classic"
## [18] "2022 Winter Cup"
## [19] "2022 9th Senior Artistic Gymnastics Asian Championships"
## [20] "2023 10th Senior Artistic Gymnastics Asian Championships"
## [21] "2023 Varna World Challenge Cup"
## [22] "HANGZHOU 2022 19th Asian Games"
## [23] "2022 Mersin World Challenge Cup"
## [24] "2023 Tel Aviv World Challenge Cup"
## [25] "2023 Core Hydration Classic"
## [26] "2023 FISU World University Games"
## [27] "2023 Cottbus World Cup"
## [28] "2022 Baku World Cup"
## [29] "2022 Varna World Challenge Cup"
## [30] "2023 Central American and Caribbean Games"
## [31] "2022 Doha World Cup"
## [32] "2023 Osijek World Challenge Cup"
## [33] "EnBW DTB Pokal Team Challenge 2023"
## [34] "2022 Cottbus World Cup"
## [35] "2022 Koper World Challenge Cup"
## [36] "2022 Szombathely World Challenge Cup"
## [37] "2023 U.S. Championships"
## [38] "2023 Winter Cup"

df_women_c <- df_women_c %>%
  mutate(competition_number = case_when(
    Competition == "2022 Cottbus World Cup" ~ "1",
    Competition == "2022 Winter Cup" ~ "2",
    Competition == "2022 Doha World Cup" ~ "3",
    Competition == "2022 Cairo World Cup" ~ "4",
    Competition == "2022 British Gymnastics Championships" ~ "5",
    Competition == "2022 Varna World Challenge Cup" ~ "6",
    Competition == "2022 Baku World Cup" ~ "7",
    Competition == "2022 Osijek World Challenge Cup" ~ "8",
    Competition == "2022 9th Senior Artistic Gymnastics Asian Championships" ~ "9",
    Competition == "2022 Koper World Challenge Cup" ~ "10",
    Competition == "2022 U.S. Classic" ~ "11",
    Competition == "BIRMINGHAM 2022 Commonwealth Games" ~ "12",
    Competition == "2022 Senior European Championships" ~ "13",
    Competition == "2022 U.S. Championships" ~ "14",
    Competition == "2022 Paris World Challenge Cup" ~ "15",
    Competition == "2022 Szombathely World Challenge Cup" ~ "16",

```

```

Competition == "2022 Mersin World Challenge Cup" ~ "17",
Competition == "2022 51st FIG Artistic Gymnastics World Championships" ~ "18",
Competition == "2023 Cottbus World Cup" ~ "19",
Competition == "2023 Winter Cup" ~ "20",
Competition == "2023 Doha World Cup" ~ "21",
Competition == "2023 Baku World Cup" ~ "22",
Competition == "EnBW DTB Pokal Team Challenge 2023" ~ "23",
Competition == "2023 British Gymnastics Championships" ~ "24",
Competition == "2023 Senior European Championships" ~ "25",
Competition == "2023 Cairo World Cup" ~ "26",
Competition == "2023 Varna World Challenge Cup" ~ "27",
Competition == "2023 Artistic Gymnastics Senior Pan American Championships" ~ "28",
Competition == "2023 Tel Aviv World Challenge Cup" ~ "29",
Competition == "2023 Osijek World Challenge Cup" ~ "30",
Competition == "2023 10th Senior Artistic Gymnastics Asian Championships" ~ "31",
Competition == "2023 Central American and Caribbean Games" ~ "32",
Competition == "2023 FISU World University Games" ~ "33",
Competition == "2023 Core Hydration Classic" ~ "34",
Competition == "2023 U.S. Championships" ~ "35",
Competition == "2023 52nd FIG Artistic Gymnastics World Championships" ~ "36",
TRUE ~ "Other"
))

df_women_c = df_women_c %>%
  mutate(competition_number = as.numeric(competition_number))

## Warning: There was 1 warning in `mutate()`.
## i In argument: `competition_number = as.numeric(competition_number)`.
## Caused by warning:
## ! NAs introduced by coercion

df_women_c <- df_women_c %>%
  mutate(competition_scope = case_when(
    Competition == "2022 Cottbus World Cup" ~ "World",
    Competition == "2022 Winter Cup" ~ "USA",
    Competition == "2022 Doha World Cup" ~ "World",
    Competition == "2022 Cairo World Cup" ~ "World",
    Competition == "2022 British Gymnastics Championships" ~ "non-USA",
    Competition == "2022 Varna World Challenge Cup" ~ "World",
    Competition == "2022 Baku World Cup" ~ "World",
    Competition == "2022 Osijek World Challenge Cup" ~ "World",
    Competition == "2022 9th Senior Artistic Gymnastics Asian Championships" ~ "non-USA",
    Competition == "2022 Koper World Challenge Cup" ~ "World",
    Competition == "2022 U.S. Classic" ~ "USA",
    Competition == "BIRMINGHAM 2022 Commonwealth Games" ~ "non-USA",
    Competition == "2022 Senior European Championships" ~ "non-USA",
    Competition == "2022 U.S. Championships" ~ "USA",
    Competition == "2022 Paris World Challenge Cup" ~ "World",
    Competition == "2022 Szombathely World Challenge Cup" ~ "World",
    Competition == "2022 Mersin World Challenge Cup" ~ "World",
    Competition == "2022 51st FIG Artistic Gymnastics World Championships" ~ "World",
    Competition == "2023 Cottbus World Cup" ~ "World",
    Competition == "2023 Winter Cup" ~ "USA",
    Competition == "2023 Doha World Cup" ~ "World",
  ))

```

```

Competition == "2023 Baku World Cup" ~ "World",
Competition == "EnBW DTB Pokal Team Challenge 2023" ~ "World",
Competition == "2023 British Gymnastics Championships" ~ "non-USA",
Competition == "2023 Senior European Championships" ~ "non-USA",
Competition == "2023 Cairo World Cup" ~ "World",
Competition == "2023 Varna World Challenge Cup" ~ "World",
Competition == "2023 Artistic Gymnastics Senior Pan American Championships" ~ "USA",
Competition == "2023 Tel Aviv World Challenge Cup" ~ "World",
Competition == "2023 Osijek World Challenge Cup" ~ "World",
Competition == "2023 10th Senior Artistic Gymnastics Asian Championships" ~ "non-USA",
Competition == "2023 Central American and Caribbean Games" ~ "non-USA",
Competition == "2023 FISU World University Games" ~ "USA",
Competition == "2023 Core Hydration Classic" ~ "USA",
Competition == "2023 U.S. Championships" ~ "USA",
Competition == "2023 52nd FIG Artistic Gymnastics World Championships" ~ "World",
TRUE ~ "Other"
))

```

```

df_men_c <- df_men_c %>%
  mutate(competition_number = case_when(
    Competition == "2022 Cottbus World Cup" ~ "1",
    Competition == "2022 Winter Cup" ~ "2",
    Competition == "2022 Doha World Cup" ~ "3",
    Competition == "2022 Cairo World Cup" ~ "4",
    Competition == "2022 British Gymnastics Championships" ~ "5",
    Competition == "2022 Varna World Challenge Cup" ~ "6",
    Competition == "2022 Baku World Cup" ~ "7",
    Competition == "2022 Osijek World Challenge Cup" ~ "8",
    Competition == "2022 9th Senior Artistic Gymnastics Asian Championships" ~ "9",
    Competition == "2022 Koper World Challenge Cup" ~ "10",
    Competition == "2022 U.S. Classic" ~ "11",
    Competition == "BIRMINGHAM 2022 Commonwealth Games" ~ "12",
    Competition == "2022 Senior European Championships" ~ "13",
    Competition == "2022 U.S. Championships" ~ "14",
    Competition == "2022 Paris World Challenge Cup" ~ "15",
    Competition == "2022 Szombathely World Challenge Cup" ~ "16",
    Competition == "2022 Mersin World Challenge Cup" ~ "17",
    Competition == "2022 51st FIG Artistic Gymnastics World Championships" ~ "18",
    Competition == "2023 Cottbus World Cup" ~ "19",
    Competition == "2023 Winter Cup" ~ "20",
    Competition == "2023 Doha World Cup" ~ "21",
    Competition == "2023 Baku World Cup" ~ "22",
    Competition == "EnBW DTB Pokal Team Challenge 2023" ~ "23",
    Competition == "2023 British Gymnastics Championships" ~ "24",
    Competition == "2023 Senior European Championships" ~ "25",
    Competition == "2023 Cairo World Cup" ~ "26",
    Competition == "Oceania Continental Championships 2023" ~ "27",
    Competition == "2023 Varna World Challenge Cup" ~ "28",
    Competition == "2023 Artistic Gymnastics Senior Pan American Championships" ~ "29",
    Competition == "2023 Tel Aviv World Challenge Cup" ~ "30",
    Competition == "2023 Osijek World Challenge Cup" ~ "31",
    Competition == "2023 10th Senior Artistic Gymnastics Asian Championships" ~ "32",
    Competition == "2023 Central American and Caribbean Games" ~ "33",
    Competition == "2023 FISU World University Games" ~ "34",
  )
)

```



```

    Competition == "2023 Core Hydration Classic" ~ "35",
    Competition == "2023 U.S. Championships" ~ "36",
    Competition == "2023 52nd FIG Artistic Gymnastics World Championships" ~ "37",
    TRUE ~ "Other"
  ))

df_men_c = df_men_c %>%
  mutate(competition_number = as.numeric(competition_number))

## Warning: There was 1 warning in `mutate()`.
## i In argument: `competition_number = as.numeric(competition_number)`.
## Caused by warning:
## ! NAs introduced by coercion

df_men_c <- df_men_c %>%
  mutate(competition_scope = case_when(
    Competition == "2022 Cottbus World Cup" ~ "World",
    Competition == "2022 Winter Cup" ~ "USA",
    Competition == "2022 Doha World Cup" ~ "World",
    Competition == "2022 Cairo World Cup" ~ "World",
    Competition == "2022 British Gymnastics Championships" ~ "non-USA",
    Competition == "2022 Varna World Challenge Cup" ~ "World",
    Competition == "2022 Baku World Cup" ~ "World",
    Competition == "2022 Osijek World Challenge Cup" ~ "World",
    Competition == "2022 9th Senior Artistic Gymnastics Asian Championships" ~ "non-USA",
    Competition == "2022 Koper World Challenge Cup" ~ "World",
    Competition == "2022 U.S. Classic" ~ "USA",
    Competition == "BIRMINGHAM 2022 Commonwealth Games" ~ "non-USA",
    Competition == "2022 Senior European Championships" ~ "non-USA",
    Competition == "2022 U.S. Championships" ~ "USA",
    Competition == "2022 Paris World Challenge Cup" ~ "World",
    Competition == "2022 Szombathely World Challenge Cup" ~ "World",
    Competition == "2022 Mersin World Challenge Cup" ~ "World",
    Competition == "2022 51st FIG Artistic Gymnastics World Championships" ~ "World",
    Competition == "2023 Cottbus World Cup" ~ "World",
    Competition == "2023 Winter Cup" ~ "USA",
    Competition == "2023 Doha World Cup" ~ "World",
    Competition == "2023 Baku World Cup" ~ "World",
    Competition == "EnBW DTB Pokal Team Challenge 2023" ~ "World",
    Competition == "2023 British Gymnastics Championships" ~ "non-USA",
    Competition == "2023 Senior European Championships" ~ "non-USA",
    Competition == "2023 Cairo World Cup" ~ "World",
    Competition == "Oceania Continental Championships 2023" ~ "non-USA",
    Competition == "2023 Varna World Challenge Cup" ~ "World",
    Competition == "2023 Artistic Gymnastics Senior Pan American Championships" ~ "USA",
    Competition == "2023 Tel Aviv World Challenge Cup" ~ "World",
    Competition == "2023 Osijek World Challenge Cup" ~ "World",
    Competition == "2023 10th Senior Artistic Gymnastics Asian Championships" ~ "non-USA",
    Competition == "2023 Central American and Caribbean Games" ~ "non-USA",
    Competition == "2023 FISU World University Games" ~ "USA",
    Competition == "2023 Core Hydration Classic" ~ "USA",
    Competition == "2023 U.S. Championships" ~ "USA",
    Competition == "2023 52nd FIG Artistic Gymnastics World Championships" ~ "World",
    TRUE ~ "Other"
  ))

```



```
))
```

EDA

##for women

```
ggplot(data = df_women_c, aes(y = D_Score, fill = Round)) +
  geom_boxplot() +
  facet_wrap(~Apparatus) +
  theme_light() +
  scale_fill_aaas() +
  labs(
    #title = "Distribution of the Difficulty Score by Round and Apparatus",
    # subtitle = "For Women",
    fill = "Round",
    y = "Difficulty Score"
  )
```

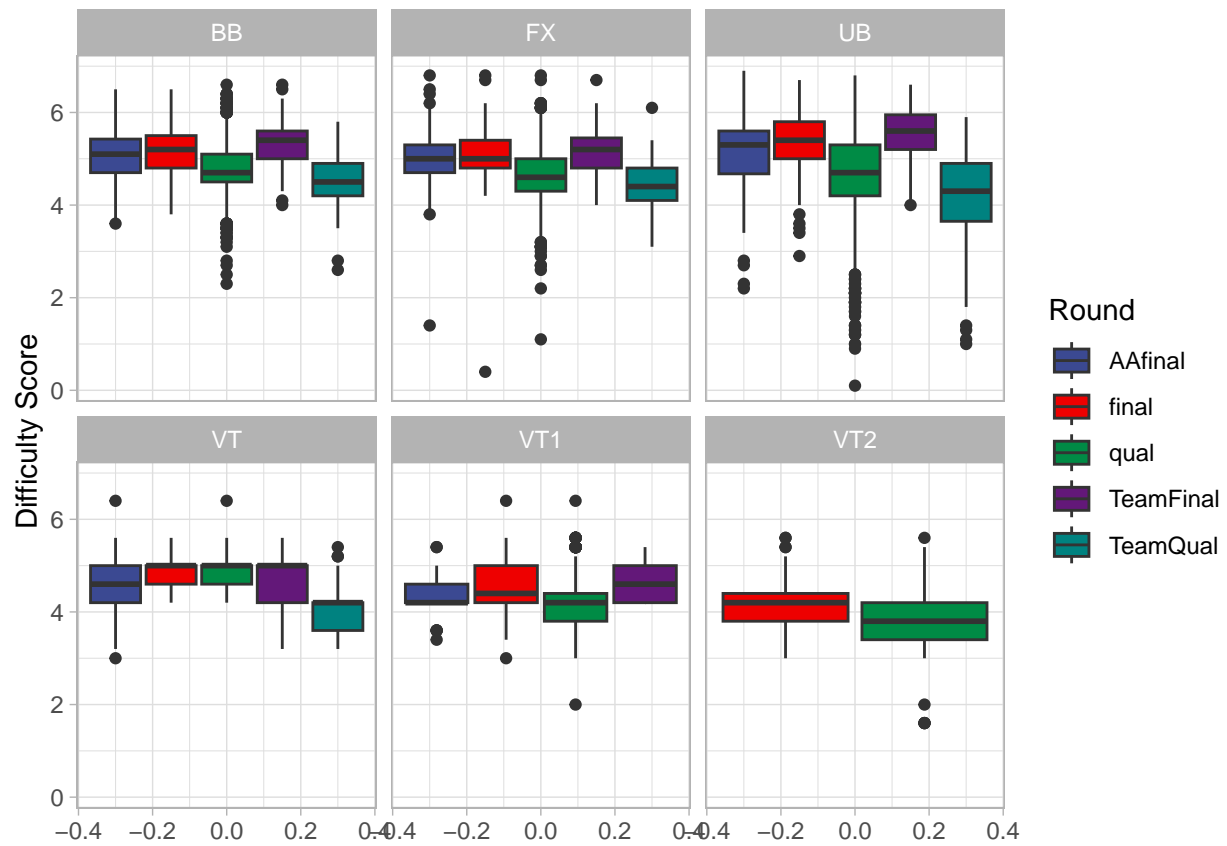


Figure 1: Distribution of the Difficulty Score by Round and Apparatus for Women

```
ggplot(data = df_women_c, aes(y = D_Score, fill = competition_scope)) +
  geom_boxplot() +
  facet_wrap(~Apparatus) +
  theme_light() +
  scale_fill_aaas() +
  labs(
```

```

fill = "Competition Scope",
y = "Difficulty Score"
)

```

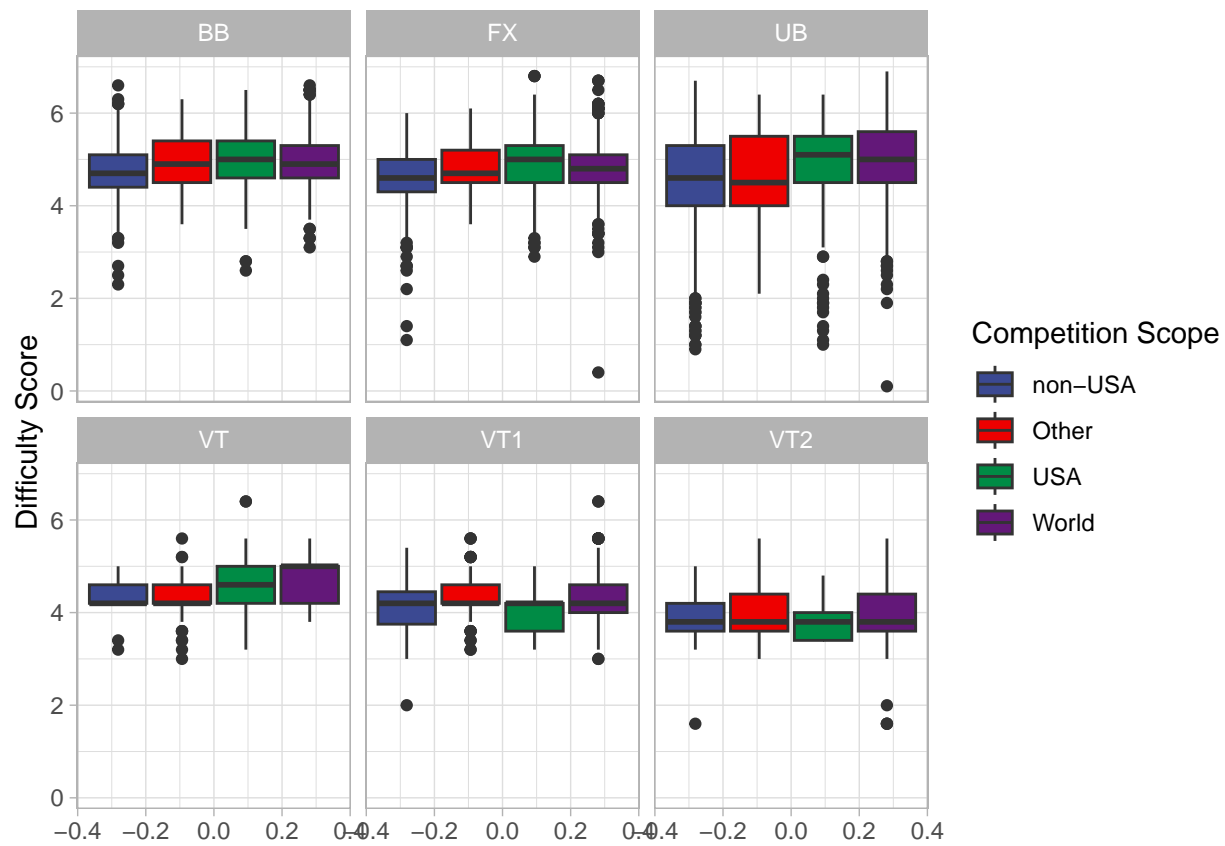


Figure 2: Distribution of Difficulty scores by the Scope of Competition and Apparatus for Women

```

ggplot(data = df_women_c, aes(y = Score, fill = competition_scope))+
  geom_boxplot()+
  facet_wrap(~Apparatus)+
  theme_light()+
  scale_fill_aaas()+
  labs(
    fill = "Competition Scope"
  )

```

For men

```

ggplot(data = df_men_c, aes(y = D_Score, fill = Round))+
  geom_boxplot()+
  facet_wrap(~Apparatus) +
  theme_light()+
  scale_fill_aaas()+
  labs(
    # title = "Distribution of the Difficulty Score by Round and Apparatus",
    # subtitle = "For Men",
  )

```

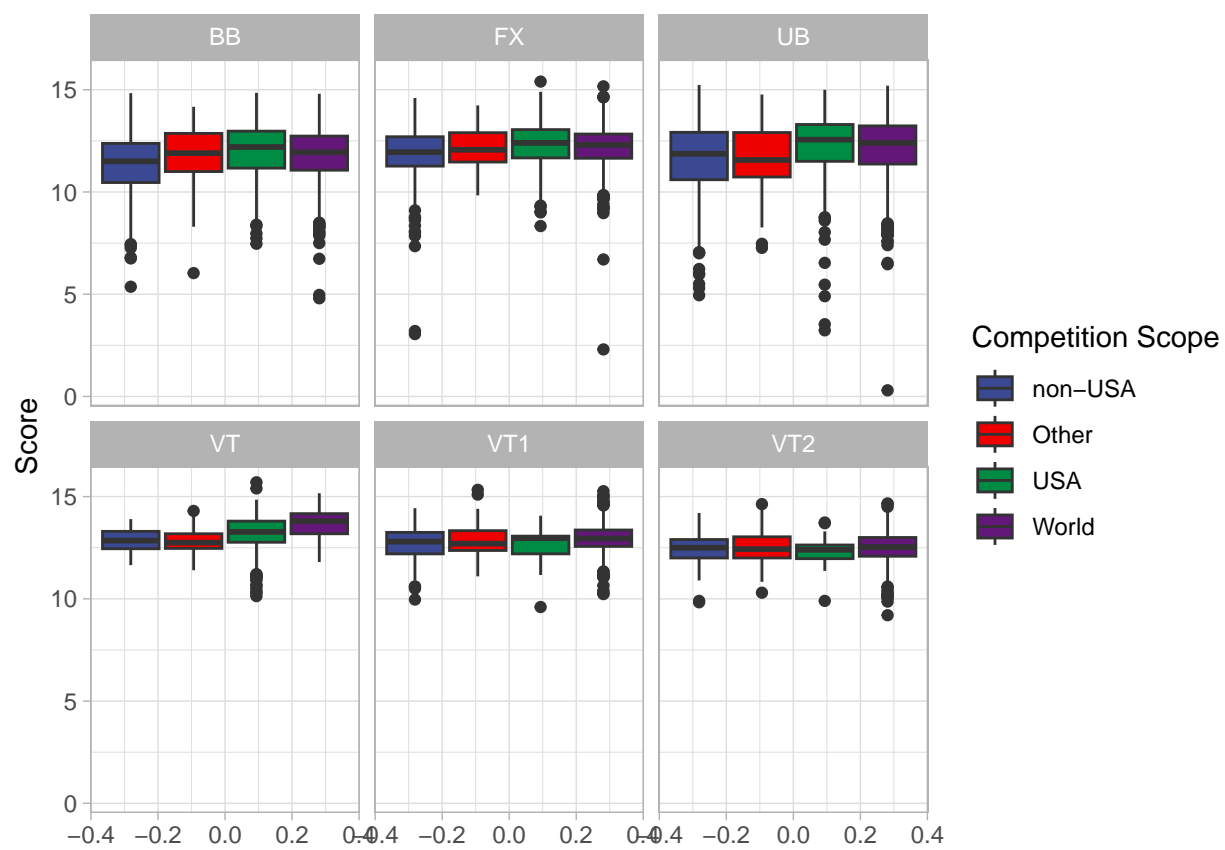
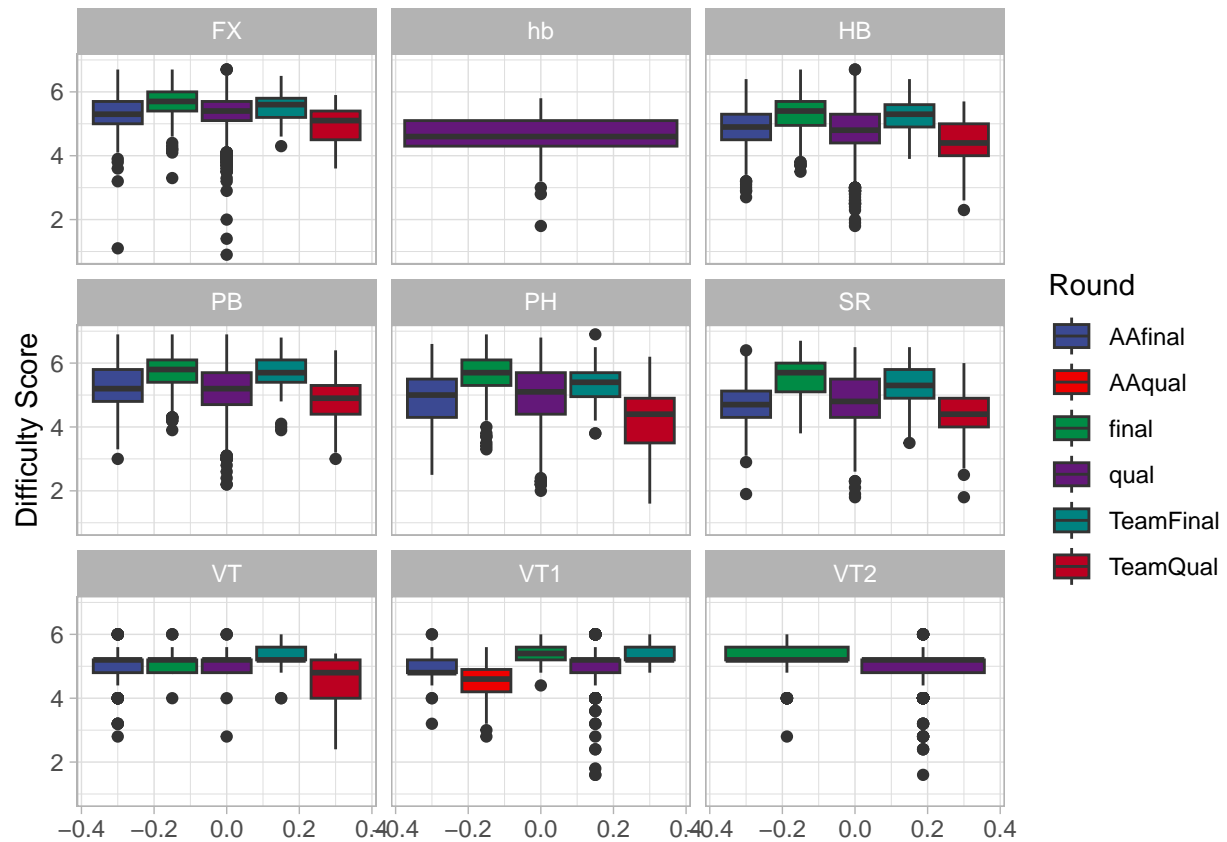


Figure 3: Distribution of Scores by Scope of Competition and Apparatus for Women

```

fill = "Round",
y = "Difficulty Score"
)

```



Waiting for competition scope for men

```

ggplot(data = df_men_c, aes(y = D_Score, fill = competition_scope))+
  geom_boxplot()+
  facet_wrap(~Apparatus)+
  theme_light()+
  scale_fill_aaas()+
  labs(
    #title = "Distribution of Difficulty scores by the Scope of Competition and Apparatus",
    #subtitle = "For Men",
    fill = "Competition Scope",
    y = "Difficulty Score"
  )

```

```

ggplot(data = df_men_c, aes(y = Score, fill = competition_scope))+
  geom_boxplot()+
  facet_wrap(~Apparatus)+
  theme_light()+
  scale_fill_aaas()+
  labs(
    # title = "Distribution of Scores by Scope of Competition and Apparatus",
    # subtitle = "For Men",
    fill = "Competition Scope"
  )

```

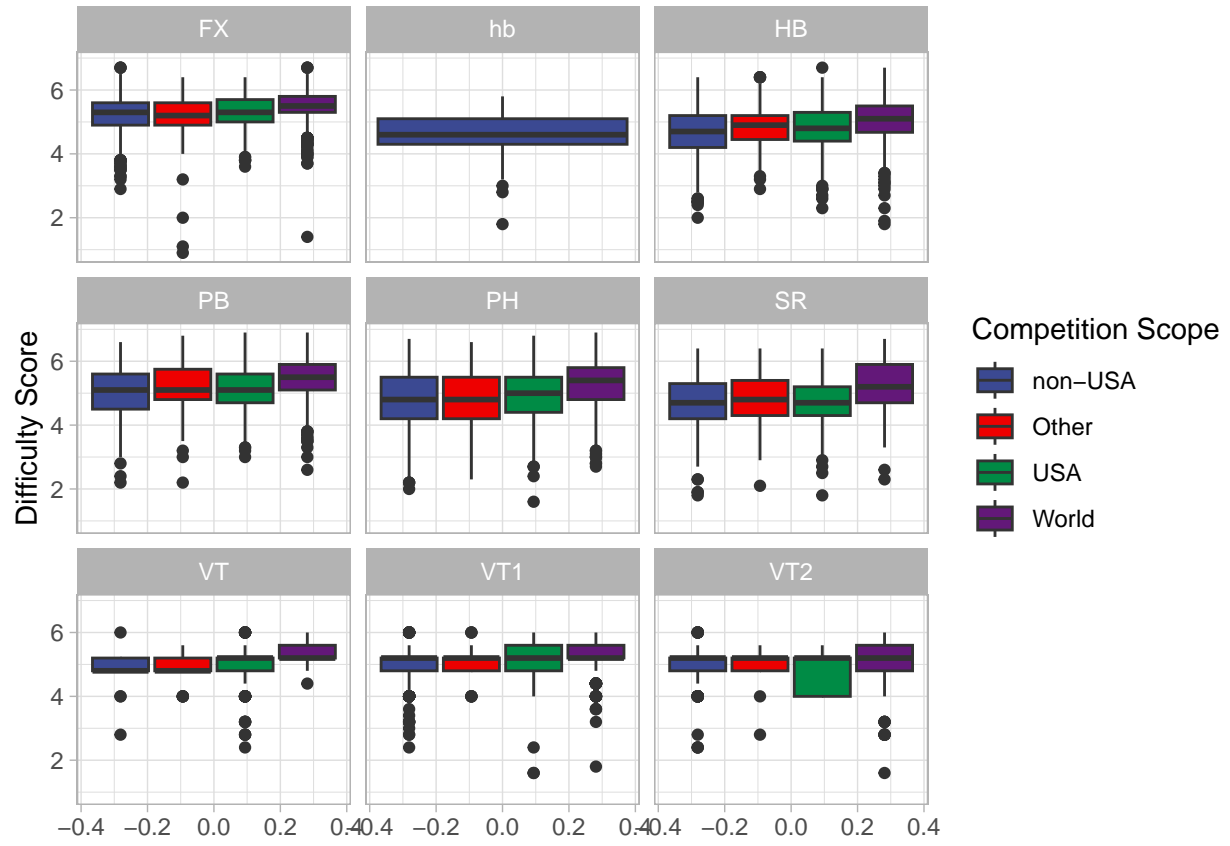


Figure 4: Distribution of Difficulty scores by the Scope of Competition and Apparatus for Men

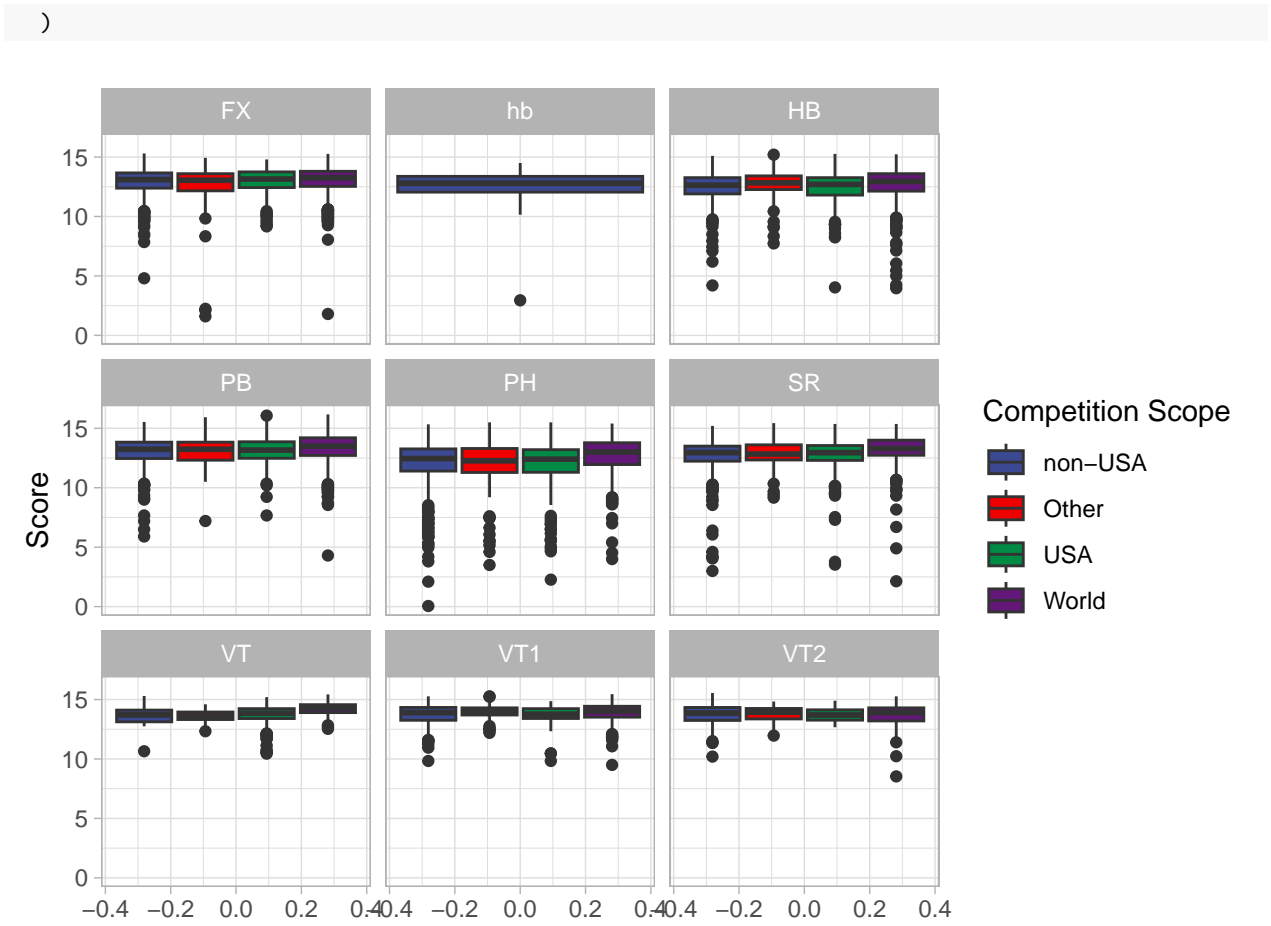


Figure 5: Distribution of Scores by Scope of Competition and Apparatus for Men