Project1

Frederick Jones

2023-09-17

Import libraries

```
library(tidyverse)
```

```
## -- 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
```

```
library(stringr)
```

Load the data

```
text_data <- read_lines("https://raw.githubusercontent.com/jewelercart/R/main/tournamentinfo.txt")
head(text_data)
```

```
## [1] "-----------------------------------------------------------------------------------------"
## [2] " Pair | Player Name                     |Total|Round|Round|Round|Round|Round|Round|Round| "
## [3] " Num  | USCF ID / Rtg (Pre->Post)       | Pts |  1  |  2  |  3  |  4  |  5  |  6  |  7  | "
## [4] "-----------------------------------------------------------------------------------------"
## [5] "    1 | GARY HUA                        |6.0  |W  39|W  21|W  18|W  14|W   7|D  12|D   4|"
## [6] "   ON | 15445895 / R: 1794   ->1817     |N:2  |W    |B    |W    |B    |W    |B    |W    |"
```

Preprocessing the data

Getting names of all the players

```r
player_names <- character(0)
total_point <- numeric(0)
# Define a regular expression pattern to match player names
pattern <- "^\\s*\\d+\\s+\\|\\s+(.+?)\\s+\\|.*$"

# Iterate through lines in the file
for (line in text_data) {
  # Use regular expression to extract player names
  if (grepl(pattern, line)) {
    match_data <- str_match(line, pattern)
    player_name <- match_data[2]
    player_names <- c(player_names, player_name)
    point<- str_extract(line, "[[:digit:]]+\\.[[:digit:]]")
    total_point<- c(total_point, as.numeric(point))
  }

}

# Print the extracted player names
print("Players are : ")
```

```
## [1] "Players are : "
```

```r
print(player_names)
```

```
##  [1] "GARY HUA"                  "DAKSHESH DARURI"
##  [3] "ADITYA BAJAJ"              "PATRICK H SCHILLING"
##  [5] "HANSHI ZUO"                "HANSEN SONG"
##  [7] "GARY DEE SWATHELL"         "EZEKIEL HOUGHTON"
##  [9] "STEFANO LEE"               "ANVIT RAO"
## [11] "CAMERON WILLIAM MC LEMAN"  "KENNETH J TACK"
## [13] "TORRANCE HENRY JR"         "BRADLEY SHAW"
## [15] "ZACHARY JAMES HOUGHTON"    "MIKE NIKITIN"
## [17] "RONALD GRZEGORCZYK"        "DAVID SUNDEEN"
## [19] "DIPANKAR ROY"              "JASON ZHENG"
## [21] "DINH DANG BUI"             "EUGENE L MCCLURE"
## [23] "ALAN BUI"                  "MICHAEL R ALDRICH"
## [25] "LOREN SCHWIEBERT"          "MAX ZHU"
## [27] "GAURAV GIDWANI"            "SOFIA ADINA STANESCU-BELLU"
## [29] "CHIEDOZIE OKORIE"          "GEORGE AVERY JONES"
## [31] "RISHI SHETTY"              "JOSHUA PHILIP MATHEWS"
## [33] "JADE GE"                   "MICHAEL JEFFERY THOMAS"
## [35] "JOSHUA DAVID LEE"          "SIDDHARTH JHA"
## [37] "AMIYATOSH PWNANANDAM"      "BRIAN LIU"
## [39] "JOEL R HENDON"             "FOREST ZHANG"
## [41] "KYLE WILLIAM MURPHY"       "JARED GE"
## [43] "ROBERT GLEN VASEY"         "JUSTIN D SCHILLING"
## [45] "DEREK YAN"                 "JACOB ALEXANDER LAVALLEY"
## [47] "ERIC WRIGHT"               "DANIEL KHAIN"
## [49] "MICHAEL J MARTIN"          "SHIVAM JHA"
## [51] "TEJAS AYYAGARI"            "ETHAN GUO"
## [53] "JOSE C YBARRA"             "LARRY HODGE"
```

```
## [55] "ALEX KONG"                  "MARISA RICCI"
## [57] "MICHAEL LU"                 "VIRAJ MOHILE"
## [59] "SEAN M MC CORMICK"          "JULIA SHEN"
## [61] "JEZZEL FARKAS"              "ASHWIN BALAJI"
## [63] "THOMAS JOSEPH HOSMER"       "BEN LI"
```

```
print("Total points are: ")
```

```
## [1] "Total points are: "
```

```
print(total_point)
```

```
##  [1] 6.0 6.0 6.0 5.5 5.5 5.0 5.0 5.0 5.0 5.0 4.5 4.5 4.5 4.5 4.5 4.0 4.0 4.0 4.0
## [20] 4.0 4.0 4.0 4.0 4.0 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.0
## [39] 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 2.5 2.5 2.5 2.5 2.5 2.5 2.0 2.0 2.0 2.0 2.0
## [58] 2.0 2.0 1.5 1.5 1.0 1.0 1.0
```

```
player_states=character(0)
## Firs I will select all the rows containg a player's state ON, MI or OH
states_data <- grep("\\b(ON|MI|OH)\\b", text_data, value = TRUE)
##Now I can match player's state and add to a variable
Pre_rating = numeric(0)
for (line in states_data){
 st <- str_extract(line,'ON|MI|OH')
 player_states <- c(player_states, st)

}
print(player_states)
```

```
##  [1] "ON" "MI" "MI" "MI" "MI" "OH" "MI" "MI" "ON" "MI" "MI" "MI" "MI" "MI" "MI"
## [16] "MI" "MI" "MI" "MI" "MI" "ON" "MI" "ON" "MI" "MI" "ON" "MI" "MI" "MI" "ON"
## [31] "MI" "ON" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI"
## [46] "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI"
## [61] "ON" "MI" "MI" "MI"
```

We cann also extract subpart of string without using the loop as follows:

```
rating<-str_extract_all(states_data, ".\\: \\s?[[:digit:]]{3,4}")
rating <- gsub(rating, pattern="R: ", replacement="", fixed = TRUE)
pre_rating <- as.numeric(rating)
print(pre_rating)
```

```
##  [1] 1794 1553 1384 1716 1655 1686 1649 1641 1411 1365 1712 1663 1666 1610 1220
## [16] 1604 1629 1600 1564 1595 1563 1555 1363 1229 1745 1579 1552 1507 1602 1522
## [31] 1494 1441 1449 1399 1438 1355  980 1423 1436 1348 1403 1332 1283 1199 1242
## [46]  377 1362 1382 1291 1056 1011  935 1393 1270 1186 1153 1092  917  853  967
## [61]  955 1530 1175 1163
```

```
text_data <- text_data[-c(0:4)]
text_data<- text_data[sapply(text_data, nchar)>0]
text_data_od <- text_data[c(seq(1, length(text_data), 3))]
```

```
opponent_player <- str_extract_all(text_data_od, "[[:digit:]]{1,2}")

opp_numeric = numeric(0)
for (line in opponent_player){
 players<- line[4: length(line)]
 opp_numeric <- c(opp_numeric, list((players)))

}

print(head(opp_numeric))
```

```
## [[1]]
## [1] "39" "21" "18" "14" "7"  "12" "4"
##
## [[2]]
## [1] "63" "58" "4"  "17" "16" "20" "7"
##
## [[3]]
## [1] "8"  "61" "25" "21" "11" "13" "12"
##
## [[4]]
## [1] "23" "28" "2"  "26" "5"  "19" "1"
##
## [[5]]
## [1] "45" "37" "12" "13" "4"  "14" "17"
##
## [[6]]
## [1] "34" "29" "11" "35" "10" "27" "21"
```

```
opponent_avg_rating<-list()
for (i in 1:length(opp_numeric)){
  opponent_avg_rating[i]<- round(mean(as.numeric(unlist(opp_numeric[i]))), 2)
}
opponent_avg_rating<- unlist(opponent_avg_rating)
opponent_avg_rating
```

```
##  [1] 16.43 26.43 21.57 14.86 20.29 23.86 19.86 21.71 23.29 24.29 24.29 20.33
## [13] 20.43 24.29 30.29 20.40 23.86 22.29 19.29 27.43 25.57 36.00 32.14 40.86
## [25] 25.71 23.14 27.83 19.00 38.00 46.86 40.14 31.57 33.86 31.43 43.29 33.00
## [37] 30.00 20.00 28.57 34.71 38.50 51.43 41.57 36.83 49.71 35.57 29.71 39.20
## [49] 46.20 35.17 39.57 31.00 42.00 42.00 36.83 34.60 37.33 31.83 38.50 35.60
## [61] 35.00 55.00 37.40 39.14
```

```
df<- cbind.data.frame(player_names, player_states, total_point, pre_rating, opponent_avg_rating)
colnames(df)<- c("Player's name", "Player's state", "Total number of points", "Player's Pre-Rating", "Op
head(df)
```

```
##         Player’s name Player’s state Total number of points Player’s Pre-Rating
## 1            GARY HUA             ON                    6.0                1794
## 2     DAKSHESH DARURI             MI                    6.0                1553
## 3        ADITYA BAJAJ             MI                    6.0                1384
## 4 PATRICK H SCHILLING             MI                    5.5                1716
```

```
## 5          HANSHI ZUO             MI                 5.5               1655
## 6         HANSEN SONG             OH                 5.0               1686
##   Opponent's Average Pre-Rating
## 1                         16.43
## 2                         26.43
## 3                         21.57
## 4                         14.86
## 5                         20.29
## 6                         23.86
```

```
write.csv(df, "chess_rating.csv")
```