data 607 project 1

Yina Qiao

2023-02-18

introduction

In this project, I will perform analysis of a text file with chess tournament results where the information has some structure. My job is to create an R Markdown file that generates a .CSV file (that could for example be imported into a SQL database) with the following information for all of the players:

Player's Name, Player's State, Total Number of Points, Player's Pre-Rating, and Average Pre Chess Rating of Opponent

load packages

```
# Load required libraries
library(tidyverse)
```

load data

```
raw_data <- readLines("https://raw.githubusercontent.com/yinaS1234/data-607/main/project1/tournamentinfo.txt")
```

Extra Key Fields into a DataFrame

```
# Extract the required data using regular expressions
player_num <- as.numeric(unlist(str_extract_all(raw_data,"(?<=\\s{3,4})\\d{1,2}(?=\\s)")))
player_name <- unlist(str_extract_all(raw_data,"(?<=\\d\\s\\|\\s)([A-z, -]*\\s){1,}[[:alpha:]]*(?=\\s*\\|)"))
player_state <- unlist(str_extract_all(raw_data, "[[:upper:]]{2}(?=\\s\\|)"))
total_pts <- as.numeric(unlist(str_extract_all(raw_data, "(?<=\\|)\\d\\.\\d")))
player_pre_rat <- as.numeric(unlist(str_extract_all(raw_data, "(?<=R:\\s{1,2})(\\d{3,4}(?=\\s))|(\\d{3,4}(?=P\\d
{1,2}\\s*-))")))

# Take the extracted data and put it into a data frame
processed_data <- data.frame(player_num, player_name, player_state, total_pts, player_pre_rat)

# Check the data frame's structure to make sure it is as intended (i.e. number columns are numeric, character col
umns are character, etc..., and that it has the correct number of rows)
str(processed_data)
```

```
## 'data.frame':    64 obs. of  5 variables:
##  $ player_num    : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ player_name   : chr  "GARY HUA                        " "DAKSHESH DARURI                 " "ADITYA BAJAJ
" "PATRICK H SCHILLING             " ...
##  $ player_state  : chr  "ON" "MI" "MI" "MI" ...
##  $ total_pts     : num  6 6 6 5.5 5.5 5 5 5 5 5 ...
##  $ player_pre_rat: num  1794 1553 1384 1716 1655 ...
```

Extra-Create Opponent Player Number

```
# Had some initial challenges doing this with only regex so to make it simpler and a bit more robust I created a
new list that only had included the relevant rows from the raw data file.
secondary_rows <- raw_data[seq(5, 196, 3)]
opponent_num <- as.numeric(unlist(str_extract_all(secondary_rows, "(?<=\\|(W|L|D)\\s{2,3})[[:digit:]]{1,2}(?=\\|)
|((?<!->)(?<=\\|(U|H|B|X))\\s{4}(?=\\|))")))
```

Average PCR

```
# Create matrix to store data calculated in the for loop.  Pre-populating values with NA for more efficient processing in R.
pcr_matrix <- matrix(data = NA, nrow = 64, ncol = 2)

# Assign readable names for the matrix
colnames(pcr_matrix) <- c("total_opp_pcr", "avg_opp_pcr")

# Initialize a variable to be used as a counter in the for loop to fill the corresponding matrix row
row_counter <- 0

# Start of for loop
for(i in seq(from=1, to=length(opponent_num)-6, by=7)){
  row_counter <- row_counter + 1

# Perform a lookup of each competitor's score based on their player number and add the up for each row (corresponding to each sequence of 7 data points, w/ value from for loop serving as row 'anchor')
  pcr_matrix[row_counter, 1] <- (sum(subset(processed_data$player_pre_rat, processed_data$player_num %in% opponent_num[seq(from=i, to=i+6, by=1)])))

# Calculate the average score for each row, excluding missing entries
  pcr_matrix[row_counter, 2] <- pcr_matrix[row_counter, 1] / length(subset(opponent_num[seq(from=i, to=i+6, by=1)],!is.na(opponent_num[seq(from=i, to=i+6, by=1)])))

}
# End of for loop

# Verify that matrix was processed properly by looking at the first few rows of output
head(pcr_matrix, 5)
```

```
##      total_opp_pcr avg_opp_pcr
## [1,]         11237    1605.286
## [2,]         10285    1469.286
## [3,]         10945    1563.571
## [4,]         11015    1573.571
## [5,]         10506    1500.857
```

```
# Round the figures to the nearest whole number
pcr_matrix[, 2] <- round(pcr_matrix[,2], digits = 0)

# Add average scores to data frame with other processed data and rename for readability
processed_data <- cbind(processed_data, pcr_matrix[, 2])
processed_data <- rename(processed_data, avg_opp_pcr = `pcr_matrix[, 2]`)
```

Export to CSV

```
# Get working directory path
path <- getwd()

# Export file to working directory.  The file.path function has been used to ensure platform independence (i.e. take into account the different path syntaxes for various operating systems)
write.csv(processed_data, file.path(path, "chess_processed_data.csv"))
```

Final Result Glimpse

```
head(processed_data, 5)
```

	player_num	player_name	player_state	total_pts	player_pre_rat	avg_opp_pcr
	<dbl>	<chr>	<chr>	<dbl>	<dbl>	<dbl>
1	1	GARY HUA	ON	6.0	1794	1605
2	2	DAKSHESH DARURI	MI	6.0	1553	1469
3	3	ADITYA BAJAJ	MI	6.0	1384	1564
4	4	PATRICK H SCHILLING	MI	5.5	1716	1574
5	5	HANSHI ZUO	MI	5.5	1655	1501

5 rows