DATA607 Project 1 - Elo

David Simbandumwe

Summary

This project starts with a text file of chess tournament results. This file is processed to develop an R Markown file that generates a .CSV file that includes: + Player's Name, + Player's State, + Total Number of Points, + Player's Pre-Rating, + Average Pre Chess Rating of Opponent

Load Files

```
eloTxt <- read_delim( file = "https://raw.githubusercontent.com/dsimband/DATA607/main/Project1/tourname
         delim = " ",
         col_names = c("rec"),
         trim_ws = TRUE)
```

```
## Rows: 196 Columns: 1
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: " "
## chr (1): rec
```

```
##
## 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 data frame remove extra lines
eloTxt <- eloTxt %>%
  filter(!str_detect(rec,"\\--"))
```

```
## Warning: One or more parsing issues, see `problems()` for details
```

```
# remove heading rows
eloTxt <- eloTxt %>% slice(-c(1,2))
```

Process Odd Rows

```
eloTxt_odd <- eloTxt %>% filter(row_number() %% 2 == 1)


eloTxt_odd <- eloTxt_odd %>%
      separate(rec,
            into = c("id", "name" , "score" , "r1", "r2", "r3", "r4", "r5", "r6", "r7","x" ),
            extra = "merge",
            fill = "left",
            convert = TRUE,
            sep = "\\|"
            ) %>% gather(
              "r1","r2","r3","r4","r5","r6","r7", key = "round", value = "op_data"
      )%>%
      separate(op_data,
            into = c("r_result", "op_id"),
            convert = TRUE
      ) %>%
      drop_na(op_id)
```

Process Even Rows

```
eloTxt_even <- eloTxt %>% filter(row_number() %% 2 == 0)


eloTxt_even <- eloTxt_even %>%
        separate(rec,
              into = c("state", "score_data" , "x" ),
              extra = "merge",
              fill = "left",
              convert = TRUE,
              sep = "\\|") %>%
        mutate(id = row_number()) %>%
        separate(score_data,
              into = c("x1", "x2", "x3", "pre_rating", "post_rating"),
              extra = "merge",
              fill = "left",
              convert = TRUE) %>%
        separate(pre_rating,
              into = c("pre_rating", "x4"),
              extra = "merge",
              fill = "right",
              convert = TRUE,
              sep = "P") %>%
        separate(post_rating,
              into = c("post_rating", "x5"),
              extra = "merge",
              fill = "right",
              convert = TRUE,
              sep = "P") %>%
        select(id,state,pre_rating, post_rating)
```

Join All Rows

```
eloTxt_comb  <- eloTxt_odd %>%
  select (id, name, score, round, r_result, op_id) %>%
  left_join(eloTxt_even, by = "id") %>%
  left_join(eloTxt_even, by = c("op_id" = "id"), suffix = c("", ".op"))
```

Calculate

```
win_df <- eloTxt_comb %>%
  group_by(id,r_result) %>%
  filter(str_detect(r_result, "W")) %>%
  mutate(
    win_elo = sum(pre_rating.op) + n()*400,
    win_num = n()
  ) %>%
  select(id,r_result,win_elo, win_num) %>%
  distinct()

loss_df <- eloTxt_comb %>% group_by(id,r_result) %>%
  filter(str_detect(r_result, "L")) %>%
  mutate(
    loss_elo = sum(pre_rating.op) - n()*400,
    loss_num = n()
  ) %>%
  select(id,r_result,loss_elo, loss_num) %>%
  distinct()
```

create the summary table

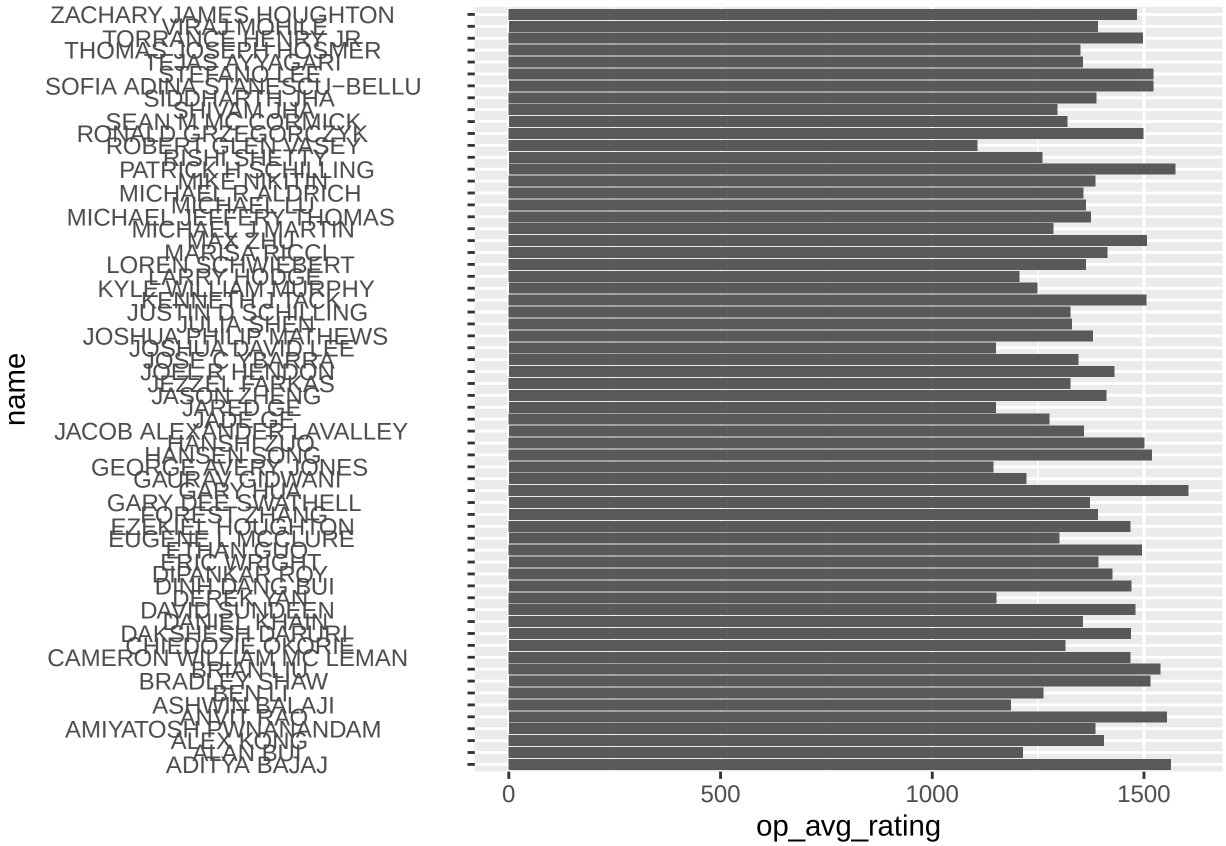
```
eloFinal <- eloTxt_comb %>%
  select (id, name, state, score, round, r_result, op_id, pre_rating, post_rating, pre_rating.op, post_
  left_join(win_df, by = "id") %>%
  left_join(loss_df, by = "id") %>%
  replace_na(list(win_elo = 0, win_num = 0, loss_elo = 0, loss_num = 0 ))

# caclulate average / post elo and filter columns
eloFinal <- eloFinal %>%
    group_by(id) %>%
      mutate(
          op_avg_rating = round(mean(pre_rating.op),0),
          num_games = n(),
          elo_post = round((win_elo + loss_elo) / n(),0)
      ) %>%
    select(id , name, score, state, pre_rating, post_rating, op_avg_rating, num_games, elo_post) %>%
    distinct()
```

Write Files

```
# write the results out to disk
write.csv(eloFinal, "tournamentinfo.csv", row.names=FALSE)
```

Plot

```
# write the results out to disk
ggplot(data=eloFinal, aes(x=name, y=op_avg_rating)) + geom_bar(stat="identity") + coord_flip()
```


ZACHARY JAMES HOUGHTON
VIRAJ MOHILE
TORRANCE HENRY JR
THOMAS JOSEPH HOSMER
TEJAS AYYAGARI
STEFANO LEE
SOFIA ADINA STANESCU−BELLU
SIDDHARTH JHA
SHIVAM JHA
SEAN M MC CORMICK
RONALD GRZEGORCZYK
ROBERT GLEN VASEY
RISHI SHETTY
PATRICK H SCHILLING
MIKE NIKITIN
MICHAEL R ALDRICH
MICHAEL LU
MICHAEL JEFFERY THOMAS
MICHAEL J MARTIN
MAX ZHU
MARISA RICCI
LOREN SCHWIEBERT
LARRY HODGE
KYLE WILLIAM MURPHY
KENNETH J TACK
JUSTIN D SCHILLING
JULIA SHEN
JOSHUA PHILIP MATHEWS
JOSHUA DAVID LEE
JOSE C YBARRA
JOEL R HENDON
JEZZEL FARKAS
JASON ZHENG
JARED GE
JADE GE
JACOB ALEXANDER LAVALLEY
HANSHI ZUO
HANSEN SONG
GEORGE AVERY JONES
GAURAV GIDWANI
GARY HUA
GARY DEE SWATHELL
FOREST ZHANG
EZEKIEL HOUGHTON
EUGENE L MCCLURE
ETHAN GUO
ERIC WRIGHT
DIPANKAR ROY
DINH DANG BUI
DEREK YAN
DAVID SUNDEEN
DANIEL KHAIN
DAKSHESH DARURI
CHIEDOZIE OKORIE
CAMERON WILLIAM MC LEMAN
BRIAN LIU
BRADLEY SHAW
BEN LI
ASHWIN BALAJI
ANVIT RAO
AMIYATOSH PURNANANDAM
ALEX KONG
ALAN BUI
ADITYA BAJAJ
name
op_avg_rating