Extra Credit Assignment #2

Anthony Conrardy

2024-02-17

Assignment Basis

Based on difference in ratings between the chess players and each of their opponents in our Project 1 tournament, calculate each player's expected score (e.g. 4.3) and the difference from their actual score (e.g 4.0). List the five players who most over performed relative to their expected score, and the five players that most under performed relative to their expected score. It should be noted that while the Elo equation will be used here, it will function as a predictor of success for each match played, but will not be individually applied to alter a rating after each match The cited source for the equation used can be found at:

Glickman, M. E., & Jones, A. C. (1999). Rating the chess rating system. CHANCE-BERLIN THEN NEW YORK-, 12, 21-28

Formula use is in the form of:

Ea = $1/(1 + 10)\hat{}(Rb\text{-}Ra)/400$, where Ea is the probability of winning a particular match against opponent b, using the pre-tournament ratings of both player a and b.

The expected score for the tournament then becomes the sum of the probabilities for each match played.

PP = Ea1 + Ea2 + ... Ea7, PP is predicted points.

We will then find the participants where the actual points were greater than predicted (over performed), identify those where the actual points were less than predicted (under performed), and identify the top five (5) in each of those categories.

We will first read in a CSV file that was created during Project 1 that has the necessary information to calculate the top five (5) participants who over performed and the five (5) participants who under performed. We will also clean up the data frame by removing some unnecessary variables for this assignment.

```
extra_credit <- read.csv(file = "C:/Users/para2/Documents/R_Working_Directory/pittsburgh+bridges/Project
extra_credit1 <- extra_credit[, c("player_number", "player_name", "total_points", "player_state", "pre_r
head(extra_credit1,5)
```

```
##   player_number                player_name total_points player_state
## 1             1  GARY HUA                            6.0           ON
## 2             2  DAKSHESH DARURI                     6.0           MI
## 3             3  ADITYA BAJAJ                        6.0           MI
## 4             4  PATRICK H SCHILLING                 5.5           MI
## 5             5  HANSHI ZUO                          5.5           MI
##   pre_rating rnd1_rating rnd2_rating rnd3_rating rnd4_rating rnd5_rating
## 1       1794        1436        1563        1600        1610        1649
## 2       1553        1175         917        1716        1629        1604
## 3       1384        1641         955        1745        1563        1712
## 4       1716        1363        1507        1553        1579        1655
## 5       1655        1242         980        1663        1666        1716
```

```
##   rnd6_rating rnd7_rating
## 1        1663        1716
## 2        1595        1649
## 3        1666        1663
## 4        1564        1794
## 5        1610        1629
```

Creating New Column Variables

Before we begin to calculate the probability of a match being won by a participant, we need to find the difference between their pre-tournament ratings for each match. We will do that by creating a "difference" column for each match played and then divide that number by 400 to match the formula input.

```r
extra_credit1 <-  extra_credit1 |> mutate(rnd1_diff = (rnd1_rating - pre_rating) / 400,
                        rnd2_diff = (rnd2_rating - pre_rating) / 400,
                        rnd3_diff = (rnd3_rating - pre_rating) / 400,
                        rnd4_diff = (rnd4_rating - pre_rating) / 400,
                        rnd5_diff = (rnd5_rating - pre_rating) / 400,
                        rnd6_diff = (rnd6_rating - pre_rating) / 400,
                        rnd7_diff = (rnd7_rating - pre_rating) / 400)
extra_credit1 <- extra_credit1 |> relocate(contains("diff"), .before = pre_rating)
```

We will now complete the calculations necessary to determine the probability of winning for each round.

```r
extra_credit1 <- extra_credit1 |> mutate(power_raise_1 = 10 ^ rnd1_diff,
                        power_raise_2 = 10 ^ rnd2_diff,
                        power_raise_3 = 10 ^ rnd3_diff,
                        power_raise_4 = 10 ^ rnd4_diff,
                        power_raise_5 = 10 ^ rnd5_diff,
                        power_raise_6 = 10 ^ rnd6_diff,
                        power_raise_7= 10 ^ rnd7_diff)
extra_credit1 <- extra_credit1 |> relocate(contains("raise"), .before = rnd1_diff)
```

Here is where we complete the probability of winning each round.

```r
extra_credit1 <-  extra_credit1 |> mutate(prob_1 = 1 / (1 + power_raise_1),
                        prob_2 = 1 / (1 + power_raise_2),
                        prob_3 = 1 / (1 + power_raise_3),
                        prob_4 = 1 / (1 + power_raise_4),
                        prob_5 = 1 / (1 + power_raise_5),
                        prob_6 = 1 / (1 + power_raise_6),
                        prob_7 = 1 / (1 + power_raise_7))
extra_credit1 <- extra_credit1 |> relocate(contains("prob"), .before = power_raise_1)
```

Now we have to conduct a little housekeeping. There were rounds where a participant did not actually play an opponent. This resulted in an NA being passed through all the calculations and now exist in the probability scores. We need to now convert them to zeroes to be summed for the final predicted score, and then add up the probabilities for that predicted score.

```r
mv1 <- ifelse(is.na(extra_credit1$prob_1), 0, extra_credit1$prob_1 )
extra_credit1$prob_1 <- mv1
mv2 <- ifelse(is.na(extra_credit1$prob_2), 0, extra_credit1$prob_2 )
```

```
extra_credit1$prob_2 <- mv2
mv3 <- ifelse(is.na(extra_credit1$prob_3), 0, extra_credit1$prob_3 )
extra_credit1$prob_3 <- mv3
mv4 <- ifelse(is.na(extra_credit1$prob_4), 0, extra_credit1$prob_4 )
extra_credit1$prob_4 <- mv4
mv5 <- ifelse(is.na(extra_credit1$prob_5), 0, extra_credit1$prob_5 )
extra_credit1$prob_5 <- mv5
mv6 <- ifelse(is.na(extra_credit1$prob_6), 0, extra_credit1$prob_6 )
extra_credit1$prob_6 <- mv6
mv7 <- ifelse(is.na(extra_credit1$prob_7), 0, extra_credit1$prob_7 )
extra_credit1$prob_7 <- mv7

extra_credit1 <-  extra_credit1 |> mutate(pred_score = round((prob_1 + prob_2 + prob_3 + prob_4 + prob_5
extra_credit1 <- extra_credit1 |> relocate(contains("pred_score"), .before = prob_1)

extra_credit1 <- extra_credit1 |> mutate(performance = total_points - pred_score)
extra_credit1 <- extra_credit1 |> relocate(contains("player_state"), .before = total_points)
extra_credit1 <- extra_credit1 |> relocate(contains("performance"), .before = prob_1)
```

Results

We can now get the final results needed to answer the questions of who were the participants that over
performed and under during this tournament.

Top 5 Over Performing Participants

```
performance <- extra_credit1[,c("player_number", "player_name", "player_state", "performance", "pred_sco
top_perf <- performance |> arrange(desc(performance))
head(top_perf,5)
```

```
##   player_number                     player_name player_state performance
## 1             3  ADITYA BAJAJ                              MI        4.05
## 2            15  ZACHARY JAMES HOUGHTON                    MI        3.13
## 3            10  ANVIT RAO                                 MI        3.06
## 4            46  JACOB ALEXANDER LAVALLEY                  MI        2.96
## 5            37  AMIYATOSH PWNANANDAM                      MI        2.73
##   pred_score total_points
## 1       1.95          6.0
## 2       1.37          4.5
## 3       1.94          5.0
## 4       0.04          3.0
## 5       0.77          3.5
```

Top 5 Under Performing Participants

```
under_perf <- performance |> arrange(performance)
head(under_perf,5)
```

```
##   player_number                          player_name player_state performance
## 1             25  LOREN SCHWIEBERT                              MI       -2.78
## 2             30  GEORGE AVERY JONES                           ON       -2.52
## 3             42  JARED GE                                      MI       -2.01
## 4             31  RISHI SHETTY                                  MI       -1.59
## 5             35  JOSHUA DAVID LEE                              MI       -1.46
##   pred_score total_points
## 1       6.28          3.5
## 2       6.02          3.5
## 3       5.01          3.0
## 4       5.09          3.5
## 5       4.96          3.5
```