DATA 607 Project 1

Albert Gilharry

February 22, 2018

Contents

Load Libraries

```
library("stringr")
```

Load Input File

```
lines <- scan("data/tournamentinfo.txt", sep="\n", what="raw")
# preview data
print(lines[1:15])
```

```
##  [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    |"
##  [7] "-----------------------------------------------------------------------------------------"
##  [8] "    2 | DAKSHESH DARURI                 |6.0  |W  63|W  58|L   4|W  17|W  16|W  20|W   7|"
##  [9] "   MI | 14598900 / R: 1553   ->1663     |N:2  |B    |W    |B    |W    |B    |W    |B    |"
## [10] "-----------------------------------------------------------------------------------------"
## [11] "    3 | ADITYA BAJAJ                    |6.0  |L   8|W  61|W  25|W  21|W  11|W  13|W  12|"
## [12] "   MI | 14959604 / R: 1384   ->1640     |N:2  |W    |B    |W    |B    |W    |B    |W    |"
## [13] "-----------------------------------------------------------------------------------------"
## [14] "    4 | PATRICK H SCHILLING             |5.5  |W  23|D  28|W   2|W  26|D   5|W  19|D   1|"
## [15] "   MI | 12616049 / R: 1716   ->1744     |N:2  |W    |B    |W    |B    |W    |B    |B    |"
```

Initialize Data Structure

I decided to use use a data frame to store the data and leverage its capabilities to do subsetting, calculations, and generate a CSV.

```
  tournament = data.frame(Player = integer(),
                          Name = character(),
                          State = character(),
                          Points = double(),
                          Pre_Rtg = integer(),
                          Opp_Rtg = double(),
                          Games = integer(),
                          Opponents = character(),
                          stringsAsFactors = FALSE)
```

Processing

This is the meat of the work. Extensive use of R's chaining and nesting capabilities were done to produce a single statement to produce most variables. I created a function to "camel case" the players' names as per the project requirements.

```
# created function to convert names to camel case format
# preventing the need to loop through the segments of the name
camelCase = function(s){
  # converts the first letter of each word to upper case and the other characters to lower case
  return(str_c(toupper(str_sub(s,1,1)),tolower(str_sub(s,2,str_count(s)))))
}

i <- 5 #start at the 5th line
while (i <= length( lines ) ){
   playerInfoRow1 <- str_trim(str_split(str_trim(lines[i]),"\\|")[[1]])
   playerInfoRow2 <- str_trim(str_split(str_trim(lines[i + 1]),"\\|")[[1]])
   playerNumber <- playerInfoRow1[1]
   playerName <- paste(unlist(camelCase(str_split(playerInfoRow1[2],"\\s")[[1]])), collapse = " ")
   playerState <- playerInfoRow2[1]
   points <- playerInfoRow1[3]
   playerRatings= str_trim(str_split(str_split(playerInfoRow2[2] ,":")[[1]][2],"->")[[1]])
   playerPreRating <- str_split(playerRatings[1],"P")[[1]][1]
   opponents= str_extract_all(str_sub(lines[i],-43),"[0-9]+")[[1]] #used as reference calc. average
   gamesPlayed = length(opponents)
   opponents=paste(unlist(opponents),collapse = "|")
   tournament[nrow(tournament) + 1,] = list(playerNumber,
                                            playerName,
                                            playerState,
                                            points,
                                            playerPreRating,
                                            0.0,
                                            gamesPlayed,
                                            opponents)

   # increment in blocks of 3 to prevent unnecessary processing of dashes
   i <- i + 3
}
```

Preview Processed Data

We now all the necessary information to perform necessary calculations and analysis.

```r
print(subset(tournament, select = c("Name","State","Points","Pre_Rtg","Opponents")))
```

```
##                         Name State Points Pre_Rtg           Opponents
## 1                   Gary Hua    ON    6.0    1794   39|21|18|14|7|12|4
## 2             Dakshesh Daruri    MI    6.0    1553   63|58|4|17|16|20|7
## 3                Aditya Bajaj    MI    6.0    1384  8|61|25|21|11|13|12
## 4         Patrick H Schilling    MI    5.5    1716    23|28|2|26|5|19|1
## 5                  Hanshi Zuo    MI    5.5    1655  45|37|12|13|4|14|17
## 6                Hansen Song    OH    5.0    1686 34|29|11|35|10|27|21
## 7           Gary Dee Swathell    MI    5.0    1649    57|46|13|11|1|9|2
## 8            Ezekiel Houghton    MI    5.0    1641   3|32|14|9|47|28|19
## 9                 Stefano Lee    ON    5.0    1411   25|18|59|8|26|7|20
## 10                 Anvit Rao    MI    5.0    1365  16|19|55|31|6|25|18
## 11   Cameron William Mc Leman    MI    4.5    1712    38|56|6|7|3|34|26
## 12             Kenneth J Tack    MI    4.5    1663       42|33|5|38|1|3
## 13          Torrance Henry Jr    MI    4.5    1666    36|27|7|5|33|3|32
## 14               Bradley Shaw    MI    4.5    1610    54|44|8|1|27|5|31
## 15     Zachary James Houghton    MI    4.5    1220 19|16|30|22|54|33|38
## 16               Mike Nikitin    MI    4.0    1604        10|15|39|2|36
## 17         Ronald Grzegorczyk    MI    4.0    1629   48|41|26|2|23|22|5
## 18              David Sundeen    MI    4.0    1600   47|9|1|32|19|38|10
## 19               Dipankar Roy    MI    4.0    1564   15|10|52|28|18|4|8
## 20                Jason Zheng    MI    4.0    1595   40|49|23|41|28|2|9
## 21              Dinh Dang Bui    ON    4.0    1563    43|1|47|3|40|39|6
## 22           Eugene L Mcclure    MI    4.0    1555    64|52|28|15|17|40
## 23                  Alan Bui    ON    4.0    1363  4|43|20|58|17|37|46
## 24          Michael R Aldrich    MI    4.0    1229 28|47|43|25|60|44|39
## 25           Loren Schwiebert    MI    3.5    1745   9|53|3|24|34|10|47
## 26                  Max Zhu    ON    3.5    1579   49|40|17|4|9|32|11
## 27             Gaurav Gidwani    MI    3.5    1552     51|13|46|37|14|6
## 28 Sofia Adina Stanescu-bellu    MI    3.5    1507   24|4|22|19|20|8|36
## 29           Chiedozie Okorie    MI    3.5    1602     50|6|38|34|52|48
## 30         George Avery Jones    ON    3.5    1522 52|64|15|55|31|61|50
## 31               Rishi Shetty    MI    3.5    1494 58|55|64|10|30|50|14
## 32      Joshua Philip Mathews    ON    3.5    1441  61|8|44|18|51|26|13
## 33                  Jade Ge    MI    3.5    1449 60|12|50|36|13|15|51
## 34    Michael Jeffery Thomas    MI    3.5    1399  6|60|37|29|25|11|52
## 35           Joshua David Lee    MI    3.5    1438  46|38|56|6|57|52|48
## 36              Siddharth Jha    MI    3.5    1355    13|57|51|33|16|28
## 37       Amiyatosh Pwnanandam    MI    3.5     980        5|34|27|23|61
## 38                 Brian Liu    MI    3.0    1423    11|35|29|12|18|15
## 39              Joel R Hendon    MI    3.0    1436  1|54|40|16|44|21|24
## 40              Forest Zhang    MI    3.0    1348 20|26|39|59|21|56|22
## 41        Kyle William Murphy    MI    3.0    1403          59|17|58|20
## 42                  Jared Ge    MI    3.0    1332 12|50|57|60|61|64|56
## 43          Robert Glen Vasey    MI    3.0    1283 21|23|24|63|59|46|55
## 44         Justin D Schilling    MI    3.0    1199    14|32|53|39|24|59
## 45                  Derek Yan    MI    3.0    1242  5|51|60|56|63|55|58
## 46   Jacob Alexander Lavalley    MI    3.0     377  35|7|27|50|64|43|23
## 47                Eric Wright    MI    2.5    1362  18|24|21|61|8|51|25
## 48              Daniel Khain    MI    2.5    1382       17|63|52|29|35
## 49           Michael J Martin    MI    2.5    1291       26|20|63|64|58
## 50                 Shivam Jha    MI    2.5    1056    29|42|33|46|31|30
## 51            Tejas Ayyagari    MI    2.5    1011 27|45|36|57|32|47|33
```

```
## 52                  Ethan Guo    MI    2.5     935 30|22|19|48|29|35|34
## 53              Jose C Ybarra    MI    2.0    1393             25|44|57
## 54                Larry Hodge    MI    2.0    1270    14|39|61|15|59|64
## 55                  Alex Kong    MI    2.0    1186    62|31|10|30|45|43
## 56                Marisa Ricci    MI    2.0    1153       11|35|45|40|42
## 57                  Michael Lu    MI    2.0    1092     7|36|42|51|35|53
## 58                Viraj Mohile    MI    2.0     917     31|2|41|23|49|45
## 59          Sean M Mc Cormick    MI    2.0     853     41|9|40|43|54|44
## 60                  Julia Shen    MI    1.5     967       33|34|45|42|24
## 61              Jezzel Farkas    ON    1.5     955  32|3|54|47|42|30|37
## 62              Ashwin Balaji    MI    1.0    1530                   55
## 63       Thomas Joseph Hosmer    MI    1.0    1175        2|48|49|43|45
## 64                     Ben Li    MI    1.0    1163 22|30|31|49|46|42|54
```

Calculate Opponents Pre Chess Rating by Subsetting

This was done by referencing the players opponents by creating a list and using the *%in%* operator as input to the mean function.

```
i <- 1
while(i <= nrow(tournament)){
  oppMean <- mean(as.integer(subset(tournament,
                                    Player %in% as.integer(str_split(tournament$Opponents[i],
                                                                     "\\|")[[1]])),select=c("Pre_
  tournament$Opp_Rtg[i] <- oppMean
  i <- i + 1
}
```

View Opponents Pre Chess Rating

```
  print(subset(tournament, select = c("Name","State","Points","Pre_Rtg","Opp_Rtg")))
```

```
##                          Name State Points Pre_Rtg  Opp_Rtg
## 1                    Gary Hua    ON    6.0    1794 1605.286
## 2             Dakshesh Daruri    MI    6.0    1553 1469.286
## 3                Aditya Bajaj    MI    6.0    1384 1563.571
## 4         Patrick H Schilling    MI    5.5    1716 1573.571
## 5                  Hanshi Zuo    MI    5.5    1655 1500.857
## 6                Hansen Song    OH    5.0    1686 1518.714
## 7           Gary Dee Swathell    MI    5.0    1649 1372.143
## 8            Ezekiel Houghton    MI    5.0    1641 1468.429
## 9                 Stefano Lee    ON    5.0    1411 1523.143
## 10                  Anvit Rao    MI    5.0    1365 1554.143
## 11   Cameron William Mc Leman    MI    4.5    1712 1467.571
## 12             Kenneth J Tack    MI    4.5    1663 1506.167
## 13          Torrance Henry Jr    MI    4.5    1666 1497.857
## 14               Bradley Shaw    MI    4.5    1610 1515.000
## 15     Zachary James Houghton    MI    4.5    1220 1483.857
## 16               Mike Nikitin    MI    4.0    1604 1385.800
## 17         Ronald Grzegorczyk    MI    4.0    1629 1498.571
## 18              David Sundeen    MI    4.0    1600 1480.000
## 19               Dipankar Roy    MI    4.0    1564 1426.286
```

```
## 20                Jason Zheng    MI    4.0    1595 1410.857
## 21              Dinh Dang Bui    ON    4.0    1563 1470.429
## 22           Eugene L Mcclure    MI    4.0    1555 1300.333
## 23                   Alan Bui    ON    4.0    1363 1213.857
## 24          Michael R Aldrich    MI    4.0    1229 1357.000
## 25           Loren Schwiebert    MI    3.5    1745 1363.286
## 26                    Max Zhu    ON    3.5    1579 1506.857
## 27              Gaurav Gidwani    MI    3.5    1552 1221.667
## 28 Sofia Adina Stanescu-bellu    MI    3.5    1507 1522.143
## 29            Chiedozie Okorie    MI    3.5    1602 1313.500
## 30         George Avery Jones    ON    3.5    1522 1144.143
## 31               Rishi Shetty    MI    3.5    1494 1259.857
## 32      Joshua Philip Mathews    ON    3.5    1441 1378.714
## 33                    Jade Ge    MI    3.5    1449 1276.857
## 34     Michael Jeffery Thomas    MI    3.5    1399 1375.286
## 35           Joshua David Lee    MI    3.5    1438 1149.714
## 36              Siddharth Jha    MI    3.5    1355 1388.167
## 37       Amiyatosh Pwnanandam    MI    3.5     980 1384.800
## 38                  Brian Liu    MI    3.0    1423 1539.167
## 39              Joel R Hendon    MI    3.0    1436 1429.571
## 40               Forest Zhang    MI    3.0    1348 1390.571
## 41        Kyle William Murphy    MI    3.0    1403 1248.500
## 42                    Jared Ge    MI    3.0    1332 1149.857
## 43          Robert Glen Vasey    MI    3.0    1283 1106.571
## 44         Justin D Schilling    MI    3.0    1199 1327.000
## 45                  Derek Yan    MI    3.0    1242 1152.000
## 46   Jacob Alexander Lavalley    MI    3.0     377 1357.714
## 47                Eric Wright    MI    2.5    1362 1392.000
## 48               Daniel Khain    MI    2.5    1382 1355.800
## 49           Michael J Martin    MI    2.5    1291 1285.800
## 50                 Shivam Jha    MI    2.5    1056 1296.000
## 51             Tejas Ayyagari    MI    2.5    1011 1356.143
## 52                  Ethan Guo    MI    2.5     935 1494.571
## 53              Jose C Ybarra    MI    2.0    1393 1345.333
## 54                Larry Hodge    MI    2.0    1270 1206.167
## 55                  Alex Kong    MI    2.0    1186 1406.000
## 56               Marisa Ricci    MI    2.0    1153 1414.400
## 57                 Michael Lu    MI    2.0    1092 1363.000
## 58               Viraj Mohile    MI    2.0     917 1391.000
## 59          Sean M Mc Cormick    MI    2.0     853 1319.000
## 60                 Julia Shen    MI    1.5     967 1330.200
## 61              Jezzel Farkas    ON    1.5     955 1327.286
## 62              Ashwin Balaji    MI    1.0    1530 1186.000
## 63       Thomas Joseph Hosmer    MI    1.0    1175 1350.200
## 64                    Ben Li    MI    1.0    1163 1263.000
```

Write CSV File

Only the necessary variables are written to the CSV file.

```
write.csv(subset(tournament, select = c("Name","State","Points","Pre_Rtg","Opp_Rtg")),
          file = "tournament.csv", quote = FALSE, row.names = FALSE)
```