Project 1

Dan Brooks

February 20, 2016

Import Information From Excel File

```
library(RCurl)
```

```
## Loading required package: bitops
```

```
library(stringr)
library(plotly)
```

```
## Loading required package: ggplot2

##
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':
##
##     last_plot

## The following object is masked from 'package:graphics':
##
##     layout
```

```
URL <- getURL("https://raw.githubusercontent.com/DanielBrooks39/IS607/master/Chess_Tournament_Info.csv")
Data <- read.csv(text = URL, header = FALSE)

names(Data) <- c("Position", "Name", "Points", "RD1", "RD2", "RD3", "RD4", "RD5", "RD6", "RD7")
```

Extract Infromation (Position, State, Name, Pre-Rating, Round Information)

- Position
  - Extract the position of the competitor in the tournament (1-64)
- State
  - Extracts the state where the competitor is from
- Name
  - Gets the full name of the participant

- Pre-Rating
  - Gets the rating of the participant beofre the start of the tournament
- Firstrd
  - The result of the first round of the tournament
  - This is repeated for all seven rounds
- OverallRate
  - The average rate of all of opponents that the participant faced
  - Total of all the Pre-rating the paticipant faced divided by the total number of rounds played
- Points
  - The total number of points the participant received
  - W - win (1 point)
  - L - Lose (0 points)
  - D - Draw (.5 points)
  - H - half point bye (.5 points)
  - X - Win by forfeit (opp. failed to appear, 1 point)
  - U - unplayed game (player withdrew, 0 point)
  - B - full point bye (1 point)

```
position <- unlist(str_extract_all(Data$Position, "[[:digit:]]{1,2}"))
position
```

```
##  [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10" "11" "12" "13" "14"
## [15] "15" "16" "17" "18" "19" "20" "21" "22" "23" "24" "25" "26" "27" "28"
## [29] "29" "30" "31" "32" "33" "34" "35" "36" "37" "38" "39" "40" "41" "42"
## [43] "43" "44" "45" "46" "47" "48" "49" "50" "51" "52" "53" "54" "55" "56"
## [57] "57" "58" "59" "60" "61" "62" "63" "64"
```

```
state <- unlist(str_extract_all(Data$Position, "[A-Z]{2}"))
state
```

```
##  [1] "ON" "MI" "MI" "MI" "MI" "OH" "MI" "MI" "ON" "MI" "MI" "MI" "MI" "MI"
## [15] "MI" "MI" "MI" "MI" "MI" "MI" "ON" "MI" "ON" "MI" "MI" "ON" "MI" "MI"
## [29] "MI" "ON" "MI" "ON" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI"
## [43] "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI"
## [57] "MI" "MI" "MI" "MI" "ON" "MI" "MI" "MI"
```

```
name <- unlist(str_extract_all(Data$Name, "[A-Z].+[A-Z][A-Z]"))
head(name)
```

```
## [1] "GARY HUA"            "DAKSHESH DARURI"     "ADITYA BAJAJ"
## [4] "PATRICK H SCHILLING" "HANSHI ZUO"          "HANSEN SONG"
```

```
tail(name)
```

```
## [1] "SEAN M MC CORMICK"    "JULIA SHEN"           "JEZZEL FARKAS"
## [4] "ASHWIN BALAJI"        "THOMAS JOSEPH HOSMER" "BEN LI"
```

```
prerating <- unlist(str_extract_all(Data$Name, "[[:space:]][[:digit:]]{3,4}"))
prerating
```

```
##  [1] " 1794" " 1553" " 1384" " 1716" " 1655" " 1686" " 1649" " 1641"
##  [9] " 1411" " 1365" " 1712" " 1663" " 1666" " 1610" " 1220" " 1604"
## [17] " 1629" " 1600" " 1564" " 1595" " 1563" " 1555" " 1363" " 1229"
## [25] " 1745" " 1579" " 1552" " 1507" " 1602" " 1522" " 1494" " 1441"
## [33] " 1449" " 1399" " 1438" " 1355" " 980"  " 1423" " 1436" " 1348"
## [41] " 1403" " 1332" " 1283" " 1199" " 1242" " 377"  " 1362" " 1382"
## [49] " 1291" " 1056" " 1011" " 935"  " 1393" " 1270" " 1186" " 1153"
## [57] " 1092" " 917"  " 853"  " 967"  " 955"  " 1530" " 1175" " 1163"
```

```
firstrd <- unlist(str_extract_all(Data$RD1, "[W][[:space:]][[:space:]][[:digit:]][[:digit:]]*|[D][[:spa
secondrd <- unlist(str_extract_all(Data$RD2, "[W][[:space:]][[:space:]][[:digit:]][[:digit:]]*|[D][[:sp
thirdrd <- unlist(str_extract_all(Data$RD3, "[W][[:space:]][[:space:]][[:digit:]][[:digit:]]*|[D][[:spa
fourthrd <- unlist(str_extract_all(Data$RD4, "[W][[:space:]][[:space:]][[:digit:]][[:digit:]]*|[D][[:sp
fifthrd <- unlist(str_extract_all(Data$RD5, "[W][[:space:]][[:space:]][[:digit:]][[:digit:]]*|[D][[:spa
sixthrd <- unlist(str_extract_all(Data$RD6, "[W][[:space:]][[:space:]][[:digit:]][[:digit:]]*|[D][[:spa
seventhrd <- unlist(str_extract_all(Data$RD7, "[W][[:space:]][[:space:]][[:digit:]][[:digit:]]*|[D][[:s

overallrate <- c(1:64)
points <- (1:64)

Chessdata <- cbind.data.frame(name, state, points, prerating, overallrate, firstrd, secondrd, thirdrd, 
names(Chessdata) <- c("Name", "State", "Points", "Pre_Rating", "Average_Opp_Rate", "RD1", "RD2", "RD3",

head(Chessdata)
```

```
##                  Name State Points Pre_Rating Average_Opp_Rate   RD1   RD2
## 1            GARY HUA    ON      1       1794               1 W  39 W  21
## 2     DAKSHESH DARURI    MI      2       1553               2 W  63 W  58
## 3        ADITYA BAJAJ    MI      3       1384               3  L  8 W  61
## 4 PATRICK H SCHILLING    MI      4       1716               4 W  23 D  28
## 5          HANSHI ZUO    MI      5       1655               5 W  45 W  37
## 6         HANSEN SONG    OH      6       1686               6 W  34 D  29
##     RD3   RD4   RD5   RD6   RD7
## 1 W  18 W  14  W  7 D  12  D  4
## 2  L  4 W  17 W  16 W  20  W  7
## 3 W  25 W  21 W  11 W  13 W  12
## 4  W  2 W  26  D  5 W  19  D  1
## 5 D  12 D  13  D  4 W  14 W  17
## 6 L  11 W  35 D  10 W  27 W  21
```

```
tail(Chessdata)
```

```
##                    Name State Points Pre_Rating Average_Opp_Rate   RD1
## 59    SEAN M MC CORMICK    MI     59        853               59 L  41
## 60           JULIA SHEN    MI     60        967               60 L  33
## 61        JEZZEL FARKAS    ON     61        955               61 L  32
## 62        ASHWIN BALAJI    MI     62       1530               62 W  55
## 63 THOMAS JOSEPH HOSMER    MI     63       1175               63  L  2
## 64               BEN LI    MI     64       1163               64 L  22
```

```
##      RD2   RD3   RD4   RD5   RD6   RD7
## 59     B  L  9 L  40 L  43 W  54 L  44
## 60 L  34 D  45 D  42 L  24     H     U
## 61  L  3 W  54 L  47 D  42 L  30 L  37
## 62     U     U     U     U     U     U
## 63 L  48 D  49 L  43 L  45     H     U
## 64 D  30 L  31 D  49 L  46 L  42 L  54
```

Function to Calculate the Total Points the Player Received

- This functions will calculate the total number of points the participant received over all 7 rounds of the tournament
- This functions takes in 8 variables
  - points (the vector that will hold the total number of points for each participant)
  - one (result for the first round the participant played)
  - two (result for the second round the participant plated)
  - etc ....
- This function goes through and extracts the letter result for each round the participant played
- It then assigns the number of points associated with the letter to each round
- It will then add up the points for each round
- It will move the total number of points to the overall data frame (Chessdata$Points)

```
pointcalc <- function(points, one, two, three, four, five, six, seven)
{
  index <- 1
  while (index <= 64)
{
     total <- 0
    if (str_extract(one[index], "[[:alpha:]]") == "W"|str_extract(one[index], "[[:alpha:]]") == "B"|str
  {
    total <- total + 1
  }
    if (str_extract(two[index], "[[:alpha:]]") == "W"|str_extract(two[index], "[[:alpha:]]") == "B"|str
  {
    total <- total + 1
  }
    if (str_extract(three[index], "[[:alpha:]]") == "W"|str_extract(three[index], "[[:alpha:]]") == "B"
  {
    total <- total + 1
  }
    if (str_extract(four[index], "[[:alpha:]]") == "W"|str_extract(four[index], "[[:alpha:]]") == "B"|s
  {
    total <- total + 1
  }
    if (str_extract(five[index], "[[:alpha:]]") == "W"|str_extract(five[index], "[[:alpha:]]") == "B"|s
  {
    total <- total + 1
  }
```

```
    if (str_extract(six[index], "[[:alpha:]]") == "W"|str_extract(six[index], "[[:alpha:]]") == "B"|str_
  {
    total <- total + 1
  }
    if (str_extract(seven[index], "[[:alpha:]]") == "W"|str_extract(seven[index], "[[:alpha:]]") == "B"
  {
    total <- total + 1
  }
     if (str_extract(one[index], "[[:alpha:]]") == "D"|str_extract(one[index], "[[:alpha:]]") == "H")
  {
    total <- total + .5
  }
    if (str_extract(two[index], "[[:alpha:]]") == "D"|str_extract(two[index], "[[:alpha:]]") == "H")
  {
    total <- total + .5
  }
    if (str_extract(three[index], "[[:alpha:]]") == "D"|str_extract(three[index], "[[:alpha:]]") == "H"
  {
    total <- total + .5
  }
    if (str_extract(four[index], "[[:alpha:]]") == "D"|str_extract(four[index], "[[:alpha:]]") == "H")
  {
    total <- total + .5
  }
    if (str_extract(five[index], "[[:alpha:]]") == "D"|str_extract(five[index], "[[:alpha:]]") == "H")
  {
    total <- total + .5
  }
    if (str_extract(six[index], "[[:alpha:]]") == "D"|str_extract(six[index], "[[:alpha:]]") == "H")
  {
    total <- total + .5
  }
    if (str_extract(seven[index], "[[:alpha:]]") == "D"|str_extract(seven[index], "[[:alpha:]]") == "H"
  {
    total <- total + .5
  }
  points[index] <- total
  index <-index + 1
  }
  return(points)
}

Chessdata$Points <- pointcalc(points, firstrd, secondrd, thirdrd, fourthrd, fifthrd, sixthrd, seventhrd

head(Chessdata)
```

```
##                  Name State Points Pre_Rating Average_Opp_Rate   RD1   RD2
## 1            GARY HUA    ON    6.0       1794                1 W  39 W  21
## 2     DAKSHESH DARURI    MI    6.0       1553                2 W  63 W  58
## 3        ADITYA BAJAJ    MI    6.0       1384                3  L  8 W  61
## 4 PATRICK H SCHILLING    MI    5.5       1716                4 W  23 D  28
## 5          HANSHI ZUO    MI    5.5       1655                5 W  45 W  37
## 6         HANSEN SONG    OH    5.0       1686                6 W  34 D  29
```

```
##      RD3   RD4   RD5   RD6   RD7
## 1 W  18 W  14  W  7 D  12  D  4
## 2  L  4 W  17 W  16 W  20  W  7
## 3 W  25 W  21 W  11 W  13 W  12
## 4  W  2 W  26  D  5 W  19  D  1
## 5 D  12 D  13  D  4 W  14 W  17
## 6 L  11 W  35 D  10 W  27 W  21
```

Function to Calculate the Average Rating of Opponent

- This function will calculate the overall average rating of the opponents the participant faced throughout the tournamnet
- This function takes in 9 variables
  - It will take in the 7 rounds of result data
  - It will take in the Pre-Rating or each participant in the tournament
  - A place holder for the Average Opponent rating for the entire tournamnet
- It intialized the counts for each individual round. This will be used to total up the amount of opponents the participant actually faced. The count will increase by one only when the participant actually played faced somebody. It will not go up if there is a forefeit or bye.
- It will initialize total total number of ratings the participant faced durig thr tournament. This follows the same pattern as the counts. It will only increase if the participant actuall faces an opponent. It will not increase if there was a bye or forfeit.
- It will read in the number of opponent. That number will be used to search the index of the player pre-rating vector. That index will take the rating of the opponent (if it exists) and add tot the total. It will also increase the count by one. It will do that for every round for that player. It will then add up the ratings for each round and divide it by the total number of players faced. That will give the overall player rating each participant faced.
- It then places that calculated value into the overall dta frame (Chessdata$Average_Opp_Rate)

```
calcrating <- function (one, two, three, four, five, six, seven, prating, avgrating)
{
  index <- 1
  while (index <= 64)
  {
    total1 <- 0
    total2 <- 0
    total3 <- 0
    total4 <- 0
    total5 <- 0
    total6 <- 0
    total7 <- 0
    count1 <- 0
    count2 <- 0
    count3 <- 0
    count4 <- 0
    count5 <- 0
    count6 <- 0
    count7 <- 0
```

```
overalltotal <- 0
overallcount <- 0
opponentnum1 <- str_extract(one[index], "[[:digit:]][[:digit:]]*")
num <- as.numeric(opponentnum1)
if (!is.na(num))
{
value <- as.numeric(prating[num])
total1 <- total1 + value
count1 <- count1 + 1
}
opponentnum2 <- str_extract(two[index], "[[:digit:]][[:digit:]]*")
num <- as.numeric(opponentnum2)
if (!is.na(num))
{
value <- as.numeric(prating[num])
total2 <- total2 + value
count2 <- count2 + 1
}
opponentnum3 <- str_extract(three[index], "[[:digit:]][[:digit:]]*")
num <- as.numeric(opponentnum3)
if (!is.na(num))
{
value <- as.numeric(prating[num])
total3 <- total3 + value
count3 <- count3 + 1
}
opponentnum4 <- str_extract(four[index], "[[:digit:]][[:digit:]]*")
num <- as.numeric(opponentnum4)
if (!is.na(num))
{
value <- as.numeric(prating[num])
total4 <- total4 + value
count4 <- count4 + 1
}
opponentnum5 <- str_extract(five[index], "[[:digit:]][[:digit:]]*")
num <- as.numeric(opponentnum5)
if (!is.na(num))
{
value <- as.numeric(prating[num])
total5 <- total5 + value
count5 <- count5 + 1
}
opponentnum6 <- str_extract(six[index], "[[:digit:]][[:digit:]]*")
num <- as.numeric(opponentnum6)
if (!is.na(num))
{
value <- as.numeric(prating[num])
total6 <- total6 + value
count6 <- count6 + 1
}
opponentnum7 <- str_extract(seven[index], "[[:digit:]][[:digit:]]*")
num <- as.numeric(opponentnum7)
if (!is.na(num))
```

```
    {
    value <- as.numeric(prating[num])
    total7 <- total7 + value
    count7 <- count7 + 1
    }
    overalltotal <- total1 + total2 + total3 + total4 + total5 + total6 + total7
    overallcount <- count1 + count2 + count3 + count4 + count5 + count6 + count7
  if (overallcount == 0)
  {
    overallcount <- 1
  }
   avgrating[index] <- overalltotal/overallcount
   index <-index + 1
  }
  return(avgrating)
}

Chessdata$Average_Opp_Rate <- calcrating(firstrd, secondrd, thirdrd, fourthrd, fifthrd, sixthrd, seventh
head(Chessdata)
```

```
##                  Name State Points Pre_Rating Average_Opp_Rate   RD1   RD2
## 1           GARY HUA    ON    6.0       1794         1605.286 W  39 W  21
## 2     DAKSHESH DARURI    MI    6.0       1553         1469.286 W  63 W  58
## 3        ADITYA BAJAJ    MI    6.0       1384         1563.571  L  8 W  61
## 4 PATRICK H SCHILLING    MI    5.5       1716         1573.571 W  23 D  28
## 5          HANSHI ZUO    MI    5.5       1655         1500.857 W  45 W  37
## 6         HANSEN SONG    OH    5.0       1686         1518.714 W  34 D  29
##     RD3   RD4   RD5   RD6   RD7
## 1 W  18 W  14  W  7 D  12  D  4
## 2  L  4 W  17 W  16 W  20  W  7
## 3 W  25 W  21 W  11 W  13 W  12
## 4  W  2 W  26  D  5 W  19  D  1
## 5 D  12 D  13  D  4 W  14 W  17
## 6 L  11 W  35 D  10 W  27 W  21
```

Write to a CSV File

```
write.csv(Chessdata, file = "Project1.csv")
```

Create a plot

- This plot will show the comparison between the points a participant earned, their pre-rating, and their average opponentrating they faced

```
plot_ly(data = Chessdata, x = Points, y = Pre_Rating, z = Average_Opp_Rate, mode = "markers")
```