

# Week3\_Project1\_Chess

*Shyam BV*

*September 17, 2016*

In this project, you're given a text file with chess tournament results where the information has some spaces. Your job is to create an R Markdown file that generates a .CSV file (that could for example be imported into Excel) 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. For the first player, the information would be:

Gary Hua, ON, 6.0, 1794, 1605

1605 was calculated by using the pre-tournament opponents' ratings of 1436, 1563, 1600, 1610, 1649, 1663, and dividing by the total number of games played.

If you have questions about the meaning of the data or the results, please post them on the discussion board. Chess, like science, is a game of back and forth...

The chess rating system (invented by a Minnesota statistician named Arpad Elo) has been used in many other contexts including assessing relative strength of employment candidates by human resource departments.

## Import necessary Library

```
library(stringr)
```

## Load the text file data into a dataframe and combine the columns

```
chess_data <- read.table("C:/CUNY/Courses/CUNY-repository/607/Project1/Data/tournamentinfo.txt",header=1)

combined_data <- paste (chess_data[seq(5,196,3),], chess_data[seq(6,196,3),1],sep = "")
```

## Fetch the location of | and split the data into list. Convert that list to dataframe

```
str_location <- str_locate_all(combined_data,"\\|")

# create a list and its size

datalist <- vector("list",NROW(combined_data))

#Loop through the datalist for all the rows trim the spaces

for(i in 1:NROW(combined_data)) {

  for(j in 1:20) {
```

```

datalist[[i]][j] <- str_trim(substr(combined_data[i],str_location[[i]][j],str_location[[i]][j+1]),"both")
}
}

# convert list to dataframe

csvdf <- do.call(rbind.data.frame, datalist)

#Set temp names for the dataframe

csvdf <- setNames(csvdf,c(1:20))

```

Replace all the | using regular expression with help of sapply

```

#Dataframe without | and it has

without_symbols_df <- data.frame(sapply(csvdf,function(x){
  x <- str_replace(x,"\\\\\\\\s?", "")
  x <- str_replace(x,"\\\\s+?\\\\\\\\", "")
  x <- str_replace(x,"\\\\\\\\|", "")
  x <- str_trim(x,"both")
  return(x)
}))

#Convert the full dataset dataframe to the required columns dataframe

avg_df <- data.frame(without_symbols_df$X1,without_symbols_df$X2,str_extract(without_symbols_df$X3,"\\\\d+"))

#Set correct names for all the fields

avg_df <- setNames(avg_df,c("Player","Points","Play1","Play2","Play3","Play4","Play5","Play6","Play7","Play8","Play9","Play10"))

#Set 0 for all the NA

avg_df[is.na(avg_df)] <- 0

```

```

#Replace the the opponent number with their previous score

i <- 0
j <- 0
avg_df1 <- avg_df
for(i in 1:nrow(avg_df1)) {
  for(j in 3:9) {

avg_df1[i,j] <- avg_df1[avg_df1[i,j],10]

  }
}

```

```

#Set 0 for all NA

avg_df1[is.na(avg_df1)] <- 0

#Convert Char columns to numeric

columns <- c(3:10)
avg_df1[,columns] <- apply(avg_df1[,columns],2,function(x){ as.numeric(as.character(x))})

avg_df1[is.na(avg_df1)] <- 0

#Finally find out the average

avg_df1$Sum <- rowMeans(avg_df1[,3:9],1)

#Remove unwanted fields

remove(chess_data, csvdf, without_symbols_df, columns, combined_data, datalist, i, j, str_location, avg_df)

head(avg_df1)

```

```

##           Player Points Play1 Play2 Play3 Play4 Play5 Play6 Play7
## 1          GARY HUA    6.0  1436  1563  1600  1610  1649  1663  1716
## 2      DAKSHESH DARURI    6.0  1175   917  1716  1629  1604  1595  1649
## 3        ADITYA BAJAJ    6.0  1641   955  1745  1563  1712  1666  1663
## 4 PATRICK H SCHILLING    5.5  1363  1507  1553  1579  1655  1564  1794
## 5          HANSHI ZUO    5.5  1242   980  1663  1666  1716  1610  1629
## 6          HANSEN SONG    5.0  1399  1602  1712  1438  1365  1552  1563
## Previous_points      Sum
## 1          1794 1605.286
## 2          1553 1469.286
## 3          1384 1563.571
## 4          1716 1573.571
## 5          1655 1500.857
## 6          1686 1518.714

```