

*Alejandro Osborne*

```
library(stringi)
library(stringr)

## Warning: package 'stringr' was built under R version 3.4.3

library(ggplot2)
library(DT)

## Warning: package 'DT' was built under R version 3.4.3
```

```
raw <- "https://raw.githubusercontent.com/adcosborne/DATA-607/master/tournamentinfo.txt"
newfile <- "tournamentinfo.txt"
downloader::download(raw, newfile)
dwnfile <- file(newfile, open = "r")
tourney <- readLines(dwnfile, warn = FALSE)
head(tourney, 10)
```

```
## [1] "-----"
## [2] " Pair | Player Name | Total | Round | Round | Round | Round | Round | Round"
## [3] " Num | USCF ID / Rtg (Pre->Post) | Pts | 1 | 2 | 3 | 4 | 5 | 6"
## [4] "-----"
## [5] " 1 | GARY HUA | 6.0 | W 39 | W 21 | W 18 | W 14 | W 7 | D 1"
## [6] " ON | 15445895 / R: 1794 ->1817 | N:2 | W | B | W | B | W | B"
## [7] "-----"
## [8] " 2 | DAKSHESH DARURI | 6.0 | W 63 | W 58 | L 4 | W 17 | W 16 | W 2"
## [9] " MI | 14598900 / R: 1553 ->1663 | N:2 | B | W | B | W | B | W"
## [10] "-----"
```

```
dash<-str_detect(tourney,"\\-----")
clean_tourney<-tourney[!dash]
clean_tourney<-clean_tourney[3:length(clean_tourney)] #taking out header
clean_tourney<-str_split(clean_tourney,"\\|")

tourney_raw<-str_extract_all(clean_tourney,"[[[:alpha:]]-?[:alpha:]] ?{2,}") #looking for n
tnames<-str_detect(unlist(tourney_raw),"[[[:alpha:]]]{3,}")
allnames<-unlist(tourney_raw)[tnames]
states<-str_detect(unlist(tourney_raw),"[[[:alpha:]]]{2,}") #now we need the players state
states<-unlist(tourney_raw)[(states)&(!tnames)]
tot_pts<-str_extract_all(clean_tourney,"\\d{1,}+\\\\.\\.\\.?") #looking for the points a player
```

```
act_pts<-str_detect(unlist(tot_pts), "\\d\\.\\d")
Points<-unlist(tot_pts)[act_pts]
```

At this juncture we've extracted the name, location and points scored data, however, we still need the pre and post ratings as well as the games played to calculate average opponent rating:

```
plyrratings<-str_extract_all(clean_tourney, "(( \\:|(>))?.?\\d{1,}P*\\.?.?")
prerate<-str_detect(unlist(plyrratings), "\\b\\d{3,4}P?\\b")
postrate<-post_loc<-str_detect(unlist(plyrratings), "\\>.?.?\\b\\d{3,4}P?\\b")
prerate<-unlist(plyrratings)[(prerate)&(!postrate)]
prerate<-str_replace_all(prerate, "P", "")
postrate<-unlist(plyrratings)[postrate]
postrate<-str_replace_all(postrate, "(>P)", "")
head(prerate)
```

```
## [1] " 1794" " 1553" " 1384" " 1716" " 1655" " 1686"
```

```
head(postrate)
```

```
## [1] "1817" "1663" "1640" "1744" "1690" "1687"
```

This Gives use the ratings we wished to extract, now we need to pull the matches played by each player

```
games<-str_extract_all(clean_tourney, "[WDL]\\.\\.\\.\\d{1,2}")
gamesplayed<-str_extract_all(games, "\\d{1,2}")
gamesplayed<-str_replace_all(gamesplayed, "\\b[0]\\b", ".")
gm_notplayed<-str_detect(gamesplayed, fixed("."))
gamesplayed<-gamesplayed[!(gm_notplayed)]
head(gamesplayed)
```

```
## [1] "c(\"39\", \"21\", \"18\", \"14\", \"7\", \"12\", \"4\")"
## [2] "c(\"63\", \"58\", \"4\", \"17\", \"16\", \"20\", \"7\")"
## [3] "c(\"8\", \"61\", \"25\", \"21\", \"11\", \"13\", \"12\")"
## [4] "c(\"23\", \"28\", \"2\", \"26\", \"5\", \"19\", \"1\")"
## [5] "c(\"45\", \"37\", \"12\", \"13\", \"4\", \"14\", \"17\")"
## [6] "c(\"34\", \"29\", \"11\", \"35\", \"10\", \"27\", \"21\")"
```

Almost there, we need now to create the first part of the final data that will enter our new CSV file:

```
PlayerID<-seq(1,64,by=1)
Name<-str_trim(allnames, "both")
Location<-str_trim(states, "both")
PreRating<-str_trim(prerate, "both")
PostRating<-str_trim(postrate, "both")
NewRankingList<-cbind(PlayerID, Name, Location, Points, PreRating, PostRating)
NewRankingList<-as.data.frame(NewRankingList)
NewRankingList$Points<-as.numeric(as.character(NewRankingList$Points)) #converting to numbers
```

```
NewRankingList$PreRating<-as.numeric(as.character(NewRankingList$PreRating)) #converting to numbers
NewRankingList$PostRating<-as.numeric(as.character(NewRankingList$PostRating)) #converting to numbers
head(NewRankingList)
```

	PlayerID	Name	Location	Points	PreRating	PostRating
## 1	1	GARY HUA	ON	6.0	1794	1817
## 2	2	DAKSHESH DARURI	MI	6.0	1553	1663
## 3	3	ADITYA BAJAJ	MI	6.0	1384	1640
## 4	4	PATRICK H SCHILLING	MI	5.5	1716	1744
## 5	5	HANSHI ZUO	MI	5.5	1655	1690
## 6	6	HANSEN SONG	OH	5.0	1686	1687

Finally We calculate the Opponents Average Rating (OAR)

```
opp_avg<-array(0,dim=nrow(NewRankingList))
for (i in 1:nrow(NewRankingList)){
  wdl<-as.numeric(str_split(unlist(str_extract_all(gamesplayed[i],"\\d{1,2}")), " "))
  opp_avg[i]<-mean(NewRankingList[wdl,colnames(NewRankingList)=="PreRating"])};

NewRankingList$OppAverageRank<-opp_avg
head(NewRankingList)
```

	PlayerID	Name	Location	Points	PreRating	PostRating
## 1	1	GARY HUA	ON	6.0	1794	1817
## 2	2	DAKSHESH DARURI	MI	6.0	1553	1663
## 3	3	ADITYA BAJAJ	MI	6.0	1384	1640
## 4	4	PATRICK H SCHILLING	MI	5.5	1716	1744
## 5	5	HANSHI ZUO	MI	5.5	1655	1690
## 6	6	HANSEN SONG	OH	5.0	1686	1687

  

	OppAverageRank
## 1	1605.286
## 2	1469.286
## 3	1563.571
## 4	1573.571
## 5	1500.857
## 6	1518.714

```
write.csv(NewRankingList,"NewRankingList.csv",row.names=FALSE)
```