Cleaning

Anuj Dahiya

December 24, 2019

The purpose of this document is to clean the `.csv` files stored in the *Step 2 - Reformat/Data csvs/* folder.

Before beginning, we need to import several packages that help run the code below.

```
library(knitr)
library(dplyr)
library(data.table)
library(lubridate)
```

Access the data

In order to access the data we need to specify the folder we want to access the files from. We do this by defining a path to find them and set the path with `opts_knit$set(root.dir = path)`. In an R-Markdown document, it is required that you set the directory using this function. We can see the files in the directory with `head()`.

We also create a destination path with `dest` for later use so that we can export the files properly. `country_path` is a path to a dataset from which we will clean up the data on.

```
path = "~/GitHub/FIDE/Chess Scripts/Step 2 - Reformat/Data csvs/"
country_path = "~/GitHub/FIDE/Chess Scripts/Step 3 - FIDE Country Codes/FIDE_codes.csv"
dest =  "~/GitHub/FIDE/Chess Scripts/Step 4 - Cleaning/Cleaned csvs/"
opts_knit$set(root.dir = path)

temp = list.files(path = path)
full_path <- paste(path, temp, sep = "")

head(temp)
```

```
## [1] "APR01.csv" "APR02.csv" "APR03.csv" "APR04.csv" "APR05.csv" "APR06.csv"
```

Create functions to rename datasets

Below, I define a few functions that help us rename the datasets.

`month` gets the first 3 letters of the temp elements, the month name `num` converts each month into a number `add_zero` converts each month number into a usable form when we create dates later on.

```
month <- function(x){return(substr(x, 1, 3))}
num <- function(x) match(tolower(x), tolower(month.abb))
add_zero <- function(x){if (x <= 9){x = paste("0", x, sep = "")}; return(x)}
```

Get the month number of all of the files in the dataset

Below, we rename create the variables names when they are assigned and imported.

```
temp%>%
sapply(month)%>%
sapply(num)%>%
sapply(add_zero)%>%
paste("20", substr(temp, 4, 5), "-", ., "-", "01", sep = "")-> month_num

head(month_num)
```

```
## [1] "2001-04-01" "2002-04-01" "2003-04-01" "2004-04-01" "2005-04-01"
## [6] "2006-04-01"
```

We can see that the datasets correspond to dates on which the is recorded.

Import all datasets

Below, we import and assign all of the datasets into memory. The only objects we need to hold onto is the data and several folder path references for later on.

```
for(i in 1:length(full_path)) {
assign(month_num[i], fread(full_path[i], sep = "*", data.table = FALSE,
                           strip.white = TRUE, blank.lines.skip = TRUE))
}

FIDE <- mget(ls(pattern = "[0-9][0-9]-[0-9][0-9]"))

rm(list=setdiff(ls(), c("FIDE", "path", "temp", "dest", "country_path")))
```

Data prep

```
vector_months <- c(month.abb, tolower(month.abb),toupper(month.abb))
string = ""
for (i in 1:length(vector_months)){
if (i == 1){string = vector_months[i]}
else if (i > 1) {string = paste(string, vector_months[i], sep = "|")}
}
string = paste(string, "RATING", sep = "|")


new = c("CM", "WCM", "WCM", "WGM", "WFM", "WFM", "GM", "IM", "FM", "WIM", "GM")
old = c("c", "wc", "WC", "wg", "WF", "wf", "g", "m", "f", "wm", "gm" )

dates = as.Date(names(FIDE))
months_vec <- months(dates)%>%toupper()%>%substr(., 1, 3)
year_vec <- year(dates)%>%as.character()%>%substr(3,4)
```

```r
files <- paste(months_vec, year_vec, ".csv", sep = "")

codes <- fread(country_path,
               sep =  ",", header = TRUE)


country_codes <- c("BDI", "BHU", "BUR", "CAF", "CAM", "CGO", "CIV", "CMR", "COD",
                   "CPV", "CUR", "DJI", "ERI", "FID", "FIE", "GAB", "GUM", "Ind",
                   "IVC", "KOR", "KOS", "KSA", "LAO", "LBN", "LBR", "LCA", "LES",
                   "MDV", "MTN", "NET", "NRU", "OMA", "PLW", "ROU", "SCG", "SGP",
                   "SLE", "SOL", "SSD", "STP", "SWZ", "TLS", "TPE", "TTO")

countries <- c("Berundi", "Bhutan", "Burkina Faso", "Central African Republic", "Cambodia",
            "Republic of the Congo", "Cote d'Ivoire", "Cameroon", "Democratic Congo",
            "Cape Verde", "Curaçao", "Djibouti", "Eritrea", "Finland", "FIE", "Gabon",
            "Guam", "India", "Côte d'Ivoire", "South Korea", "Kosovo", "Saudi Arabia",
            "Laos", "Lebanon", "Liberia", "Saint Lucia", "Lesotho", "Maldives", "Mauritania",
            "NET", "Nauru", "Oman", "Palau", "Romania", "Serbia and Montenegro", "Singapore",
            "Sierra Leonne", "Solomon Islands", "South Sudan", "Sao Tome and Principe",
            "Swaziland", "East Timor (Timor-Leste)  ", "Taiwan", "Trinidad and Tobago")
```

Ugly data cleaning

```r
for(i in 1:length(FIDE)){
  colnames(FIDE[[i]])[grepl("Name|NAME|name", colnames(FIDE[[i]]))] <- "Name"
  colnames(FIDE[[i]])[grepl("NUMBER", colnames(FIDE[[i]]))] <- "ID_Number"
  colnames(FIDE[[i]])[grepl("Fed|FED|COUNTRY", colnames(FIDE[[i]]))] <- "Country"
  colnames(FIDE[[i]])[grepl("Gms|GAMES|GM|Game|GAME", colnames(FIDE[[i]]))] <- "Games"
  colnames(FIDE[[i]])[grepl("K", colnames(FIDE[[i]]))] <- "K_factor"
  colnames(FIDE[[i]])[grepl("FLAG|Flag|flag", colnames(FIDE[[i]]))] <- "Activity"
  colnames(FIDE[[i]])[colnames(FIDE[[i]]) %in% c("Wtit","wtit","WTIT", "WTit")] <- "Womens_Title"
  colnames(FIDE[[i]])[colnames(FIDE[[i]]) %in% c("TITLE","Title","title","Tit")] <- "Title"
  colnames(FIDE[[i]])[grepl(string, colnames(FIDE[[i]]))] <- "Rating"
  colnames(FIDE[[i]])[grepl("Born|Age|age|BIRTHDAY|B-day|Bday", colnames(FIDE[[i]]))] <- "Age_Birthday"
  colnames(FIDE[[i]])[grepl("SEX", colnames(FIDE[[i]]))] <- "Sex"
  colnames(FIDE[[i]])[grepl("FOA", colnames(FIDE[[i]]))] <- "FIDE_Online_Arena"
  colnames(FIDE[[i]])[grepl("OTit", colnames(FIDE[[i]]))] <- "Other_Titles"
  FIDE[[i]] <- FIDE[[i]] %>%
               mutate(Date = as.POSIXct(names(FIDE)[i], format="%Y-%m-%d"),
                      Date_numeric = year(Date)+yday(Date)/366,
                      Rating = as.numeric(Rating),
                      Title= c(new, Title)[match(Title, c(old, Title))],
                      Country = c(codes$Country, Country)[match(Country, c(codes$Code, Country))],
                      Country = c(countries, Country)[match(Country, c(country_codes, Country))])%>%
                      filter(!Country %in% c("Fed", "Col"))%>%
                      select(-one_of("V1"))
  fwrite(FIDE[[i]] , file = paste(dest, files[i], sep = ""), sep = "*")
}
```

Example data after modifications

Name	Country	Rating	Title	Date
A C J John	India	1063		2019-12-01
A Chakravarthy	India	1151		2019-12-01
A E M, Doshtagir	Bangladesh	1840		2019-12-01
A hamed Ashraf, Abdallah	Egypt	1728		2019-12-01
A Hamid, Harman	Malaysia	1325		2019-12-01
A K M, Sourab	Bangladesh	1598		2019-12-01

```r
list.files(path = dest, pattern = "*.csv")%>%head()
```

```
## [1] "APR01.csv" "APR02.csv" "APR03.csv" "APR04.csv" "APR05.csv" "APR06.csv"
```