Tidyverse CREATE Assignment

Alexis Mekueko

10/27/2020

body .main-container { max-width: 1800px; max-height: 100px; }

R Packages

```
library(tidyverse) #loading all library needed for this assignment
#library(openintro)
#library(psych)
#head(fastfood)
library(readxl)
#library(data.table)
#library(DT)
library(knitr)

library(readr)
#library(plyr)
library(dplyr)
library(stringr)
#library(XML)
#library(RCurl)
#library(jsonlite)
#library(httr)

#library(maps)
#library(dice)
# #library(VennDiagram)
# #library(help = "dice")
#ibrary(DBI)
#library(dbplyr)

# library(rstudioapi)
# library(RJDBC)
# library(odbc)
# library(RSQLite)
# #library(rvest)

#library(readtext)
#library(ggpubr)
#library(fitdistrplus)
#library(ggplot2)
#library(moments)
#library(qualityTools)
#library(normalp)
#library(utils)
#library(MASS)
#library(qqplotr)
#library(DATA606)

#library(knitLatex)
#library(knitr)
#library(markdown)
#library(rmarkdown)
#render("DATA606_Project_Proposal.Rmd", "pdf_document")
```

Github Link: https://github.com/asmozo24/DATA606_Project_Proposal

Web link: https://rpubs.com/amekueko/682247

data source: https://www.kaggle.com/omarhanyy/500-greatest-songs-of-all-time

Description

This assignment is about getting familiar with two or more Tidyverse packages. So, I am going to write a vignette using readr, dplyr , and stringr which are part of the core tidyverse packages used for data analysis.

readr

According to tidyverse.org, readr provides a fast and friendly way to read rectangular data (like csv, tsv, and fwf). It is designed to flexibly parse many types of data found in the wild, while still cleanly failing when data unexpectedly changes. The required package for readr is under the tidyverse package (install.packages("tidyverse")) or you can just install single readr with install.packages("readr"). There is cheat sheet (https://github.com/rstudio/cheatsheets/blob/master/data-import.pdf) which can be helpful at a time. For this assignment, I will practice reading and saving csv file.

```
# setting the working directory
setwd("~/R/DATA607_Tidyverse")



# load the csv file which has all the variable.

Top_500Songs <- read.csv("https://raw.githubusercontent.com/asmozo24/DATA607_Tidyverse-CREATE-Assignmen/main/Top%20500%20Songs.csv", stringsAsFactors=FALSE)
str(Top_500Songs)
```

```
## 'data.frame':    500 obs. of  8 variables:
##  $ title      : chr  "Shop Around" "Buddy Holly" "Miss You" "The Rising" ...
##  $ description: chr  "Robinson thought Barrett Strong should record \"Shop Around,\" but Gordy persuaded Smokey that he was the right"| __truncated__ "In the early 1990s, Cuomo had an awkward girlfriend who was routinely picked on. His efforts to stick up for he"| __truncated__ "The Stones were in Toronto, rehearsing for their classic gigs at the El Mocambo Club, when Jagger, jamming with"| __truncated__ "Springsteen wrote the track about 9/11, taking the viewpoint of a firefighter entering one of the Twin Towers ("| __truncated__ ...
##  $ appears.on : chr  "The Ultimate Collection (Motown)" "Weezer (Geffen)" "Some Girls (Virgin)" "The Rising (Columbia)" ...
##  $ artist     : chr  "Smokey Robinson and the Miracles" "Weezer" "The Rolling Stones" "Bruce Springsteen" ...
##  $ writers    : chr  "Berry Gordy, Robinson" "Rivers Cuomo" "Mick Jagger, Keith Richards" "Springsteen" ...
##  $ producer   : chr  "Gordy" "Ric Ocasek" "The Glimmer Twins" "Brendan O'Brien" ...
##  $ released   : chr  "Dec. '60, Tamla" "Aug. '94, DGC" "May '78, Rolling Stones" "July '02, Columbia" ...
##  $ streak     : chr  "16 weeks; No. 2" "21 weeks; No. 18" "20 weeks; No. 1" "11 weeks; No. 52" ...
```

```
view(Top_500Songs)

# file to big, cleaning/removing the column I don't need
Top_500Songs <- Top_500Songs [, -2]
# saving the new csv file
write.csv(Top_500Songs,'Top_500Songs.csv')

glimpse(Top_500Songs)
```

```
## Rows: 500
## Columns: 7
## $ title      <chr> "Shop Around", "Buddy Holly", "Miss You", "The Rising", ...
## $ appears.on <chr> "The Ultimate Collection (Motown)", "Weezer (Geffen)", "...
## $ artist     <chr> "Smokey Robinson and the Miracles", "Weezer", "The Rolli...
## $ writers    <chr> "Berry Gordy, Robinson", "Rivers Cuomo", "Mick Jagger, K...
## $ producer   <chr> "Gordy", "Ric Ocasek", "The Glimmer Twins", "Brendan O'B...
## $ released   <chr> "Dec. '60, Tamla", "Aug. '94, DGC", "May '78, Rolling St...
## $ streak     <chr> "16 weeks; No. 2", "21 weeks; No. 18", "20 weeks; No. 1"...
```

```
#view(Top_500Songs)
```

dplyr

According to tidyverse.org, dplyr provides a grammar of data manipulation, providing a consistent set of verbs that solve the most common data manipulation challenges. package for dplyr is uner tidyverse or single installation similar to readr. dplyr offers some verbs used for data manipulation: mutate(): uses this verb to add new variable based on the manipulation of the existing variables. select(): lets you select specific variables of your interest. filter(): similar to select, just that this verb goes into the variable and picks your value (s) of interest. summarise(): provides a summary of your dataframe. arrange(): this verb is good for ordering of rows in your dataframe.

Based on previous assginments, I can say dply syntax eliminate the need to use '$' which is used in the base function. In addition multiple verbs can be used or group together with %>%.

```
# let's check if there is a missing value in a specific column
# return 06 rows with empty values...tempted to delete data but will not do it now....no need
Top_500Songs %>%
  filter( is.na(streak) | streak == "")
```

title <chr>	**appears.on** <chr>	**artist** <chr>	**writers** <chr>
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin

title <chr>	appears.on <chr>	artist <chr>	writers <chr>
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin

6 rows | 1-5 of 7 columns

```
# another way
filter(Top_500Songs, is.na(streak) | streak == "")
```

title <chr>	appears.on <chr>	artist <chr>	writers <chr>
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin
Candle in the Wind	Goodbye Yellow Brick Road (Island)	Elton John	John, Bernie Taupin

6 rows | 1-5 of 7 columns

```
filter(Top_500Songs, !grepl("weeks", streak))
```

title <chr>	appears.on <chr>
Time to Pretend	Oracular Spectacular (Columbia)
Sabotage	Ill Communication (Capitol)

title <chr>	appears.on <chr>
Last Nite	Is This It (RCA)
How Soon Is Now?	Meat Is Murder (Warner Bros.)
Into the Mystic	Moondance (Warner Bros.)
Rollin Stone	The Anthology: 1947-1972 (Chess/MCA)
Heroin	The Velvet Underground and Nico (Polydor)
Pressure Drop	The Harder They Come (Hip-O)
Time to Pretend	Oracular Spectacular (Columbia)
Sabotage	Ill Communication (Capitol)

1-10 of 102 rows | 1-2 of 7 columns Previous **1** 2 3 4 5 6 … 11 Next

```
# Being in the top 500 greatest songs of all time, I will assum the song hits the hit parade of billboard for few months...lets check that
Top_500Songs %>%
  select(streak)%>%
  filter(grepl("weeks", streak))
```

streak <chr>
16 weeks; No. 2
21 weeks; No. 18
20 weeks; No. 1
11 weeks; No. 52
17 weeks; No. 11
12 weeks; No. 1
42 weeks; No. 2
27 weeks; No. 1
20 weeks; No. 1
22 weeks; No. 29

1-10 of 398 rows Previous **1** 2 3 4 5 6 … 40 Next

```
# what if I want to find the songs that stayed on top for longest period...this is like string sea
rch comparison which is bit tidious
# I think a manual search and create a new variable called ranking

Top_500Songs %>%
  select(streak)%>%
  filter(grepl( "No. 1", streak))
```

streak
<chr>
21 weeks; No. 18
20 weeks; No. 1
17 weeks; No. 11
12 weeks; No. 1
27 weeks; No. 1
20 weeks; No. 1
18 weeks; No. 1
24 weeks; No. 16
21 weeks; No. 1
14 weeks; No. 1

1-10 of 188 rows Previous **1** 2 3 4 5 6 … 19 Next

```
# or but not really helpful ...the nature of the data
songRank <- Top_500Songs %>%
  arrange(desc(streak))

#view(songRank)  ...if streak was numerical ...this would be perfect

Top_500Songs %>%
  mutate(rank = min_rank(desc(streak)))%>%
  arrange(desc(rank))
```

title	appears.on
<chr>	<chr>
Candle in the Wind	Goodbye Yellow Brick Road (Island)
Candle in the Wind	Goodbye Yellow Brick Road (Island)
Candle in the Wind	Goodbye Yellow Brick Road (Island)
Candle in the Wind	Goodbye Yellow Brick Road (Island)

title <chr>	appears.on <chr>
Candle in the Wind	Goodbye Yellow Brick Road (Island)
Candle in the Wind	Goodbye Yellow Brick Road (Island)
Penny Lane	Magical Mystery Tour (Capitol/Apple)
Penny Lane	Magical Mystery Tour (Capitol/Apple)
Standing in the Shadows of Love	The Ultimate Collection (Motown)
Standing in the Shadows of Love	The Ultimate Collection (Motown)

1-10 of 500 rows | 1-2 of 8 columns Previous 1 2 3 4 5 6 ... 50 Next

```
# let's check if R.Kelly is on the list
Top_500Songs %>%
  filter(artist == "R. Kelly" )
```

title <chr>	appears.on <chr>	artist <chr>	writers <chr>	produ… <chr>	released <chr>	str <c
Ignition (Remix)	Chocolate Factory (Jive)	R. Kelly	Kelly	Kelly	Oct. '02, Jive	42
Ignition (Remix)	Chocolate Factory (Jive)	R. Kelly	Kelly	Kelly	Oct. '02, Jive	42

2 rows

```
# let's say I only want to see R.Kelly record (song title , release date and streak)
Top_500Songs %>%
  select(title, artist, released, streak) %>%
  filter(artist == "R. Kelly")
```

title <chr>	artist <chr>	released <chr>	streak <chr>
Ignition (Remix)	R. Kelly	Oct. '02, Jive	42 weeks; No. 2
Ignition (Remix)	R. Kelly	Oct. '02, Jive	42 weeks; No. 2

2 rows

```
# How about I add a new variable which shows R.Kelly youtube view of the title song.
#Top_500Songs %>%
#  mutate(youtubeView = ifelse(filter(Top_500Songs, artist == "R. Kelly"), "R.Kelly: 232,560,092"
 ))
# https://www.youtube.com/watch?v=y6y_4_b6RS8
```

stringr

According to tidyverse.org, stringr provides a cohesive set of functions designed to make working with strings as easy as possible. It is built on top of stringi, which uses the ICU C library to provide fast, correct implementations of common string manipulations. the required package comes under tidyverse package or single stringr. helpful, cheat sheet…https://github.com/rstudio/cheatsheets/blob/master/strings.pdf

```
#let say I want to check my favorite artist and I don't remember their full name
Top_500Songs %>%
  #select(artist) %>%
  filter(grepl("50 Cent", artist))
```

title <chr>	appears.on <chr>	artist <chr>
In Da Club	Get Rich or Die Tryin' (Interscope/Aftermath/Shady)	50 Cent
In Da Club	Get Rich or Die Tryin' (Interscope/Aftermath/Shady)	50 Cent

2 rows | 1-3 of 7 columns

```
artist <- unlist(Top_500Songs %>%
  #select(artist) %>%
  filter(grepl("50 Cent", artist)))

# another way to detect matching pattern
str_detect(artist, "Rich")
```

```
##  [1] FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [13] FALSE FALSE
```

```
# find a matching pattern and display/extract
str_subset(artist, "Dr.")
```

```
## [1] "50 Cent, Dr. Dre, Mike Elizondo" "50 Cent, Dr. Dre, Mike Elizondo"
## [3] "Dr. Dre, Elizondo"               "Dr. Dre, Elizondo"
```

```
# Find the length of a vector
str_length(artist)
```

```
##  [1] 10 10 51 51  7  7 31 31 17 17 36 36 15 15
```

```
#string count
str_count(artist, "Dr.")
```

```
##  [1] 0 0 0 0 0 0 2 2 2 2 0 0 0 0
```

```
# convert string to upper case
str_to_upper(artist)
```

```
##  [1] "IN DA CLUB"
##  [2] "IN DA CLUB"
##  [3] "GET RICH OR DIE TRYIN' (INTERSCOPE/AFTERMATH/SHADY)"
##  [4] "GET RICH OR DIE TRYIN' (INTERSCOPE/AFTERMATH/SHADY)"
##  [5] "50 CENT"
##  [6] "50 CENT"
##  [7] "50 CENT, DR. DRE, MIKE ELIZONDO"
##  [8] "50 CENT, DR. DRE, MIKE ELIZONDO"
##  [9] "DR. DRE, ELIZONDO"
## [10] "DR. DRE, ELIZONDO"
## [11] "DEC. '02, INTERSCOPE/AFTERMATH/SHADY"
## [12] "DEC. '02, INTERSCOPE/AFTERMATH/SHADY"
## [13] "30 WEEKS; NO. 1"
## [14] "30 WEEKS; NO. 1"
```

```
# convert string to lower case
str_to_lower(artist)
```

```
##  [1] "in da club"
##  [2] "in da club"
##  [3] "get rich or die tryin' (interscope/aftermath/shady)"
##  [4] "get rich or die tryin' (interscope/aftermath/shady)"
##  [5] "50 cent"
##  [6] "50 cent"
##  [7] "50 cent, dr. dre, mike elizondo"
##  [8] "50 cent, dr. dre, mike elizondo"
##  [9] "dr. dre, elizondo"
## [10] "dr. dre, elizondo"
## [11] "dec. '02, interscope/aftermath/shady"
## [12] "dec. '02, interscope/aftermath/shady"
## [13] "30 weeks; no. 1"
## [14] "30 weeks; no. 1"
```

```
#string view
#str_view(artist, "Cent")
str_match(artist, 'Cent')
```

```
##       [,1]
##  [1,] NA
##  [2,] NA
##  [3,] NA
##  [4,] NA
##  [5,] "Cent"
##  [6,] "Cent"
##  [7,] "Cent"
##  [8,] "Cent"
##  [9,] NA
## [10,] NA
## [11,] NA
## [12,] NA
## [13,] NA
## [14,] NA
```

Conclusion

I think readr and dplyr are inevitable in R analysis. stringr is more focus on doing search or learning about a particular variable.