Assignment 5

STATS 220 Semester One 2022

This assignment draws on lecture and lab content from Module 5: Creating data from digital sources. The knowledge and skills assessed by the assignment will be covered in lectures/labs by the end of Module 5. Consequently, the order of the assignment instructions below does not necessarily reflect the order the material is covered in lectures/labs.

Computing information

You will need access to a computer that has R installed and RStudio installed. **You should already have installed the R packages {tidyverse}, {jsonlite} and {gganimate}. You will need to install the package {rvest} as well.**

Use `install.packages("rvest")` to install the {rvest} package but do not leave this code in your `index.Rmd` file as you only need to run this once. If installing R, RStudio, or these packages onto your computer is a problem, you can use the free level of RStudio Cloud.

Please note that web scraping with {rvest} requires you to be about to access the web and make requests from within R Studio. This is very likely going to be a problem for students who use a VPN. Therefore, you will need to use RStudio Cloud to complete this assignment if you have VPN issues.

Within your *stats220* folder on your computer, create a new project using RStudio that is called "Assignment5".

Create all R script files and Rmd files in this folder, and save any data and image files (e.g. plots, animated GIFs) to this folder.

For this assignment, you will submit ONLY your index.html file.

Provided data context and sources

Apple music releases daily top 100 playlists for countries around the world. We crowd sourced some of these in Lecture 5B1.

For example, this is the link to the top 100 most played songs in New Zealand: https://music.apple.com/us/playlist/top-100-new-zealand/pl.d8742df90f43402ba5e708eefd6d949a

The link to the each song on the Apple music top 100 playlists contains the unique identifier for that song in the Apple/iTunes database (its iTunes ID or trackId).

Apple also provides an API for iTunes (the older version of Apple Music). You can request information about a particular song/track, using its iTunes ID/trackId.

For this assignment, you need to combine data from at least four different daily top 100 playlists from those provided by Apple music.

You then need to join your data from the Apple music top 100 playlists with data sourced from the iTunes API. You will need to use the iTunes ID/trackId to do this.

The overall goal of the data sourcing is to create a visualisation that compares something about the different playlists.

If you want to use a different data context and sources for your assignment, you need to request permission from Anna.

You will need:

- a webpage that has more than one "page" of data that you will scrape
- a webpage that allows robots i.e. permits web scraping
- another data source to join your data with e.g. an API, another webpage, CSV file, JSON file, database etc.
- a method/strategy to join your two sources of data
- data sources that will allow you to demonstrate the required R code approaches outlined in each of the parts of the assignment.

Part A

Create a R script file called "data_web.R".

Load the libraries {tidyverse} and {rvest}.

Choose at least four different Apple music top 100 playlists to scrape. *These can be four different countries, or four different days for the same countries, or two different days for two different countries, etc.*

Use {tidyverse} and {rvest} functions to create a data frame called `apple_data` with at least three columns/variables. *Consider the advice given in lectures about what information you should record when web scraping.*

- **One of these variables must be the iTunes ID/trackId and be called `trackId` in your data frame**
- Use either `bind_rows()` or `map_df()`, along with `tibble()`, to create your combined data frame from at least four different playlists
- If you use `map_df()`, make sure to take at least two second breaks between each scrape using `Sys.sleep(2)`

Save your combined data frame as a rds file within your project folder using `saveRDS(apple_data, "apple_data.rds")`

The code below is a demonstration of how to scrape the itunes ID/trackId for each song on a Apple music top 100 playlist.

```
url <- "https://link.here"

page <- read_html(url)

trackId <- page %>%
  html_elements(".songs-list") %>%
  html_elements("a") %>%
  html_attr("href") %>%
  str_remove_all("https://music.apple.com/us/song/")
```

Please refer to lecture examples for other help/guidance for how to scrape other information for each song on a Apple music top 100 playlist.

Part B

Create a R script file called "data_api.R".

Load the libraries {tidyverse} and {jsonlite}.

Create a data frame called `apple_data` by reading in the data you created in Part A using `readRDS("apple_data.rds")`.

Extract `trackId` as a vector called `trackIds` and only keep the unique value e.g.

```
trackIds <- apple_data$trackId %>% unique()
```

Use the `track_ids` to query the iTunes API and request data about each song. The code to use to make requests to the iTunes API is given below **for one song**.

```
song_id <- trackIds[1]
url <- paste0("https://itunes.apple.com/lookup?id=", song_id)
response <- fromJSON(url)
itunes_data <- response$results
```

You will need to adapt this code so that you:

- iterate through all of the `trackIds` using `map_df()`
- combine the iTunes API data for each song into a data frame called `itunes_data`
- take at least a two second break between each API call using `Sys.sleep(2)`

Save your combined data frame as a rds file within your project folder using `saveRDS(itunes_data, "itunes_data.rds")`

Part C

Create a R script file called "data_join.R".

Load the libraries {tidyverse}.

Create a data frame called `apple_data` by reading in the data you created in Part A using `readRDS("apple_data.rds")`.

Create a data frame called `itunes_data` by reading in the data you created in Part B using `readRDS("itunes_data.rds")`.

Create a data frame called `song_data` by joining the data frames `apple_data` and `itunes_data` by the variable `trackId`.

Inspect the data frame `song_data` to understand what variables/columns of data you have.

Save your joined data frame as a rds file within your project folder using `saveRDS(song_data, "song_data.rds")`

Part D

Create a R script file called "data_viz.R".

Load the libraries {tidyverse}.

Create a data frame called `song_data` by reading in the data you created in Part C using `readRDS("song_data.rds")`.

You need to make **ONE** visualisation with your data that compares something about the different playlists using the data you have sourced. Explore your data and use {dplyr} to manipulate your data so that you can use it for visualisation.

Similar to Assignment 4, there is an expectation that you will manipulate the data in some way using functions such as `group_by()`, `summarise()`, `mutate()`, `filter()`, `arrange()`, `slice()`, etc. You do not need to use all of these functions, just the ones that help you obtain your goal for visualisation. **You can also use functions from {stringr} and {lubridate}.**

Use {ggplot2} to create **ONE** visualisation (you can also use {gganimate} if you wish). Apply the grammar of graphics and try to produce a really "professional" looking visualisation e.g. titles, labels, annotations, not the default gray background, etc.

An optional package you can use for your visualisation is the package {patchwork}. Information about how to install and use this package is available here: https://patchwork.data-imaginist.com/.

After you produce your plot, save it in your project folder as a PNG file using `ggsave("song_vis.png", width = 8, height = 5, units = "in")`. You can change the width and height of the dimensions of your plot if you need.

Part E

Create a R script file called "data_creativity.R".

Create something using R code and your data, using any knowledge or techniques from across all STATS 220 e.g. lectures, labs, assignments.

Save what you have created to your project folder so you can include it in your report in Part F.

Please read the special note at the end of this document.

Part F

Create a new Rmd file called "index.Rmd" with four sections: Introduction, Visualisation, Data creation, Final reflection, and Code appendix.

In the **Introduction** section, explain why you selected the Apple music top 100 playlists that you did, and what you were trying (or hoping) to find out when you compared them. *You need to write at least 50 words for this section.*

In the **Visualisation** section, include your visualisation using markdown and *write at least 50 words about your visualisation* e.g. what story does it communicate?

In the **Data creation** section, include what you created using markdown and *write at least 50 words* explaining how what you created uses knowledge or techniques from STATS 220.

In the **Final reflection** section, describe in your own words at least ONE important idea you learned from Module 5 *Creating data from digital sources*. Additionally, write an overall reflection on your learning across all the assignments in the course e.g. What did you learn from these? What are you curious to learn more about? *You need to write at least 100 words for this section.*

In the **Code appendix** section, use the following code chunks in your Rmd file to provide all of your R script file code in the appendix.

````
```{r file='data_web.R', eval=FALSE}

```
````

````
```{r file='data_api.R', eval=FALSE}

```
````

````
```{r file='data_join.R', eval=FALSE}

```
````

````
```{r file='data_viz.R', eval=FALSE}

```
````

````
```{r file='data_creativity.R', eval=FALSE}

```
````

Knit your *index.Rmd* file to create a self-contained *index.html* file.

For this assignment, you will ONLY submit your index.html file. Make sure that the HTML file does not contain any code (apart from the appendix).

Marking guide

The assignment will be marked out of 10.

- The instructions for Parts A, B, C, D, E and F were followed, and the final report was submitted as a HTML file **(7 marks)**
- The R code used is from {tidyverse}, is readable, is used to source, combine, join, manipulate and visualise data for a meaningful purpose, and is clearly informed by examples used in lectures and labs for STATS 220 **(1 mark)**
- The web page (HTML file) uses markdown and CSS effectively **(1 mark)**
- The assignment submitted demonstrates creativity and good communication skills **(1 mark)**

Special note about Part E

I have decided to offer a "redemption" opportunity as part of this assignment.

This applies to students who lost marks in Assignments 1, 2 or 3.

Students can do MORE than what is asked within Part E of the assignment, to gain up to an additional **THREE** marks, to be added to the total marks for Assignments 1, 2 or 3 (that is, you will only get three marks in total, not three marks for each assignment).

To take up this "redemption" opportunity, you will need to clearly show how what you have created demonstrates the understanding that you lost marks for in one or more of Assignments 1, 2 or 3.

Note, you need to do MORE work for Part E to gain these "redemption" marks.