Data Science - Practice 3

모든 문제에 대하여 코드만 작성하지 말고 데이터를 해석한 결과를 함께 작성하시오.

Put your explanation of your findings from dataset with your answer in your report as well as R code.

Problem 1

country 데이터는 국가별 지표와 대륙 정보를 담고 있는 데이터 프레임입니다.

The data frame called 'country' contains various national indicators and continental information.

variable	의미	mean
code	국가 코드	country's code
country_name	국가 이름	country's name
continent	대륙	continent
GDP	1인당 국내총생산(USD)	Gross Domestic Product per capita (USD)
life_expect	기대수명	life expectancy
population	인구 수	population
CO2	CO2 배출량 (추정치)	CO2 emission quantity (estimated)
battle_death	전투 중 사망자 (100,000명당)	a death in battle (per 100,000)
child.per.women	여성 1명당 아이의 수	number of children per woman
programmable.aid	국가별 프로그램 원조	national program aid

Loading data into R

```
load(url('https://github.com/hbchoi/SampleData/raw/master/country.RData'))
```

```
str(country)
```

```
## 'data.frame':    126 obs. of  10 variables:
##  $ code            : chr  "afg" "alb" "dza" "arg" ...
##  $ country_name    : chr  "Afghanistan" "Albania" "Algeria" "Argentina" ...
##  $ continent       : chr  "Asia" "Europe" "Africa" "South America" ...
##  $ GDP             : int  1757 11357 13940 18645 8159 44606 44671 16132 43732 3424 ...
##  $ life_expect     : num  61.2 78.1 77.4 76.5 75.4 ...
##  $ population      : int  35400000 2890000 40600000 43500000 2940000 24300000 8750000 9740000 1430000
##  $ CO2             : num  8660 4540 148000 200000 5180 413000 67400 37200 31500 76100 ...
##  $ battle_death    : num  9.45 0.13 3.41 0 0 0 0 0.0726 0 0.165 ...
##  $ child.per.woman : num  4.64 1.71 2.78 2.29 1.63 1.85 1.49 2.08 2.03 2.1 ...
##  $ programmable.aid: num  3663.3 277.2 108.3 59.1 373.1 ...
```

< Question 1 > - 'apply' function

수치형 변수의 경우에는 type이 integer 또는 double로 되어 있다. 이를 이용하여 수치형 변수에 해당하는 변수만 추출해보자.

Numeric variables can be either in type of "integer" or "double". Select the numeric variables from the dataset.

tip) typeof와 sapply를 사용한다. integer 또는 double인지 확인하기 위해서 아래의 코드를 사용하면 된다.

tip) You may combine 'typeof' and 'sapply' function. To find out whether a variable is "interger" or "double" type, refer to code below.

```
x %in% c("integer", "double")
```

Sample Result

```
## [1] "GDP"            "life_expect"      "population"        "CO2"
## [5] "battle_death"   "child.per.woman"  "programmable.aid"
```

< Question 2 > - 'rank' function

rank 함수는 변수에서 특정 값의 순위를 표시해준다. 다음 예시를 보며 확인해보자.

The 'rank' function find the rank of each value in a variable or vector as shown in the example below.

```
movie <- c('Harry Potter' = 4.2, 'Toy Story' = 4.1, 'Frozen' = 4.4, 'The Notebook' = 3.6)
rank(movie)
```

```
## Harry Potter      Toy Story        Frozen The Notebook
##            3              2             4            1
```

```
rank(desc(movie))
```

```
## Harry Potter      Toy Story        Frozen The Notebook
##            2              3             1            4
```

이를 활용하여, country 데이터프레임에서 각 수치형 변수들에 따른 나라별 순위를 표시하세요. rank의 기준은 큰 값을 1위로 한다 (내림차순 정렬).

List the countries with their rank of each numerical variable in the 'country' data frame. The largest value will be ranked as the first one.

Sample Result

```
##      code country_name     continent GDP life_expect population CO2 battle_death
## 1   afg   Afghanistan         Asia   116         119       33.0  86            5
## 2   alb       Albania       Europe    71          43      106.5 103           36
## 3   dza       Algeria       Africa    65          47       30.0  32           12
## 4   arg     Argentina South America   51          54       29.0  27           93
## 5   arm       Armenia         Asia    81          63      104.0 100           93
## 6   aus     Australia      Oceania    16          11       44.0  15           93
## 7   aut       Austria       Europe    14          16       76.0  41           93
## 8   aze    Azerbaijan         Asia    55          90       71.0  58           43
## 9   bhr       Bahrain         Asia    17          34      113.0  61           93
## 10  bgd    Bangladesh         Asia   102          79        7.0  38           35
##      child.per.woman programmable.aid
## 1              13.0                 3
## 2              93.0                95
## 3              35.0               110
## 4              56.0               121
## 5              99.0                91
## 6              81.5                47
## 7             110.0                47
## 8              62.5               103
## 9              67.0                47
## 10             61.0                 7
```

< Question 3 >

South Korea의 경우에는 순위가 어떻게 되는지 확인해보자. 무엇을 알 수 있는가?

Seeing the rank of South Korea, what can be inferred?

Sample Result

```
##      code country_name continent GDP life_expect population CO2 battle_death
## 62   kor   South Korea      Asia  27          10         24   9           38
##      child.per.woman programmable.aid
## 62            122.5                47
```

< Question 4 >

각 국가별로 지표들의 순위의 평균값을 계산한 후, 평균 순위가 높은 순으로 국가들을 정렬하여 나타내어라.

Calculate the average ranks for each country, and then list the countries in ascending order of their average rank.

Sample Result

```
##     code   country_name avg_rank
## 1   usa  United States 29.42857
## 2   sau   Saudi Arabia 31.14286
## 3   fra         France 35.57143
## 4   gbr United Kingdom 35.92857
## 5   isr         Israel 36.50000
## 6   idn      Indonesia 37.28571
## 7   col       Colombia 38.21429
## 8   pak       Pakistan 39.57143
## 9   ind          India 39.64286
## 10  kor    South Korea 39.64286
## 11  rus         Russia 39.85714
## 12  esp          Spain 40.00000
## 13  can         Canada 40.14286
## 14  tur         Turkey 40.21429
## 15  ita          Italy 40.71429
## 16  deu        Germany 40.78571
## 17  jpn          Japan 41.92857
## 18  phl    Philippines 43.07143
## 19  nld    Netherlands 43.64286
## 20  aus      Australia 43.92857
## 21  bgd     Bangladesh 47.00000
## 22  dza        Algeria 47.28571
## 23  chn          China 47.28571
## 24  egy          Egypt 47.42857
## 25  jor         Jordan 47.57143
## 26  zaf   South Africa 47.64286
## 27  vnm        Vietnam 48.14286
## 28  pol         Poland 50.21429
## 29  eth       Ethiopia 50.92857
## 30  kwt         Kuwait 51.14286
```

Problem 2

apps_delimiter.csv 파일은 Google play에 올라온 app들의 정보를 담고 있는 데이터입니다.

The 'apps_delimiter.csv' file is data that contains information of mobile applications from Google play.

variable	의미	mean
App	앱의 이름	App's name
Category	앱의 카테고리	category of app
Rating	앱의 평점	rating of app
Reviews	리뷰를 남긴 사람의 수	number of people who wrote the review
Size	앱의 사이즈	size of app
Install	다운로드 횟수	download frequency
Type	유/무료 여부	paid or free
Price	앱의 가격 (무료 앱의 경우 0)	price of app (Free app is 0)
Content.Rating	사용 연령 등급	age grade of use
Genres	앱의 장르	genres of app
Last.Updated	가장 최근 업데이트 날짜	recent update date
Current.Ver	현재 버전	current version
Android.Ver	사용 가능한 안드로이드 버전	available android version

< Question 5 >

이 파일은 쉼표 (,)가 아닌 쐐기 (^)로 데이터가 구분되어 있다. 이 파일을 불러와서 "app"이라는 이름의 데이터프레임으로 저장하라. 이때 string형식이 factor로 자동변환 되지 않도록 하자. 첫 columns인 'x'는 의미 없는 인덱스이기 때문에 삭제하자. 또한 변수중에 의미적으로 categorical인 변수가 있다면 factor로 변환해보자.

This dataset is delimited with wedges (^) rather than commas (,) in the file. Load this file and save it as a data frame named "app". Try not to automatically transform string date into factors type. Delete the first column, 'x', because they are meaningless index. Also, if there are variables inherently categorical, then convert those variables into factors.

```
str(app)
```

```
## 'data.frame':    1894 obs. of  13 variables:
## $ App         : chr  "Photo Editor & Candy Camera & Grid & ScrapBook" "Coloring book moana" "U Lau
## $ Category    : chr  "ART_AND_DESIGN" "ART_AND_DESIGN" "ART_AND_DESIGN" "ART_AND_DESIGN" ...
## $ Rating      : num  4.1 3.9 4.7 4.5 4.3 4.4 3.8 4.1 4.4 4.7 ...
## $ Reviews     : int  159 967 87510 215644 967 167 178 36815 13791 121 ...
## $ Size        : chr  "19M" "14M" "8.7M" "25M" ...
## $ Installs    : chr  "10^000+" "500^000+" "5^000^000+" "50^000^000+" ...
## $ Type        : chr  "Free" "Free" "Free" "Free" ...
## $ Price       : chr  "0" "0" "0" "0" ...
## $ Content.Rating: chr  "Everyone" "Everyone" "Everyone" "Teen" ...
## $ Genres      : chr  "Art & Design" "Art & Design;Pretend Play" "Art & Design" "Art & Design" ...
## $ Last.Updated : chr  "07-Jan-18" "15-Jan-18" "01-Aug-18" "08-Jun-18" ...
## $ Current.Ver : chr  "1.0.0" "2.0.0" "1.2.4" "Varies with device" ...
## $ Android.Ver : chr  "4.0.3 and up" "4.0.3 and up" "4.0.3 and up" "4.2 and up" ...
```

< Question 6 > - 'aggregate' function

aggregate 함수를 이용해서 장르 별로 평균 Rating을 구하여라.

Use the 'aggregate' function to find the average rating for each genre.

Sample Result

```
##                              Genres   Rating
## 1                           Action 4.456522
## 2     Action;Action & Adventure 4.400000
## 3                        Adventure 4.300000
## 4 Adventure;Action & Adventure 4.500000
## 5                           Arcade 4.409091
## 6                   Art & Design 4.311905
```

< Question 7 > - 'sort and order' function

app들을 Review의 개수가 많은 순서부터 내림차순해서 정렬하도록 하자.

Let's sort mobile applications in descending order of their number of reviews.

Sample Result

```
##                                       App       Category Rating  Reviews
## 323                       WhatsApp Messenger COMMUNICATION    4.4 69119316
## 367                       WhatsApp Messenger COMMUNICATION    4.4 69119316
## 368  Messenger ??Text and Video Chat for Free COMMUNICATION    4.0 56646578
## 322  Messenger ??Text and Video Chat for Free COMMUNICATION    4.0 56642847
## 1776                          Clash of Clans          GAME    4.6 44893888
## 1569                          Clash of Clans          GAME    4.6 44891723
##                  Size       Installs Type Price Content.Rating         Genres
## 323  Varies with device 1^000^000^000+ Free     0       Everyone Communication
## 367  Varies with device 1^000^000^000+ Free     0       Everyone Communication
## 368  Varies with device 1^000^000^000+ Free     0       Everyone Communication
## 322  Varies with device 1^000^000^000+ Free     0       Everyone Communication
## 1776             98M  100^000^000+ Free     0   Everyone 10+      Strategy
## 1569             98M  100^000^000+ Free     0   Everyone 10+      Strategy
##      Last.Updated       Current.Ver        Android.Ver
## 323     03-Aug-18 Varies with device Varies with device
## 367     03-Aug-18 Varies with device Varies with device
## 368     01-Aug-18 Varies with device Varies with device
## 322     01-Aug-18 Varies with device Varies with device
## 1776    15-Jul-18          10.322.16        4.1 and up
## 1569    15-Jul-18          10.322.16        4.1 and up
```