Assignment 3

2022-09-19

Assignment 3

Question 1

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
WNBA = read.csv("WNBA_Stats_21.csv", sep = ",", header = TRUE)
head(select(WNBA, contains("FG")))
```

```
##   FGM FGA
## 1 207 466
## 2   6  26
## 3  56 174
## 4  61 143
## 5   8  34
## 6 121 270
```

a.)

```
count(filter(WNBA, FTM > 50, AST > 75))
```

```
##    n
## 1 18
```

There are 18 players with FTM $>$ 50 and AST $>$ 75.

b.)

```
WNBA %>%
  select(PLAYER, TEAM, FGM, TO, PTS) %>%
  arrange(desc(PTS)) %>%
  head(10)
```

```
##                  PLAYER TEAM FGM  TO PTS
## 1          Tina Charles  WAS 238  59 631
## 2       Brittney Griner  PHO 248  66 615
## 3      Arike Ogunbowale  DAL 199  68 599
## 4           A'ja Wilson  LVA 207  46 584
## 5       Breanna Stewart  SEA 194  47 569
## 6       Kelsey Mitchell  IND 212  65 569
## 7  Skylar Diggins-Smith  PHO 177  82 566
## 8           Jewell Loyd  SEA 193  71 555
## 9        Betnijah Laney  NYL 203 119 536
## 10    Courtney Williams  ATL 228  58 529
```

Brittney Griner had the second highest points.

c.)

```
WNBA = WNBA %>% mutate(FGP = (FGM / FGA) * 100)
WNBA$FGP = round(WNBA$FGP, digits = 2)
WNBA = WNBA %>% mutate(FTP = (FTM / FTA) * 100)
WNBA$FTP = round(WNBA$FTP, digits = 2)
head(WNBA)
```

```
##           PLAYER TEAM AGE  G  MIN FGM FGA X3PM X3PA FTM FTA OREB DREB REB AST
## 1    A'ja Wilson  LVA  25 32 1021 207 466    1    1 169 193   63  235 298  98
## 2 Aaliyah Wilson  IND  23 14  119   6  26    1    7   2   4    5    7  12   8
## 3  Aari McDonald  ATL  23 30  493  56 174   32  104  45  51    9   40  49  59
## 4  Aerial Powers  MIN  28 14  309  61 143   11   35  55  60   13   38  51  29
## 5   Alanna Smith  PHO  25 18  117   8  34    4   21   1   4    5   19  24  10
## 6  Allie Quigley  CHI  36 26  635 121 270   54  119  47  49   17   52  69  60
##   STL BLK TO PTS DD2 TD3   FGP   FTP
## 1  28  40 46 584  17   0 44.42 87.56
## 2   3   2  9  15   0   0 23.08 50.00
## 3  25   5 35 189   0   0 32.18 88.24
## 4   5   5 41 188   0   0 42.66 91.67
## 5   7   6  6  21   0   0 23.53 25.00
## 6  13   7 30 343   0   0 44.81 95.92
```

```
WNBA %>%
  filter(PLAYER == "Tina Charles") %>%
  select(FGP, FTP)
```

```
##     FGP   FTP
## 1 44.91 82.03
```

Tina Charles' FGP is 44.91%, and their FTP is 82.03%.

d.)

```
WNBA %>%
  group_by(TEAM) %>%
  summarise(avg_REB = mean(REB, na.rm = TRUE), min_REB = min(REB, na.rm = TRUE),
            max_REB = max(REB, na.rm = TRUE)) %>%
  arrange(desc(avg_REB))
```

```
## # A tibble: 12 x 4
##    TEAM  avg_REB min_REB max_REB
##    <chr>   <dbl>   <int>   <int>
##  1 LVA     115.        0     298
##  2 CON     105        10     303
##  3 PHO     104.        4     302
##  4 CHI      98.9      11     193
##  5 DAL      95.8       3     173
##  6 SEA      93.9      19     267
##  7 NYL      93.5      21     171
##  8 MIN      93.3       4     312
##  9 ATL      89.5      14     219
## 10 WAS      86.6      13     258
## 11 IND      84.4       6     308
## 12 LAS      78         2     154
```

Team MIN has the max REB with an REB of 312.

e.)

```
WNBA = WNBA %>%
  group_by(TEAM) %>%
  mutate(FTP_fix = ifelse(is.na(FTP), FGP * mean(FTP, na.rm = TRUE), FTP))
WNBA$FTP_fix = round(WNBA$FTP_fix, digits = 2)

WNBA2 = WNBA %>%
  group_by(TEAM) %>%
  mutate(FTP_fix = ifelse(is.na(FTP), mean(FTP, na.rm = TRUE), FTP))
WNBA2$FTP_fix = round(WNBA2$FTP_fix, digits = 2)

head(WNBA)
```

```
## # A tibble: 6 x 24
## # Groups:   TEAM [6]
##   PLAYER TEAM    AGE     G   MIN   FGM   FGA  X3PM  X3PA   FTM   FTA  OREB  DREB
##   <chr>  <chr> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int>
## 1 A'ja ~ LVA      25    32  1021   207   466     1     1   169   193    63   235
## 2 Aaliy~ IND      23    14   119     6    26     1     7     2     4     5     7
## 3 Aari ~ ATL      23    30   493    56   174    32   104    45    51     9    40
## 4 Aeria~ MIN      28    14   309    61   143    11    35    55    60    13    38
## 5 Alann~ PHO      25    18   117     8    34     4    21     1     4     5    19
## 6 Allie~ CHI      36    26   635   121   270    54   119    47    49    17    52
```

```
## # ... with 11 more variables: REB <int>, AST <int>, STL <int>, BLK <int>,
## #   TO <int>, PTS <int>, DD2 <int>, TD3 <int>, FGP <dbl>, FTP <dbl>,
## #   FTP_fix <dbl>
```

```
head(WNBA2)
```

```
## # A tibble: 6 x 24
## # Groups:   TEAM [6]
##   PLAYER TEAM    AGE     G   MIN   FGM   FGA  X3PM  X3PA   FTM   FTA  OREB  DREB
##   <chr>  <chr> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int>
## 1 A'ja ~ LVA      25    32  1021   207   466     1     1   169   193    63   235
## 2 Aaliy~ IND      23    14   119     6    26     1     7     2     4     5     7
## 3 Aari ~ ATL      23    30   493    56   174    32   104    45    51     9    40
## 4 Aeria~ MIN      28    14   309    61   143    11    35    55    60    13    38
## 5 Alann~ PHO      25    18   117     8    34     4    21     1     4     5    19
## 6 Allie~ CHI      36    26   635   121   270    54   119    47    49    17    52
## # ... with 11 more variables: REB <int>, AST <int>, STL <int>, BLK <int>,
## #   TO <int>, PTS <int>, DD2 <int>, TD3 <int>, FGP <dbl>, FTP <dbl>,
## #   FTP_fix <dbl>
```

The first method assumes that FGP is not, unlike FTP, NaN. Additionally, both assumes that because the team average FTP is some number, it means that that player's FTP is the same number or that number multiplied by that player's FGP, when its possible that its either lower or higher. Moreover, in the multiplication method, the resulting value is exceedingly higher than the highest FTP, which would not make sense.

I personally feel like imputing missing data via the average is the best way as it predicts or approximates, based on the average, what the FTP of a player in a team, is. However, if that doesn't work, then removing that player's name and data entries is also possible, as the team average's FTP is not affected either way by that player.

Question 2

Before starting, the data from who.csv was tidied.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.8     v stringr 1.4.1
## v tidyr   1.2.0     v forcats 0.5.2
## v readr   2.1.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
who = read.csv("who.csv", sep = ",", header = TRUE)
who5 = who %>%
  pivot_longer(
    cols = new_sp_m014:newrel_f65,
    names_to = "key",
    values_to = "cases",
```

```
    values_drop_na = TRUE
  ) %>%
  mutate(
    key = stringr::str_replace(key, "newrel", "new_rel")
  ) %>%
  separate(key, c("new", "type", "sexage")) %>%
  select(-new, -iso2, -iso3) %>%
  separate(sexage, c("sex", "age"), sep = 1)
```

a.)

mutate(key = stringr::str_replace(key, "newrel", "new_rel")

This line is required for the data to be properly tidied because without it, the "key" column containing the column names from the data set as variables, would be inconsistent as the variables have information split by an underscore.

b.)

```
sum(is.na(who))
```

```
## [1] 329428
```

There are 329,428 entries that were removed as NA.

c.)

An explicit missing value is a missing entry found in the data set of a column, while an implicit missing value is a missing entry or column entirely from the data set itself; the column or entry is not found in the data set at all.

Implicit missing values include the 'Recency' column which should contain how recent the specific data set is, and since the current data is all 'new' cases, another implicit missing value is the data for 'old' cases. In other words, what's missing is the recency of the cases which would only be filled with 'new' or 'old', and the country, year, type, sex, age, and number of cases of entries regarded as 'old'.

d.)

```
str(who5)
```

```
## tibble [76,046 x 6] (S3: tbl_df/tbl/data.frame)
##  $ country: chr [1:76046] "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
##  $ year   : int [1:76046] 1997 1997 1997 1997 1997 1997 1997 1997 1997 1997 ...
##  $ type   : chr [1:76046] "sp" "sp" "sp" "sp" ...
##  $ sex    : chr [1:76046] "m" "m" "m" "m" ...
##  $ age    : chr [1:76046] "014" "1524" "2534" "3544" ...
##  $ cases  : int [1:76046] 0 10 6 3 5 2 0 5 38 36 ...
```

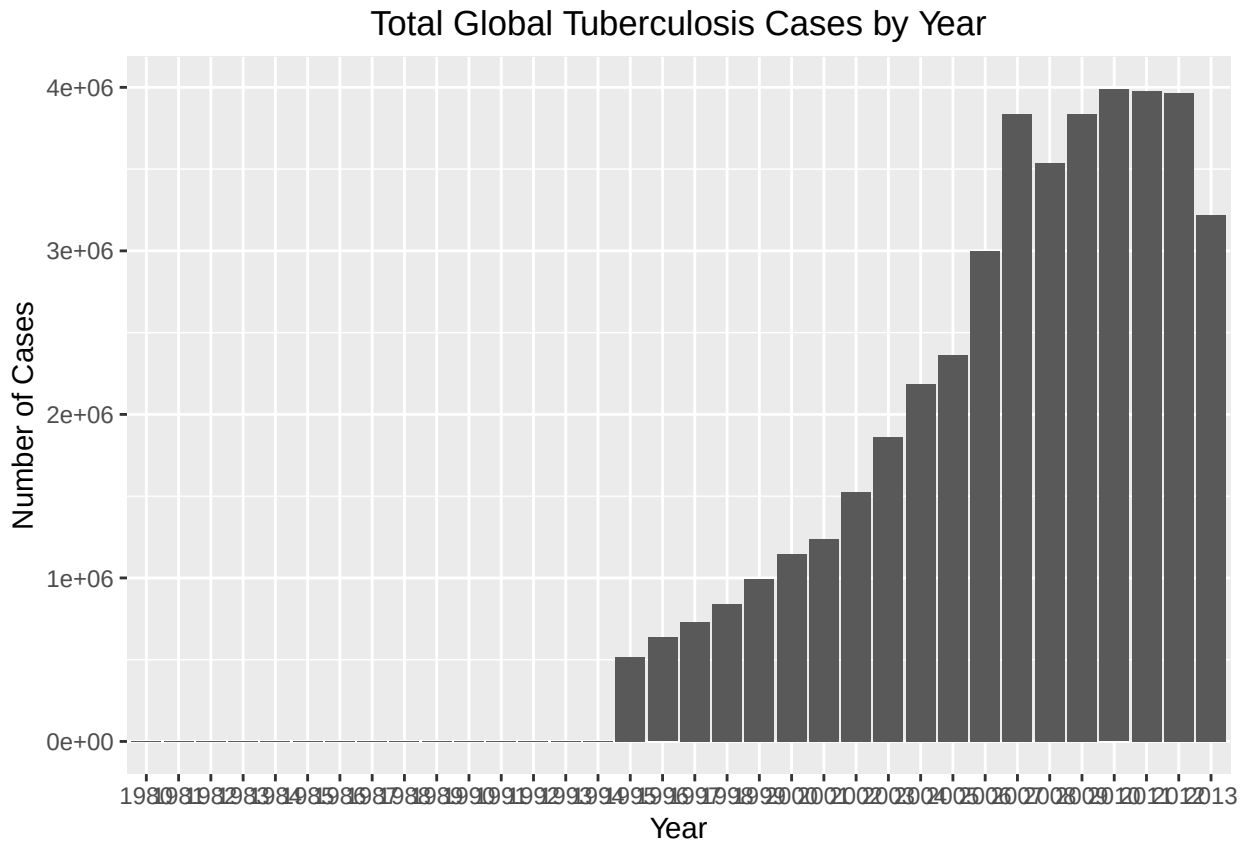
The 'sex' column entries should be capitalized instead of lower-cased, and the 'age' column entries should require a hyphen to better show the age range. Moreover, 'country' and 'year' would be better as a factor type, and 'sex' would be better as a Boolean type.

e.)

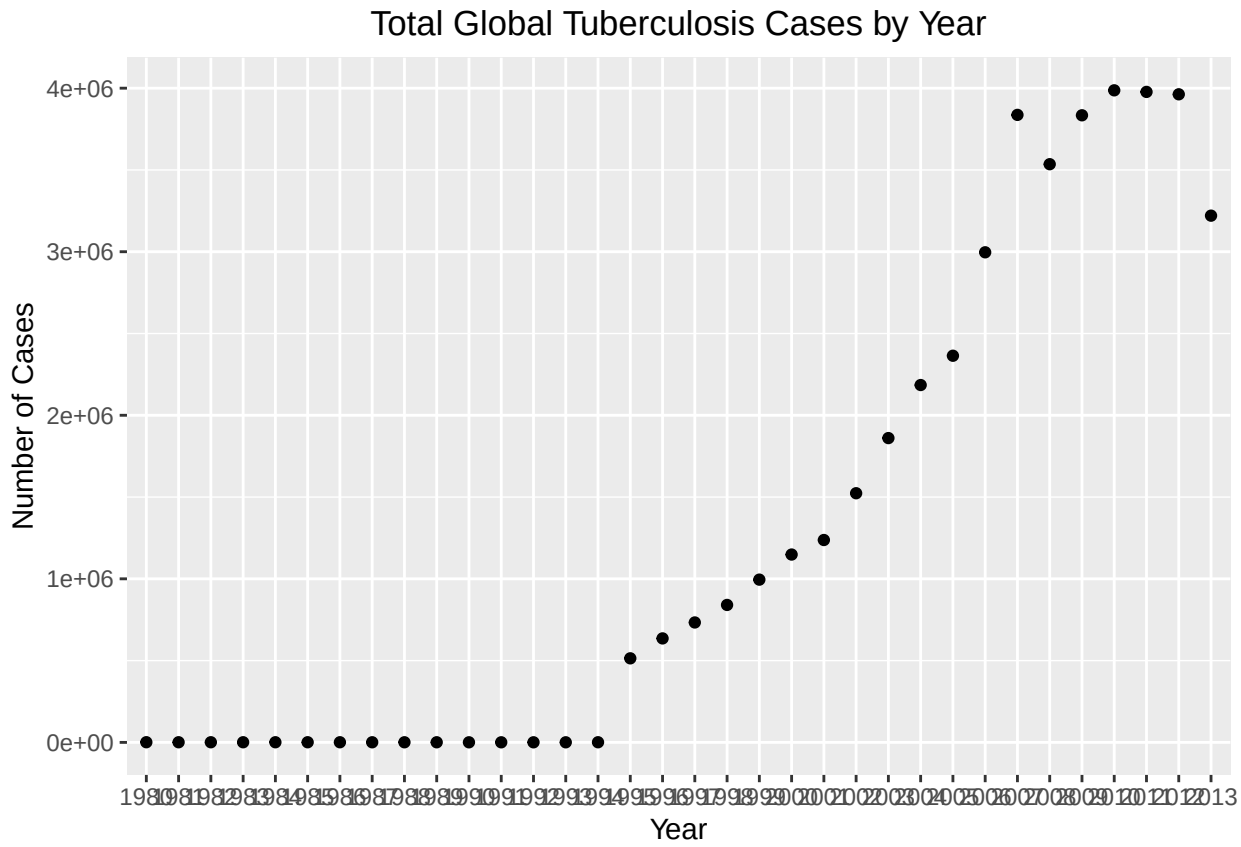
```
who5$year = as.factor(who5$year)

who6 = aggregate(who5$cases, by = list(group = who5$year), FUN = sum)

ggplot(who6, aes(x = group, y = x)) +
  geom_bar(stat="identity") +
  labs(x = "Year", y = "Number of Cases", title = "Total Global Tuberculosis Cases by Year") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
ggplot(who6, aes(x = group, y = x)) +
  geom_point() +
  labs(x = "Year", y = "Number of Cases", title = "Total Global Tuberculosis Cases by Year") +
  theme(plot.title = element_text(hjust = 0.5))
```



The graph shows the total recorded number of global cases of Tuberculosis sorted by year. It shows that the number of cases skyrockets between 1994 and 1995. But considering the amount of data removed as NA, especially between 1980 and 1994, it implies that the data is incomplete. This is more clearly shown in the two graphs, as in the bar graph, the data from 1980-1994 are very small in comparison to 1995-2013 and not even shown, while the scatter plot shows that data as practically 0. However, there is an exponential trend regardless of the missing data a trend curve could approximate the amount of cases between 1980 to 1994.

f.)

```
schqtr = read.csv("SchQtr.csv")

schqtr1 = schqtr %>%
  pivot_longer(
    cols = Qtr.1:Qtr.4,
    names_to = "Quarter",
    values_to = "Student_Count",
    values_drop_na = TRUE
  ) %>%
  mutate(Quarter = stringr::str_replace(Quarter, "Qtr_2", "Qtr.2")) %>%
  separate(Quarter, c("Interval_Type", "Interval_ID"))

str(schqtr1)
```

```
## tibble [48 x 5] (S3: tbl_df/tbl/data.frame)
```

```
##  $ School       : chr [1:48] "UNI" "UNI" "UNI" "UNI" ...
##  $ Year         : int [1:48] 2018 2018 2018 2018 2018 2018 2018 2018 2018 2018 ...
##  $ Interval_Type: chr [1:48] "Qtr" "Qtr" "Qtr" "Qtr" ...
##  $ Interval_ID  : chr [1:48] "1" "2" "3" "4" ...
##  $ Student_Count: int [1:48] 27 90 12 84 42 27 62 1 6 51 ...
```

The new data set has 48 rows.