Homework 13

Andrew Shao (NetID: as13381)

You are required to process the data `us_contagious_diseases` (available from package `dslabs`) via the 5 sequential steps given in the questions below.

Question 1 (1 pt): From the data `us_contagious_diseases`, ignoring the variable `weeks_reporting`, compute the yearly incidence rate of each disease for the entire country. Store the result into a new data frame, named as `US_incidence`, with columns `disease`, `year`, and `incidence_per_millon` (i.e., the yearly incidence rate times one million). Provide the output of `head(US_incidence)` and `dim(US_incidence)`. Note that you need to drop the missing values (`NA`) of `us_contagious_diseases` after deleting the variable `weeks_reporting`.

Answer:

```
US_incidence <- us_contagious_diseases %>%
  select(-weeks_reporting) %>%
  drop_na() %>%
  group_by(disease, year) %>%
  summarise(incidence_per_million = signif((sum(count) / sum(population)) * 10**6, 3))
```

```
## `summarise()` has grouped output by 'disease'. You can override using the
## `.groups` argument.
```

```
head(US_incidence)
```

```
## # A tibble: 6 x 3
## # Groups:   disease [1]
##   disease      year incidence_per_million
##   <fct>       <dbl>                 <dbl>
## 1 Hepatitis A  1966                   167
## 2 Hepatitis A  1967                   195
## 3 Hepatitis A  1968                   228
## 4 Hepatitis A  1969                   229
## 5 Hepatitis A  1970                   272
## 6 Hepatitis A  1971                   287
```

```
dim(US_incidence)
```

```
## [1] 315   3
```

Question 2 (1 pt): Pivot the data frame `US_incidence` into a new one that shows the `incidence_per_millon` values for all diseases of the same year at the same row. Still use the name `US_incidence` for the new data frame. Then provide the output of `head(US_incidence)` and `dim(US_incidence)`. Note that the output of `head(US_incidence)` should look like as follows:

```
## # A tibble: 6 x 8
##    year `Hepatitis A` Measles Mumps Pertussis  Polio Rubella Smallpox
##   <dbl>         <dbl>   <dbl> <dbl>     <dbl>  <dbl>   <dbl>    <dbl>
## 1  1966          167.   1036.    NA        NA  0.489    231.       NA
## 2  1967          195.    302.    NA        NA  0.219    215.       NA
## 3  1968          228.    115.   718.       NA  0.282    241.       NA
## 4  1969          229.    120.   405.       NA NA        262.       NA
## 5  1970          272.    225.   477.       NA NA        265.       NA
## 6  1971          287.    350.   556.       NA NA        207.       NA
```

Answer:

```
US_incidence <- US_incidence %>% pivot_wider(names_from = disease, values_from =
↪  incidence_per_million)

head(US_incidence)
```

```
## # A tibble: 6 x 8
##    year `Hepatitis A` Measles Mumps Pertussis  Polio Rubella Smallpox
##   <dbl>         <dbl>   <dbl> <dbl>     <dbl>  <dbl>   <dbl>    <dbl>
## 1  1966           167    1040    NA        NA  0.489     231       NA
## 2  1967           195     302    NA        NA  0.219     215       NA
## 3  1968           228     115   718        NA  0.282     241       NA
## 4  1969           229     120   405        NA NA         262       NA
## 5  1970           272     225   477        NA NA         265       NA
## 6  1971           287     350   556        NA NA         207       NA
```

```
dim(US_incidence)
```

```
## [1] 84  8
```

Question 3 (1 pt): Carefully read the documentation of the function `cor()` of package `stats`. According to the new data frame `US_incidence` from Question 2, compute the Pearson's correlation between `Hepatitis A` and `Measles` in terms of `incidence_per_millon`. Due to missing values, you need to choose an appropriate value for the `use` argument in `cor()`.

Answer:

```
cor(US_incidence$`Hepatitis A`, US_incidence$Measles, use = 'complete.obs')
```

```
## [1] 0.511971
```

Question 4 (1 pt): According to the new data frame `US_incidence` from Question 2, use one of the `map` functions to compute the Pearson's correlation between `Hepatitis A` and each of the other 6 diseases in terms of `incidence_per_millon`, and return a `double` vector as the output.

Answer:

```
map_dbl(select(US_incidence, -c(1, 2)), cor, y = US_incidence$`Hepatitis A`, use =
↪  'na.or.complete')
```

```
##    Measles      Mumps  Pertussis      Polio    Rubella   Smallpox
##  0.5119710  0.9121473 -0.6685702 -0.6997284  0.8836105         NA
```

Question 5 (1 pt): According to the new data frame `US_incidence` from Question 2, for each of the 7 diseases, find its most positively (Pearson's) correlated disease (except itself) and corresponding correlation. Simplify your code by loops or the `map` functions.

Answer:

```
max_cors <- tibble(disease = character(),
                   most_cor_disease = character(),
                   cor_coef = numeric())
for (disease in colnames(US_incidence[-1])) {
  cors <- map_dbl(select(US_incidence, -year, -disease), cor, y = US_incidence[disease],
↪  use = 'na.or.complete')
  mcor <- cors[which.max(cors)]
  max_cors <- max_cors %>%
    add_row(disease = disease,
            most_cor_disease = names(mcor),
            cor_coef = mcor)
}
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(disease)
##
##   # Now:
##   data %>% select(all_of(disease))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
max_cors
```

```
## # A tibble: 7 x 3
##   disease     most_cor_disease cor_coef
##   <chr>       <chr>               <dbl>
## 1 Hepatitis A Mumps               0.912
```

```
## 2 Measles     Pertussis          0.813
## 3 Mumps       Rubella            0.933
## 4 Pertussis   Measles            0.813
## 5 Polio       Rubella            0.350
## 6 Rubella     Mumps              0.933
## 7 Smallpox    Pertussis          0.641
```