covid_19_practice

Lang Liu

12/10/2022

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.5      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.1.4      v stringr 1.4.0
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
df <- read_csv("../US_counties_COVID19_health_weather_data_trimmed.csv")
```

```
## Rows: 3000 Columns: 227
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (8): county, state, fips, stay_at_home_announced, stay_at_home_effect...
## dbl  (215): cases, deaths, lat, lon, total_population, area_sqmi, population...
## lgl    (1): presence_of_water_violation
## date   (3): date, date_stay_at_home_announced, date_stay_at_home_effective
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
#select columns of interest
keep <- c("date","county","state","cases","deaths","total_population")
df_sub <- df %>% select(keep)
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use ‘all_of(keep)‘ instead of ‘keep‘ to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
#remove missing values
df_sub %>% summarise_all(~sum(is.na(.)))
```

```
## # A tibble: 1 x 6
##    date county state cases deaths total_population
##   <int>  <int> <int> <int>  <int>            <int>
## 1     0      0     0     0     44               50
```

```
df_sub <- df_sub %>% drop_na(deaths,total_population)
#discover the data in state Texas
df_sub %>%
  filter(state=="Texas") %>%
  arrange(county)
```

```
## # A tibble: 223 x 6
##    date       county    state cases deaths total_population
##    <date>     <chr>     <chr> <dbl>  <dbl>            <dbl>
##  1 2020-09-23 Anderson  Texas  2822     31            57772
##  2 2020-06-30 Angelina  Texas   476      6            87657
##  3 2020-05-27 Archer    Texas     1      0             8750
##  4 2020-05-09 Armstrong Texas     2      0             1913
##  5 2020-08-22 Austin    Texas   371      4            29107
##  6 2020-06-03 Bailey    Texas    19      0             7131
##  7 2020-05-05 Bandera   Texas     6      0            21015
##  8 2020-11-26 Baylor    Texas    54      4             3639
##  9 2020-04-13 Bee       Texas     2      0            32706
## 10 2020-11-21 Bee       Texas  1936     40            32706
## # ... with 213 more rows
```

```
#calculate total population in texas
total_texas <- df_sub %>%
  filter(state=="Texas") %>%
  arrange(county) %>%
  distinct(county,.keep_all=TRUE) %>%
  group_by(state) %>%
  summarise(total_texas = sum(total_population)) %>%
  select(total_texas)
total_texas
```

```
## # A tibble: 1 x 1
##   total_texas
##         <dbl>
## 1    19481937
```

```
total_texas = total_texas$total_texas
```

```
#Normalize the data with total population of Texas
#sort the data by date
df_sub %>%
  filter(state=="Texas") %>%
  group_by(date) %>%
  select(date,cases,deaths) %>%
  summarise_all(list(total=sum)) %>%
  mutate(cases_rate = cases_total/total_texas, death_rate = deaths_total/total_texas) %>%
  arrange(date)
```

```
## # A tibble: 147 x 5
##    date       cases_total deaths_total   cases_rate   death_rate
##    <date>           <dbl>        <dbl>        <dbl>        <dbl>
##  1 2020-03-14           2            0 0.000000103  0
##  2 2020-03-16           2            0 0.000000103  0
##  3 2020-03-23           5            0 0.000000257  0
##  4 2020-03-25           1            0 0.0000000513 0
##  5 2020-03-31           2            0 0.000000103  0
##  6 2020-04-10          14            1 0.000000719  0.0000000513
##  7 2020-04-13           2            0 0.000000103  0
##  8 2020-04-14          83            0 0.00000426   0
##  9 2020-04-15           8            2 0.000000411  0.000000103
## 10 2020-04-18          11            0 0.000000565  0
## # ... with 137 more rows
```