milestone 4

Sandya Krishna, Mai sedki, Lourdes Valdez

11/16/2020

R Markdown

```
#Loading libraries:
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(readr)
library(knitr)
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

We will use : Dataset: Infectious Diseases by County, Year and Sex (in California) 2001-2018 Source : https://data.chhs.ca.gov/dataset/infectious-disease

Step 1 : Calling dataset and cleaning data

```r
#Calling the dataset and naming dataframe id_data:
id_data <- read.csv("idb_odp_2001-2018 (1) (1).csv")
```

```r
# Now will look at the dataframe from the environment plus the first five rows
# using the head function. Noticed the following:
#The "rate" column has a lot of "dashes" or missing values because of 0 cases for
#some diseases. Will look at all the values to see using the function unique()
#unique(id_data$rate), and other than a dash, we see empty cells and "SC"
# Will now clean the data by eliminating empty values ," ", "-" or NA
# For correcting the NA values I will have to correct at the top of this code
# when I call the dataframe. I will do that now by adding to the code:
#na = (c(" ","-","SC","NA"))
```

```r
id_data <- read_csv("idb_odp_2001-2018 (1) (1).csv", na = (c(" ","-","SC","NA")))
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Disease = col_character(),
##   County = col_character(),
##   Year = col_double(),
##   Sex = col_character(),
##   Cases = col_double(),
##   Population = col_double(),
##   `Lower 95% CI` = col_double(),
##   `Upper 95% CI` = col_double(),
##   Rate = col_character()
## )
```

```r
# Change the column names to be more r friendly,using the function rename  column names will changeto a
id_data <- rename_with(id_data, ~ tolower(gsub(" ","_",.x,fixed=TRUE)))
```

```r
# next We would like to do the following changes
# to the rate column: change it to numeric, round it to 2 decimal values and d/c
# the asterisk "*"
id_data$rate <- as.numeric(as.character(id_data$rate))
```

```
## Warning: NAs introduced by coercion
```

```r
# I view the table and run str() and now rate is numeric and asterisk is gone.
# Now will round to 2 decimals, will also round the CI to 2 decimals
id_data$rate <- round(id_data$rate,2)
id_data$`lower_95%_ci` <- round(id_data$`lower_95%_ci`,2)
id_data$`upper_95%_ci` <- round(id_data$`upper_95%_ci`,2)
```

Step 2 Creating variables for data analysis: *The new variables we will create will be:*
"region" for the 10 different california regions
*ID_type" to group each of the listed diseases by "type of disease" , following conventional microbiology classification

```
#Creating the first variable "region"
#We will group the counties that correspond to the California regions, as done
#by the Census bureau: For census purposes, California is divided in 10
#regions https://census.ca.gov/regions/
#First I will use  function unique() to see all the values

#Next  will create vectors with function c() to group into the appropiate region
#looking carefully on how counties are spelled
region1 <- c("NEVADA","PLACER","PLUMAS","SACRAMENTO","SHASTA","SIERRA",
"SISKIYOU","SUTTER","TEHAMA", "YOLO", "YUBA", "MODOC", "EL DORADO", "BUTTE",
"GLENN", "LASSEN")
region2 <- c("DEL NORTE", "HUMBOLDT", "LAKE", "MENDOCINO", "NAPA","SONOMA",
"TRINITY")
region3 <- c("ALAMEDA","CONTRA COSTA", "MARIN", "SAN FRANCISCO", "SAN MATEO",
             "SANTA CLARA", "SOLANO")
region4 <- c("ALPINE", "AMADOR", "CALAVERAS", "MADERA", "MARIPOSA",
"MERCED", "MONO","SAN JOAQUIN", "STANISLAUS", "TUOLUMNE")
region5 <- c("MONTEREY", "SAN BENITO", "SAN LUIS OBISPO", "SANTA BARBARA",
             "SANTA CRUZ", "VENTURA")
region6 <- c("FRESNO","INYO", "KERN", "KINGS", "TULARE")
region7 <- c("RIVERSIDE", "SAN BERNARDINO")
region8 <- c("LOS ANGELES")
region9 <- c("ORANGE")
region10 <- c("IMPERIAL", "SAN DIEGO")
California <- c("CALIFORNIA")
```

```
#Now will create new column using function mutate and add to current table,
I #noticed aside for counties there is a "CALIFORNIA" value, will add that
```

```
## function (x)
## {
##     structure(x, class = unique(c("AsIs", oldClass(x))))
## }
## <bytecode: 0x556ecdfd3788>
## <environment: namespace:base>
```

```
IDtable1 <- id_data %>% mutate(region = case_when(
  county %in% region1 ~ "Superior",
  county %in% region2 ~ "N_coast",
  county %in% region3 ~ "Bay_area",
  county %in% region4 ~ "NSJ_valley",
  county %in% region5 ~ "C_coast",
  county %in% region6 ~ "SSJ_valley",
  county %in% region7 ~ "I_Empire",
  county %in% region8 ~ "LA_county",
  county %in% region9 ~ "O_county",
  county %in% region10 ~ "SDI_county",
  county %in% California ~ "California",
  TRUE ~ NA_character_
)) %>%
  drop_na(region)
```

```r
#it worked new column formed  called "group" on ID_table1
```

```r
# Creating the second variable "ID_type"
# Our group decided to create the seven following groups:
parasitic <- c("Amebiasis","Babesiosis", "Cryptosporidiosis", "Cyclosporiasis",
"Cysticercosis or Taeniasis", "Malaria", "Giardiasis", "Trichinosis")
toxin_related <- c("Botulism, Foodborne","Botulism, Other", "Botulism, Wound",
"Ciguatera Fish Poisoning", "Domoic Acid Poisoning","Paralytic Shellfish
Poisoning", "Scombroid Fish Poisoning")
viral <- c("Chikungunya Virus Infection", "Dengue Virus Infection","Flavivirus
Infection of Undetermined Species","Hantavirus Infection","Hepatitis E acute
infection","Rabies, human","Yellow Fever", "Yersiniosis","Zika Virus Infection")
prions <- c("Creutzfeldt-Jakob Disease and other Transmissible Spongiform
Encephalopathies")
fungal <- c("Coccidioidomycosis")
bacterial <- c("Anaplasmosis", "Anaplasmosis and Ehrlichiosis", "Anthrax",
"Brucellosis", "Campylobacteriosis","Cholera","E. coli O157","E. coli Other STEC
(non-O157)", "Legionellosis","Leprosy (Hansen's Disease)", "Leptospirosis",
"Listeriosis", "Lyme Disease","Plague, human","Q Fever","Spotted Fever
Rickettsiosis", "Streptococcal Infection (cases in food and dairy workers)",
"Ehrlichiosis", "Psittacosis", "Salmonellosis", "Shigellosis", "Tularemia",
"Typhoid Fever", "Paratyphoid Fever", "Typhus Fever", "Relapsing Fever", "Shiga
toxin-producing E. coli (STEC) without Hemolytic Uremic Syndrome (HUS)",
"Vibrio Infection (non-Cholera)", "Shiga Toxin Positive Feces (without culture
confirmation)")
infectious_complications <- c("Hemolytic Uremic Syndrome (HUS) without evidence
of Shiga toxin-producing E. coli (STEC)","Hemolytic Uremic Syndrome (HUS)",
"Shiga toxin-producing E. coli (STEC) with Hemolytic Uremic Syndrome (HUS)")

# now I will create a new table naming it IDtable2 with new column ID_type
IDtable2 <- IDtable1 %>% mutate(ID_type = case_when(
  disease %in% parasitic ~ "Parasitic",
  disease %in% toxin_related ~ "Toxin_related",
  disease %in% viral ~ "Viral",
  disease %in% prions ~ "Prions",
  disease %in% fungal ~ "Fungal",
  disease %in% bacterial ~ "Bacterial",
  disease %in% infectious_complications ~ "ID_complication",
  TRUE ~ NA_character_
)) %>%
  drop_na(ID_type)
```

```r
#Create ID_tableyears_grouptotal . Grouping every 3 years as a single period and change years to charac
ID_tableyears_group_total <- IDtable2 %>%
  mutate(year_range =case_when(
    year %in% c(2001,2002,2003)~ "2001-2003",
    year %in% c(2004,2005,2006)~ "2004-2006",
    year %in% c(2007,2008,2009)~ "2007-2009",
    year %in% c(2010,2011,2012)~ "2010-2012",
    year %in% c(2013,2014,2015)~ "2013-2015",
    year %in% c(2016,2017,2018)~ "2016-2018",
     )) %>%
  filter(sex=="TOTAL")
```