shiny app code

Sarah Sandmaier

2025-04-02

data cleaning for shiny app

CovState <- **read.csv**("Weekly\_United\_States\_COVID-19\_Cases\_and\_Deaths\_by\_State\_-\_ARCHIVED\_20250304\_cleaned.csv")  
  
*#change format to dates*  
CovState**$**date\_updated <- **as.Date**(CovState**$**date\_updated, format = "%Y-%m-%d")  
  
*#change states to their names*  
CovState <- CovState **%>%**  
 **mutate**(state = **case\_when**(  
 state **==** "AK" **~** "alaska",  
 state **==** "AL" **~** "alabama",  
 state **==** "AR" **~** "arkansas",  
 state **==** "AZ" **~** "arizona",  
 state **==** "CA" **~** "california",  
 state **==** "CO" **~** "colorado",  
 state **==** "CT" **~** "connecticut",  
 state **==** "DE" **~** "delaware",  
 state **==** "FL" **~** "florida",  
 state **==** "GA" **~** "georgia",  
 state **==** "ID" **~** "idaho",  
 state **==** "IL" **~** "illinois",  
 state **==** "IN" **~** "indiana",  
 state **==** "IA" **~** "iowa",  
 state **==** "KS" **~** "kansas",  
 state **==** "KY" **~** "kentucky",  
 state **==** "LA" **~** "louisiana",  
 state **==** "ME" **~** "maine",  
 state **==** "MD" **~** "maryland",  
 state **==** "MA" **~** "massachusetts",  
 state **==** "MI" **~** "michigan",  
 state **==** "MN" **~** "minnesota",  
 state **==** "MS" **~** "mississippi",  
 state **==** "MO" **~** "missouri",  
 state **==** "MT" **~** "montana",  
 state **==** "NE" **~** "nebraska",  
 state **==** "NV" **~** "nevada",  
 state **==** "NH" **~** "new hampshire",  
 state **==** "NJ" **~** "new jersey",  
 state **==** "NM" **~** "new mexico",  
 state **==** "NY" **~** "new york",  
 state **==** "NC" **~** "north carolina",  
 state **==** "ND" **~** "north dakota",  
 state **==** "OH" **~** "ohio",  
 state **==** "OK" **~** "oklahoma",  
 state **==** "OR" **~** "oregon",  
 state **==** "PA" **~** "pennsylvania",  
 state **==** "RI" **~** "rhode island",  
 state **==** "SC" **~** "south carolina",  
 state **==** "SD" **~** "south dakota",  
 state **==** "TN" **~** "tennessee",  
 state **==** "TX" **~** "texas",  
 state **==** "UT" **~** "utah",  
 state **==** "VT" **~** "vermont",  
 state **==** "VA" **~** "virginia",  
 state **==** "WA" **~** "washington",  
 state **==** "WV" **~** "west virginia",  
 state **==** "WI" **~** "wisconsin",  
 state **==** "WY" **~** "wyoming",  
 TRUE **~** state  
 ))   
  
*# delete states that dont appear in US map: AS, DC, FSM, GU, HI, MP, NYC, PR, PW, RMI, VI*  
CovState <- CovState **%>%**   
 **filter**(**!**state **%in%** **c**("AS", "DC","FSM","GU","HI","MP","NYC","PR","PW","RMI","VI"))  
  
*#US map data*  
us\_map <- **map\_data**("state")  
  
*#save cleaned data as csv*  
  
**write.csv**(CovState, "/Users/sarita/Documents/UVIC/STAT/STAT 321/visuals/CovState.csv", row.names = FALSE)

shiny app creation

CovState <- **read.csv**("CovState.csv")  
  
**library**(leaflet)  
**library**(tidyr)  
**library**(sf)   
**library**(dplyr)   
**library**(maps)   
**library**(shiny)   
**library**(viridis)   
**library**(gridExtra)  
**library**(tmap)  
**library**(rnaturalearth)   
**library**(rnaturalearthdata)  
  
*# function for map*  
covid\_den\_map <- **function**(start\_date, end\_date) {  
 start\_date <- **as.Date**(start\_date)  
 end\_date <- **as.Date**(end\_date)  
 filtered\_data <- CovState[CovState**$**date\_updated **>=** start\_date **&** CovState**$**date\_updated **<=** end\_date, ]  
 total\_new\_cases\_by\_state <- **tapply**(filtered\_data**$**new\_cases, filtered\_data**$**state, sum, na.rm = TRUE)  
 total\_new\_deaths\_by\_state <- **tapply**(filtered\_data**$**new\_deaths, filtered\_data**$**state, sum, na.rm = TRUE)  
 us\_states <- **st\_as\_sf**(maps**::map**("state", fill = TRUE, plot = FALSE))  
 us\_states**$**state <- **tolower**(us\_states**$**ID)   
 us\_states**$**new\_cases <- total\_new\_cases\_by\_state[us\_states**$**state]   
 us\_states**$**new\_deaths <- total\_new\_deaths\_by\_state[us\_states**$**state]  
 map\_cases <- **ggplot**(data = us\_states) **+**  
 **geom\_sf**(**aes**(fill = new\_cases)) **+**  
 **scale\_fill\_viridis\_c**(option = "mako", trans = "log", direction = **-**1) **+**   
 **theme\_minimal**() **+**  
 **labs**(title = **paste**("COVID-19 New Cases by State**\n**from", start\_date, "to", end\_date),  
 fill = "New Cases") **+**  
 **theme**(  
 axis.text = **element\_blank**(),  
 axis.ticks = **element\_blank**(),  
 axis.title = **element\_blank**()  
 )  
 map\_deaths <- **ggplot**(data = us\_states) **+**  
 **geom\_sf**(**aes**(fill = new\_deaths)) **+**  
 **scale\_fill\_viridis\_c**(option = "magma", trans = "log",direction = **-**1) **+**   
 **theme\_minimal**() **+**  
 **labs**(title = **paste**("COVID-19 New Deaths by State**\n**from", start\_date, "to", end\_date),  
 fill = "New Deaths") **+**  
 **theme**(  
 axis.text = **element\_blank**(),  
 axis.ticks = **element\_blank**(),  
 axis.title = **element\_blank**()  
 )  
 **grid.arrange**(map\_cases, map\_deaths, nrow = 2)  
}  
  
  
ui <- **fluidPage**(  
 **titlePanel**("COVID-19 Data by State"),  
 **sidebarLayout**(  
 **sidebarPanel**(  
 **dateInput**(  
 inputId = "start\_date",  
 label = "Select Start Date",  
 value = "2020-01-23",   
 min = "2020-01-23",   
 max = "2023-05-11"  
 ),  
 **dateInput**(  
 inputId = "end\_date",  
 label = "Select End Date",  
 value = "2022-01-01",   
 min = "2020-01-23",   
 max = "2023-05-11"   
 )  
 ),  
 **mainPanel**(  
 **plotOutput**("covid\_map")  
 )  
 )  
)  
  
  
server <- **function**(input, output) {  
 output**$**covid\_map <- **renderPlot**({  
   
 **covid\_den\_map**(input**$**start\_date, input**$**end\_date)  
   
 })  
}  
  
  
**shinyApp**(ui = ui, server = server)