

Miniproject 3: Mapping COVID Worldwide

Zane Shango

3/27/2020

Guidelines

This homework is to be submitted to the relevant d2l Dropbox folder by 12pm on **April 2**. You may work individually or in a team of at most three people. If you work in a team, please just turn in one assignment and list all names at the top of the assignment. You are allowed to consult any online resource. If you do so, please include a link to the website that you used in your script. If you use code from my lectures, there's no need to include a citation.

Data

The only data file for this assignment is `countries-aggregated.csv` and it is up on d2l in the Homework 3 folder. This data set contains the number of confirmed, recovered, and fatal COVID cases by country and time.

Q1

For each country calculate the number of confirmed, recovered and fatal cases of COVID-19 by month. This will leave you with three observations (Jan, Feb, and Mar) by country.

```
# filter and group by to get 3 month observations per country

totals <- legis %>%
  group_by(Country, Month, Month_Name) %>%
  summarize(
    Total_Confirmed = sum(Confirmed),
    Total_Recovered = sum(Recovered),
    Total_Deaths = sum(Deaths)
  )

# print out the summary table
totals
```

Country <fctr>	M... <dbl>	Month_N... <chr>	Total_Confirmed <int>	Total_Recovered <int>	Total_Deaths <int>
Afghanistan	1	January	0	0	
Afghanistan	2	February	6	0	
Afghanistan	3	March	1219	26	
Albania	1	January	0	0	

4/1/2020

Miniproject 3: Mapping COVID Worldwide

Country <fctr>	M... <dbl>	Month_N... <chr>	Total_Confirmed <int>	Total_Recovered <int>	Total_Deaths <int>							
Albania	2	February	0	0								
Albania	3	March	2232	241								
Algeria	1	January	0	0								
Algeria	2	February	5	0								
Algeria	3	March	4823	594	5							
Andorra	1	January	0	0								
1-10 of 540 rows			Previous	1	2	3	4	5	6	...	54	Next

Q2

Develop a set of spatial visualizations that show the change in COVID-19 cases by country over time. You should have separate visualizations for the confirmed, recovered, and fatal variables.

```
## [1] 180
```

```
## [1] 252
```

```
# confirmed visualization
confirmed_viz <- ggplot(data = totals, aes(map_id = Country)) +
  geom_map(map = world_map,
    aes(fill = Total_Confirmed),
    color = 'black') +
  expand_limits(x = world_map$long, y = world_map$lat) +
  theme_void() +
  scale_fill_gradientn(colours = c('white', 'yellow', 'orange', 'red'),
    values = scales::rescale(c(0,.05,.5,1,1.5,2,4,10,14,18)), na.valu
e = 'white', labels = comma) +
  labs(
    fill = "Total Confirmed\nCOVID-19 Cases",
    title = "Global Confirmed COVID-19 Cases") +
  theme(legend.position = "left") +
  facet_wrap(~Month_Name, nrow = 3) +
  theme(
    legend.text = element_text(size = 15),
    legend.title = element_text(size = 15),
    # legend.key.height = unit(3, 'line'),
    strip.text = element_text(size = 13),
    plot.title = element_text(size = 25, hjust = .5),
    panel.border = element_rect(linetype = 'longdash', fill = NA),
    panel.background = element_rect(fill = 'grey90'),
    plot.margin = unit(c(1,1,1,1), "cm")
  )

confirmed_viz
```

Global Confirmed COVID-19 Cases

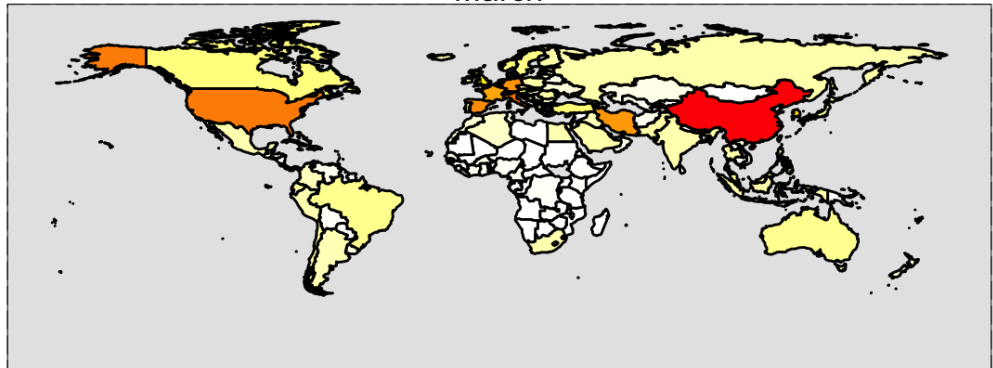
January



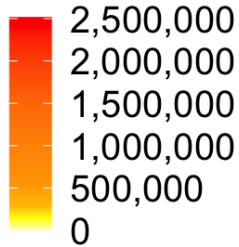
February



March



Total Confirmed
COVID-19 Cases



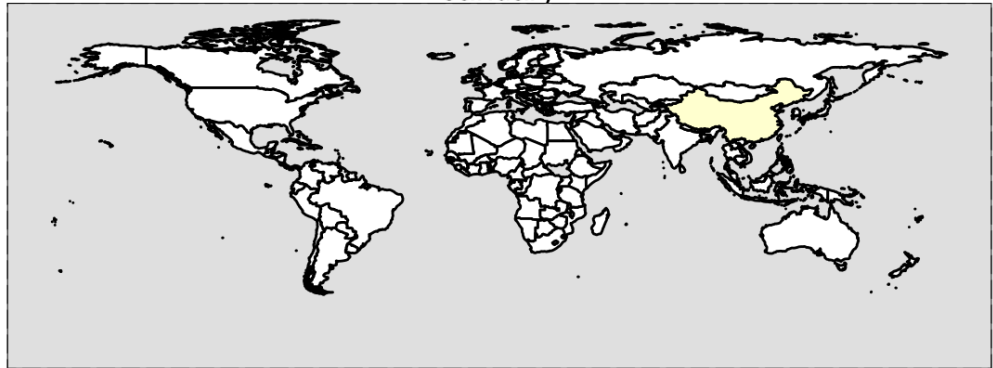
how to stretch out the colorbar/legend

```
# recovered visualization
recovered_viz <- ggplot(data = totals, aes(map_id = Country)) +
  geom_map(map = world_map,
    aes(fill = Total_Recovered),
    color = 'black') +
  expand_limits(x = world_map$long, y = world_map$lat) +
  theme_void() +
  scale_fill_gradientn(colours = c('white', 'yellow', 'green', 'darkgreen'),
    values = scales::rescale(c(0,.001,.01, .05, .1, .5, 1)), na.value
= 'white', labels = comma) +
  labs(
    fill = "Total Recovered\nCOVID-19 Cases",
    title = "Global Recovered COVID-19 Cases") +
  theme(legend.position = "left") +
  facet_wrap(~Month_Name, nrow = 3) +
  theme(
    legend.text = element_text(size = 15),
    legend.title = element_text(size = 15),
    strip.text = element_text(size = 13),
    plot.title = element_text(size = 25, hjust = .5),
    panel.border = element_rect(linetype = 'longdash', fill = NA),
    panel.background = element_rect(fill = 'grey90'),
    plot.margin = unit(c(1,1,1,1), "cm")
  )

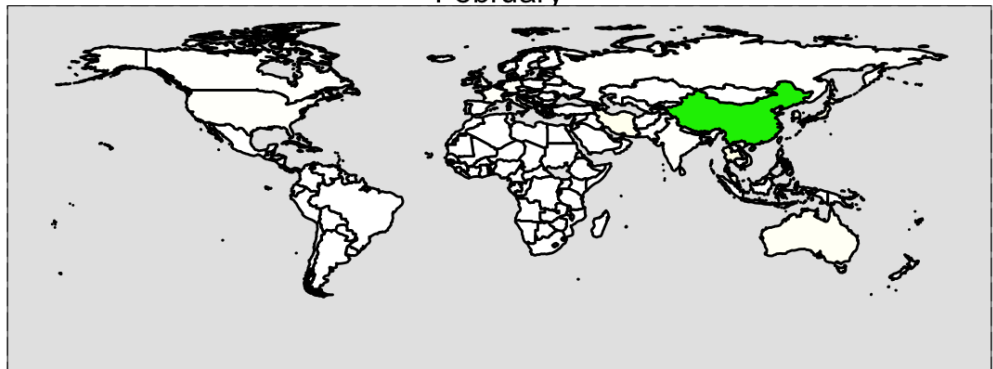
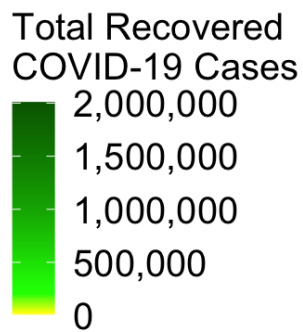
recovered_viz
```

Global Recovered COVID-19 Cases

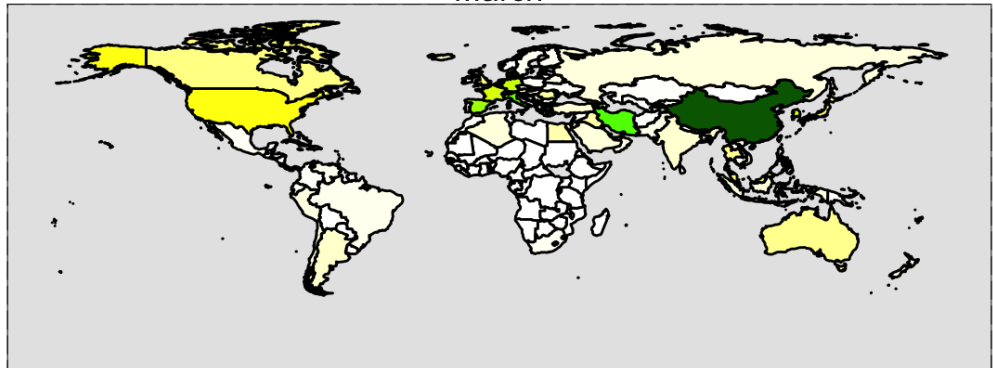
January



February



March



```
# death visualization
deaths_viz <- ggplot(data = totals, aes(map_id = Country)) +
  geom_map(map = world_map,
    aes(fill = Total_Deaths),
    color = 'black') +
  expand_limits(x = world_map$long, y = world_map$lat) +
  theme_void() +
  scale_fill_gradientn(colours = c('white', 'grey70', 'grey30', 'black'),
    values = scales::rescale(c(0,.001,.01, .05, .1, .5, 1)), na.value
= 'white', labels = comma) +
  labs(
    fill = "Total COVID-19 \nCase Deaths",
    title = "Global COVID-19 Case Deaths") +
  theme(legend.position = "left") +
  facet_wrap(~Month_Name, nrow = 3) +
  theme(
    legend.text = element_text(size = 15),
    legend.title = element_text(size = 15),
    strip.text = element_text(size = 13),
    plot.title = element_text(size = 25, hjust = .5),
    panel.border = element_rect(linetype = 'longdash', fill = NA),
    panel.background = element_rect(fill = 'white'),
    plot.margin = unit(c(1,1,1,1), "cm")
  )

deaths_viz
```

Global COVID-19 Case Deaths

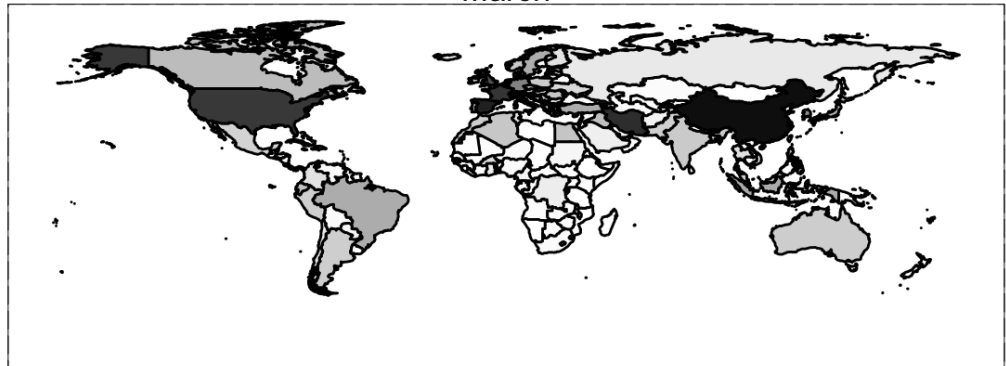
January



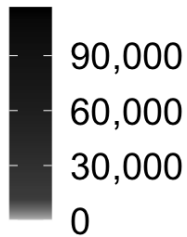
February



March



Total COVID-19
Case Deaths



Q3: Bonus question

Pick any country in the world. Find COVID-19 statistics by some administrative unit (e.g., states in the United States), and develop a spatial visualization to show any COVID related statistics (e.g., number of confirmed cases, number tested).

Further points will be awarded if you are able to show change in number of COVID cases over time.


```
library(transformr)
library(gganimate)

state_map <- map_data("state")

# load in the US county data
us_state = read.csv("https://raw.githubusercontent.com/nytimes/covid-19-data/master/us-states.csv", header = TRUE)
```

```
# need to change us_state to all lowercase, to make formatting of state_map
us_state$state <- tolower(us_state$state)

# see which states from us_state are missing from state_map (just alaska, hawaii, and territories)
setdiff(us_state$state, state_map$region)
```

```
## [1] "hawaii"           "alaska"
## [3] "puerto rico"     "virgin islands"
## [5] "guam"             "northern mariana islands"
```

```
# do same as over dataset, make first column dates, add month column, create a summary table
us_state$date = as.Date(us_state$date, "%Y-%m-%d")

us_state$month = as.numeric(format(us_state$date, "%m"))
us_state <- us_state %>%
  mutate(month_name = ifelse(month == 1, "January",
    ifelse(month == 2, "February",
    ifelse(month == 3, "March", NA))))

# summary table

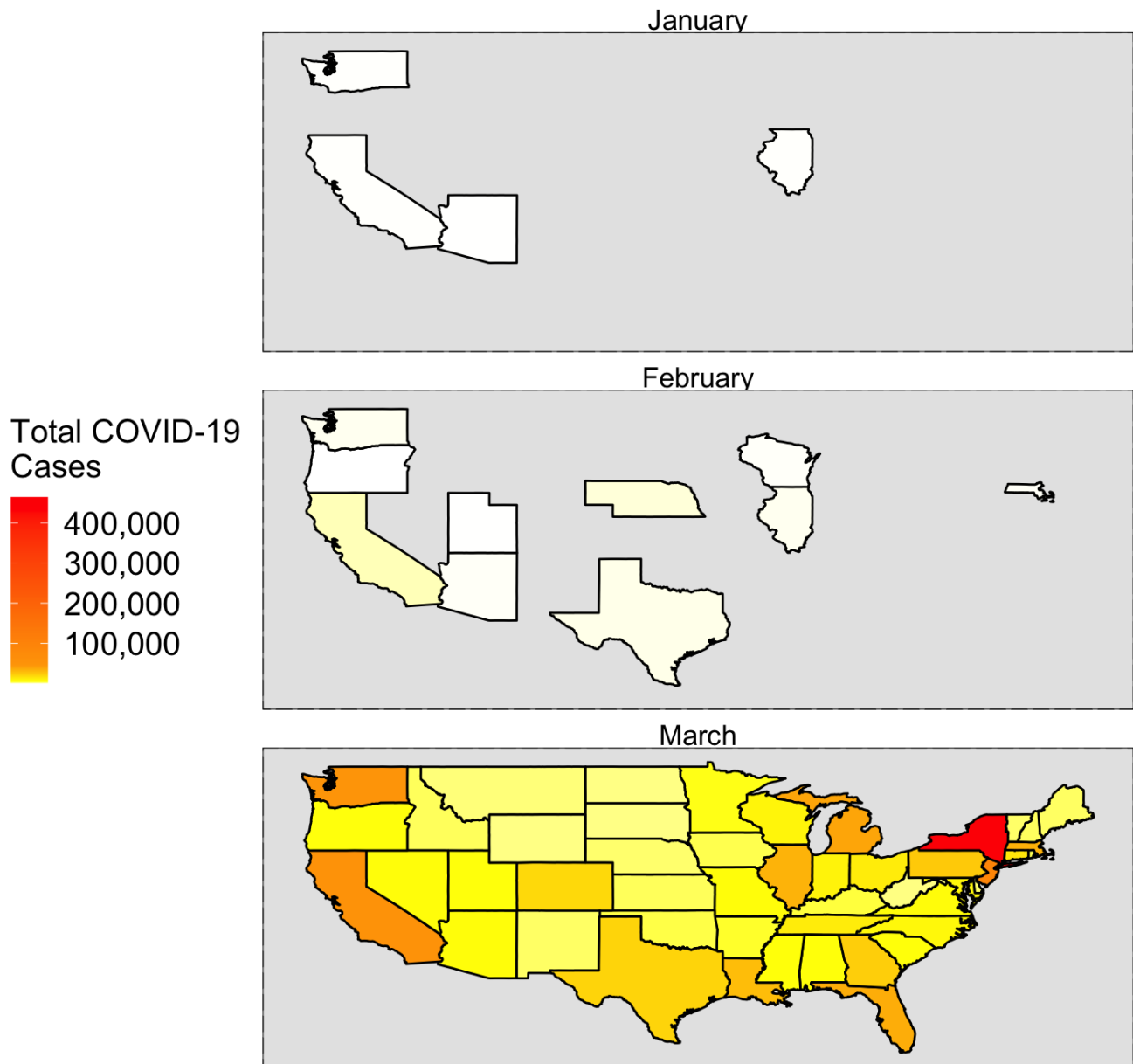
state_totals <- us_state %>%
  group_by(state, month_name) %>%
  summarize(
    total_cases = sum(cases),
    total_deaths = sum(deaths)
  )

# need to set months as factors, in order for them to plot correctly
state_totals$month_name <- factor(state_totals$month_name, levels = c("January", "February", "March"))
```

US State Monthly COVID-19 Cases

```
us_viz <- ggplot(data = state_totals, aes(map_id = state)) +  
  geom_map(map = state_map,  
    aes(fill = total_cases),  
    color = 'black') +  
  expand_limits(x = state_map$long, y = state_map$lat) +  
  theme_void() +  
  scale_fill_gradientn(colours = c('white', 'yellow', 'orange', 'red'),  
    values = scales::rescale(c(0,.001,.01, .05, .1, .5, 1)), na.value  
= 'white', labels = comma) +  
  labs(  
    fill = "Total COVID-19 \nCases",  
    title = "US State COVID-19 Cases") +  
  theme(legend.position = "left") +  
  facet_wrap(~month_name, nrow = 3) +  
  theme(  
    legend.text = element_text(size = 15),  
    legend.title = element_text(size = 15),  
    strip.text = element_text(size = 13),  
    plot.title = element_text(size = 25, hjust = .5),  
    panel.border = element_rect(linetype = 'longdash', fill = NA),  
    panel.background = element_rect(fill = 'grey90'),  
    plot.margin = unit(c(1,1,1,1), "cm")  
  )  
  
us_viz
```

US State COVID-19 Cases



This US state visualization was quite frustrating, I attempted to use gganimate to animate the plot but I found that very difficult to do with two separate dataframes. I also wanted to use the US county data to show density in states around big cities, but there were a lot of counties spelled with different formatting, which would have been a lot to fix. Finally, This plot is only displaying a state once there is a case there which was not my intention, but it is interesting to see even moreso how quickly the cases have risen.

March Daily US State COVID-19 Cases

```

us_state$day = as.numeric(format(us_state$date, "%d"))

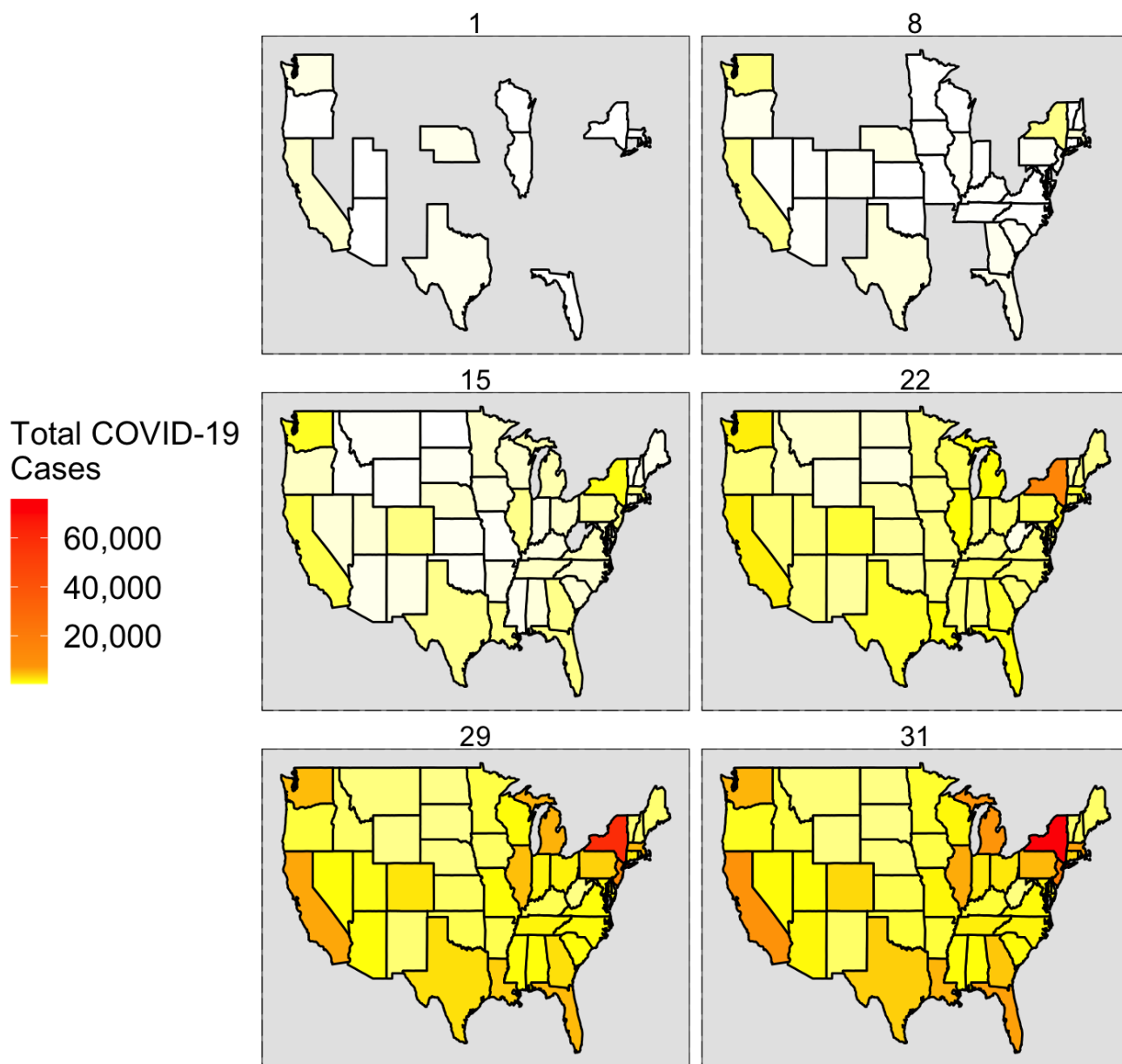
march_daily <- us_state %>%
  filter(month_name == "March") %>%
  filter(day %in% c(1,8,15,22,29,31)) %>%
  group_by(state, day) %>%
  summarize(
    total_cases = sum(cases),
    total_deaths = sum(deaths)
  )

# us state cases month of march
march_viz <- ggplot(data = march_daily, aes(map_id = state)) +
  geom_map(map = state_map,
    aes(fill = total_cases),
    color = 'black') +
  expand_limits(x = state_map$long, y = state_map$lat) +
  theme_void() +
  scale_fill_gradientn(colours = c('white', 'yellow', 'orange', 'red'),
    values = scales::rescale(c(0,.001,.01, .05, .1, .5, 1)), na.value
= 'white', labels = comma) +
  labs(
    fill = "Total COVID-19 \nCases",
    title = "March US State COVID-19 Cases") +
  theme(legend.position = "left") +
  facet_wrap(~day, nrow = 3) +
  theme(
    legend.text = element_text(size = 15),
    legend.title = element_text(size = 15),
    strip.text = element_text(size = 13),
    plot.title = element_text(size = 25, hjust = .5),
    panel.border = element_rect(linetype = 'longdash', fill = NA),
    panel.background = element_rect(fill = 'grey90'),
    plot.margin = unit(c(1,1,1,1), "cm")
  )

march_viz

```

March US State COVID-19 Cases



This plot shows the daily growth in cases over the month of March. The dates chosen were Sundays and the most recent with data recorded.