Project

Andrew Shao

2024-11-13

Introduction

The COVID-19 pandemic has had profound impacts on public health and policymaking across the United States. Understanding the dynamics of case and death rates is critical for designing effective interventions. This analysis investigates how varying temporal resolutions (daily, weekly, and monthly) and geographical scales (county, state, and national levels) influence the interpretation of COVID-19 data. By comparing these dimensions, this study aims to uncover patterns that may remain hidden when data is aggregated differently, thereby providing insights into the optimal levels of granularity for monitoring and response planning.

To address this problem, datasets from The New York Times GitHub repository will be utilized, specifically focusing on national, state, and county-level COVID-19 data. The analysis will involve data cleaning, aggregation, and visualization to explore the interplay between timeframes and geographical granularity. The findings aim to guide researchers and policymakers in tailoring their analytical approaches to the complexity of public health data, ultimately supporting informed decision-making during public health crises.

Packages Required

For this analysis I used the following packages: `tidyverse`, `USAboundaries`, `lubridate`, and `knitr`. I used several packages included in `tidyverse` including `dplyr`, `tibble`, and `tidyr` to organize and clean the data and `ggplot2` to draw plots. Additionally, the county data in the package `USAboundaries` was used to verify the names of the counties within the New York Times datasets. `lubridate` was used to manipulate the dates within the data for temporal analysis.

```r
# install.packages('remotes')
# remotes::install_github("ropensci/USAboundaries")
# install.packages("USAboundariesData", repos = "https://ropensci.r-universe.dev", type =
↪  "source")

library(tidyverse)
library(USAboundaries)
library(lubridate)
```

Data Preparation

The dataset used in this analysis originates from The New York Times' COVID-19 Data Repository, which was created to provide a comprehensive and up-to-date record of COVID-19 cases and deaths across the United States. The data spans from the onset of the pandemic in early 2020 through March 23, 2023, and is regularly updated to reflect cumulative statistics. It includes three main datasets representing different geographical scales: national, state, and county levels. Each dataset contains variables such as date, geographical location (e.g., state or county names), cumulative case counts, and cumulative death counts. Notably, the data captures daily updates, allowing for temporal aggregation into weekly or monthly intervals. Missing values are recorded as blanks or "Unknown", and no explicit imputation has been performed within the raw datasets. Users of the data must account for these gaps during analysis, as they may reflect unreported cases or delays in data collection. Additionally, the county-level dataset integrates information from multiple files to provide a consistent and unified view, potentially introducing discrepancies that should be carefully addressed during preprocessing. Overall, this dataset serves as a valuable resource for tracking the trajectory of the pandemic and analyzing its impacts across various levels of granularity.

```
US <- read_csv('us.csv', show_col_types = F)
states <- read_csv('us-states.csv', show_col_types = F)
counties <- read_csv('us-counties-all.csv', show_col_types = F)
```

The national and state level data files had no missing values while the county level data file had 35,156 missing values for the county `fips` variable, which is a code for geographical regions used by the Federal Communications Commission, and 82,097 missing values within the `deaths` variable.

```
# Missing values
colSums(is.na(US))
```

```
##   date  cases deaths
##      0      0      0
```

```
colSums(is.na(states))
```

```
##   date  state   fips  cases deaths
##      0      0      0      0      0
```

```
colSums(is.na(counties))
```

```
##   date county  state   fips  cases deaths
##      0      0      0  35156      0  82097
```

```
# Missing values in deaths
na_deaths <- counties %>% filter(is.na(deaths))
head(na_deaths)
```

```
## # A tibble: 6 x 6
##   date       county       state       fips  cases deaths
##   <date>     <chr>        <chr>       <chr> <dbl>  <dbl>
## 1 2020-05-05 Adjuntas     Puerto Rico 72001     3     NA
## 2 2020-05-05 Aguada       Puerto Rico 72003     7     NA
## 3 2020-05-05 Aguadilla    Puerto Rico 72005    11     NA
## 4 2020-05-05 Aguas Buenas Puerto Rico 72007    22     NA
## 5 2020-05-05 Aibonito     Puerto Rico 72009    13     NA
## 6 2020-05-05 Anasco       Puerto Rico 72011     5     NA
```

```
unique(na_deaths$state)
```

```
## [1] "Puerto Rico"
```

To check the `county` variable values for missing values potentially marked by values such as “Unknown”, I compared the `county` values with the list of county names contained within the `USAboundaries` package. Besides “Unknown” values there were “Pending County Assignment” values which were interestingly only in Texas.

```
# Extract list of county names using USAboundaries package
county_names <- us_counties()
county_names <- county_names$name

unique(counties$county[!(counties$county %in% county_names)])
```

```
##  [1] "New York City"                           "Unknown"
##  [3] "Virginia Beach city"                 "Alexandria city"
##  [5] "Harrisonburg city"                   "Baltimore city"
##  [7] "Fairbanks North Star Borough"        "Ketchikan Gateway Borough"
##  [9] "Charlottesville city"                "Williamsburg city"
## [11] "Kenai Peninsula Borough"             "St. Louis city"
## [13] "Richmond city"                       "Kansas City"
## [15] "Newport News city"                   "Norfolk city"
## [17] "Portsmouth city"                     "Suffolk city"
## [19] "Juneau City and Borough"             "Matanuska-Susitna Borough"
## [21] "Danville city"                       "Chesapeake city"
## [23] "Fredericksburg city"                 "Manassas city"
## [25] "Hampton city"                        "Lynchburg city"
## [27] "Poquoson city"                       "Radford city"
## [29] "Bristol city"                        "Galax city"
## [31] "Roanoke city"                        "Hopewell city"
## [33] "Manassas Park city"                  "Winchester city"
## [35] "Petersburg city"                     "Franklin city"
## [37] "Waynesboro city"                     "Yukon-Koyukuk Census Area"
## [39] "Salem city"                          "Southeast Fairbanks Census Area"
## [41] "Buena Vista city"                    "Emporia city"
## [43] "Lexington city"                      "Staunton city"
## [45] "Petersburg Borough"                  "St. John"
## [47] "St. Thomas"                          "Bethel Census Area"
## [49] "Colonial Heights city"               "Fairfax city"
## [51] "Prince of Wales-Hyder Census Area"   "Falls Church city"
## [53] "Nome Census Area"                    "Kodiak Island Borough"
## [55] "Norton city"                         "Sitka City and Borough"
## [57] "Martinsville city"                   "Anasco"
## [59] "Bayamon"                             "Canovanas"
## [61] "Catano"                              "Comerio"
## [63] "Guanica"                             "Juana Diaz"
## [65] "Las Marias"                          "Loiza"
## [67] "Manati"                              "Mayaguez"
## [69] "Penuelas"                            "Rincon"
## [71] "San German"                          "San Sebastian"
## [73] "Valdez-Cordova Census Area"          "Covington city"
## [75] "Bristol Bay plus Lake and Peninsula" "Northwest Arctic Borough"
## [77] "North Slope Borough"                 "Dillingham Census Area"
## [79] "Aleutians West Census Area"          "Wrangell City and Borough"
## [81] "Aleutians East Borough"              "Haines Borough"
## [83] "Denali Borough"                      "Joplin"
## [85] "Kusilvak Census Area"                "Saipan"
## [87] "Tinian"                              "Yakutat plus Hoonah-Angoon"
## [89] "Skagway Municipality"                "Rota"
## [91] "Pending County Assignment"
```

```r
unknown_counties <- counties %>% filter(county  == 'Unknown')
pending_county <- counties %>% filter(county == 'Pending County Assignment')

# Dataframe of rows with "Unknown" county value
head(unknown_counties)
```

```
## # A tibble: 6 x 6
##   date       county  state        fips  cases deaths
##   <date>     <chr>   <chr>        <chr> <dbl>  <dbl>
## 1 2020-03-01 Unknown Rhode Island <NA>      2      0
## 2 2020-03-02 Unknown Rhode Island <NA>      2      0
## 3 2020-03-03 Unknown Rhode Island <NA>      2      0
## 4 2020-03-04 Unknown Rhode Island <NA>      2      0
## 5 2020-03-05 Unknown Rhode Island <NA>      2      0
## 6 2020-03-06 Unknown Rhode Island <NA>      3      0
```

```r
dim(unknown_counties)
```

```
## [1] 31937     6
```

```r
# Dataframe of rows with "Pending County Assignment" county value
head(pending_county)
```

```
## # A tibble: 6 x 6
##   date       county                    state fips  cases deaths
##   <date>     <chr>                     <chr> <chr> <dbl>  <dbl>
## 1 2022-10-24 Pending County Assignment Texas 48999    65      0
## 2 2022-10-25 Pending County Assignment Texas 48999    65      0
## 3 2022-10-26 Pending County Assignment Texas 48999    65      0
## 4 2022-10-27 Pending County Assignment Texas 48999    65      0
## 5 2022-10-28 Pending County Assignment Texas 48999    65      0
## 6 2022-10-29 Pending County Assignment Texas 48999    65      0
```

```r
dim(pending_county)
```

```
## [1] 90  6
```

Additionally, all counties in Puerto Rico lacked death data as deaths were only tracked on the state-level within this dataset. All of the missing values in `deaths` were accounted for by this fact.

```r
PR_counties <- counties %>% filter(state == 'Puerto Rico')
head(PR_counties)
```

```
## # A tibble: 6 x 6
##   date       county  state       fips  cases deaths
##   <date>     <chr>   <chr>       <chr> <dbl>  <dbl>
## 1 2020-03-13 Unknown Puerto Rico <NA>      3      0
## 2 2020-03-14 Unknown Puerto Rico <NA>      4      0
## 3 2020-03-15 Unknown Puerto Rico <NA>      5      0
## 4 2020-03-16 Unknown Puerto Rico <NA>      5      0
## 5 2020-03-17 Unknown Puerto Rico <NA>      5      0
## 6 2020-03-18 Unknown Puerto Rico <NA>      5      0
```

```r
dim(PR_counties)
```

```
## [1] 83203     6
```

```r
# Puerto Rico county data
dim(intersect(PR_counties, unknown_counties))
```

```
## [1] 1106    6
```

```r
colSums(is.na(intersect(PR_counties, unknown_counties)))
```

```
##   date county  state   fips  cases deaths
##      0      0      0   1106      0      0
```

```
dim(setdiff(PR_counties, unknown_counties))
```

```
## [1] 82097     6
```

```
colSums(is.na(setdiff(PR_counties, unknown_counties)))
```

```
##   date county  state   fips  cases deaths 
##      0      0      0      0      0  82097
```

Ultimately to tidy the data I decided to exclude all "Unknown" and "Pending County Assignment" observations as these were ambiguous to the actual location of residence of those individuals. By excluding those rows, this also eliminated all missing values from the data. I also split each dataset into smaller tibbles for each geographic scale and metric (i.e. death vs case count).

```
county_cases <- counties %>%
  select(date, county, state, cases) %>%
  filter(county != 'Unknown' & county != 'Pending County Assignment')
# colSums(is.na(county_cases))

county_deaths <- counties %>%
  select(date, county, state, deaths) %>%
  drop_na() %>%
  filter(county != 'Unknown' & county != 'Pending County Assignment')
# colSums(is.na(county_deaths))

state_cases <- states %>% select(date, state, cases)

state_deaths <- states %>% select(date, state, deaths)

US_cases <- US %>% select(-deaths)

US_deaths <- US %>% select(-cases)
```

```
head(county_cases)
```

```
## # A tibble: 6 x 4
##   date       county    state      cases
##   <date>     <chr>     <chr>      <dbl>
## 1 2020-01-21 Snohomish Washington     1
## 2 2020-01-22 Snohomish Washington     1
## 3 2020-01-23 Snohomish Washington     1
## 4 2020-01-24 Cook      Illinois       1
## 5 2020-01-24 Snohomish Washington     1
## 6 2020-01-25 Orange    California     1
```

```
dim(county_cases)
```

```
## [1] 3493134       4
```

```
summary(county_cases$date)
```

```
##         Min.      1st Qu.       Median         Mean      3rd Qu.         Max. 
## "2020-01-21" "2020-12-30" "2021-09-27" "2021-09-26" "2022-06-25" "2023-03-23"
```

```
length(unique(county_cases$county))
```

```
## [1] 1930
```

```r
head(county_deaths)
```

```
## # A tibble: 6 x 4
##   date       county    state      deaths
##   <date>     <chr>     <chr>       <dbl>
## 1 2020-01-21 Snohomish Washington      0
## 2 2020-01-22 Snohomish Washington      0
## 3 2020-01-23 Snohomish Washington      0
## 4 2020-01-24 Cook      Illinois        0
## 5 2020-01-24 Snohomish Washington      0
## 6 2020-01-25 Orange    California      0
```

```r
dim(county_deaths)
```

```
## [1] 3411037       4
```

```r
summary(county_deaths$date)
```

```
##         Min.      1st Qu.       Median         Mean      3rd Qu.         Max. 
## "2020-01-21" "2020-12-30" "2021-09-27" "2021-09-26" "2022-06-25" "2023-03-23"
```

```r
length(unique(county_deaths$county))
```

```
## [1] 1854
```

```r
head(state_cases)
```

```
## # A tibble: 6 x 3
##   date       state      cases
##   <date>     <chr>      <dbl>
## 1 2020-01-21 Washington     1
## 2 2020-01-22 Washington     1
## 3 2020-01-23 Washington     1
## 4 2020-01-24 Illinois       1
## 5 2020-01-24 Washington     1
## 6 2020-01-25 California     1
```

```r
dim(state_cases)
```

```
## [1] 61942     3
```

```r
head(state_deaths)
```

```
## # A tibble: 6 x 3
##   date       state      deaths
##   <date>     <chr>       <dbl>
## 1 2020-01-21 Washington      0
## 2 2020-01-22 Washington      0
## 3 2020-01-23 Washington      0
## 4 2020-01-24 Illinois        0
## 5 2020-01-24 Washington      0
## 6 2020-01-25 California      0
```

```r
dim(state_deaths)
```

```
## [1] 61942     3
```

```
head(US_cases)
```

```
## # A tibble: 6 x 2
##   date       cases
##   <date>     <dbl>
## 1 2020-01-21     1
## 2 2020-01-22     1
## 3 2020-01-23     1
## 4 2020-01-24     2
## 5 2020-01-25     3
## 6 2020-01-26     5
```

```
dim(US_cases)
```

```
## [1] 1158    2
```

```
head(US_deaths)
```

```
## # A tibble: 6 x 2
##   date       deaths
##   <date>      <dbl>
## 1 2020-01-21      0
## 2 2020-01-22      0
## 3 2020-01-23      0
## 4 2020-01-24      0
## 5 2020-01-25      0
## 6 2020-01-26      0
```

```
dim(US_deaths)
```

```
## [1] 1158    2
```

Dataframe	**Number of Rows**
County Cases	3,493,134
County Deaths	3,411,037
State Cases	61,942
State Deaths	61,942
US Cases	1,158
US Deaths	1,158

Table 1: Sizes of Cleaned Dataframes

Exploratory Data Analysis

For the county and state levels, I summarized the number of cases and deaths using the mean case and death counts due to the lack of easily-accessible demographic data to calculate prevalence/incidence metrics.

```
county_cases_daily <- county_cases %>%
  group_by(date) %>%
  summarise(mean_cases = mean(cases)) %>%
  ungroup()

county_deaths_daily <- county_deaths %>%
  group_by(date) %>%
  summarise(mean_deaths = mean(deaths)) %>%
```

```r
  ungroup() %>%
  mutate(daily_change = mean_deaths - lag(mean_deaths))
```

```r
state_cases_daily <- state_cases %>%
  group_by(date) %>%
  summarise(mean_cases = mean(cases)) %>%
  ungroup()

state_deaths_daily <- state_deaths %>%
  group_by(date) %>%
  summarise(mean_deaths = mean(deaths)) %>%
  ungroup()
```

```r
US_cases_daily <- US_cases %>%
  mutate(daily_change = cases - lag(cases))
US_deaths_daily <- US_deaths %>%
  mutate(daily_change = deaths - lag(deaths))
```

```r
county_cases_weekly <- county_cases %>%
  mutate(week = floor_date(date, unit = "week")) %>%
  group_by(week) %>%
  summarise(mean_cases = mean(cases, na.rm = TRUE)) %>%
  ungroup()

county_deaths_weekly <- county_deaths %>%
  mutate(week = floor_date(date, unit = "week")) %>%
  group_by(week) %>%
  summarise(mean_deaths = mean(deaths, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(weekly_change = mean_deaths - lag(mean_deaths))
```

```r
state_cases_weekly <- state_cases %>%
  mutate(week = floor_date(date, unit = "week")) %>%
  group_by(week) %>%
  summarise(mean_cases = mean(cases, na.rm = TRUE)) %>%
  ungroup()

state_deaths_weekly <- state_deaths %>%
  mutate(week = floor_date(date, unit = "week")) %>%
  group_by(week) %>%
  summarise(mean_deaths = mean(deaths, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(weekly_change = mean_deaths - lag(mean_deaths))
```

```r
US_cases_weekly <- US_cases %>%
  mutate(week = floor_date(date, unit = "week")) %>%
  group_by(week) %>%
  summarise(cases = max(cases, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(weekly_change = cases - lag(cases))

US_deaths_weekly <- US_deaths %>%
  mutate(week = floor_date(date, unit = "week")) %>%
  group_by(week) %>%
  summarise(deaths = max(deaths, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(weekly_change = deaths - lag(deaths))
```

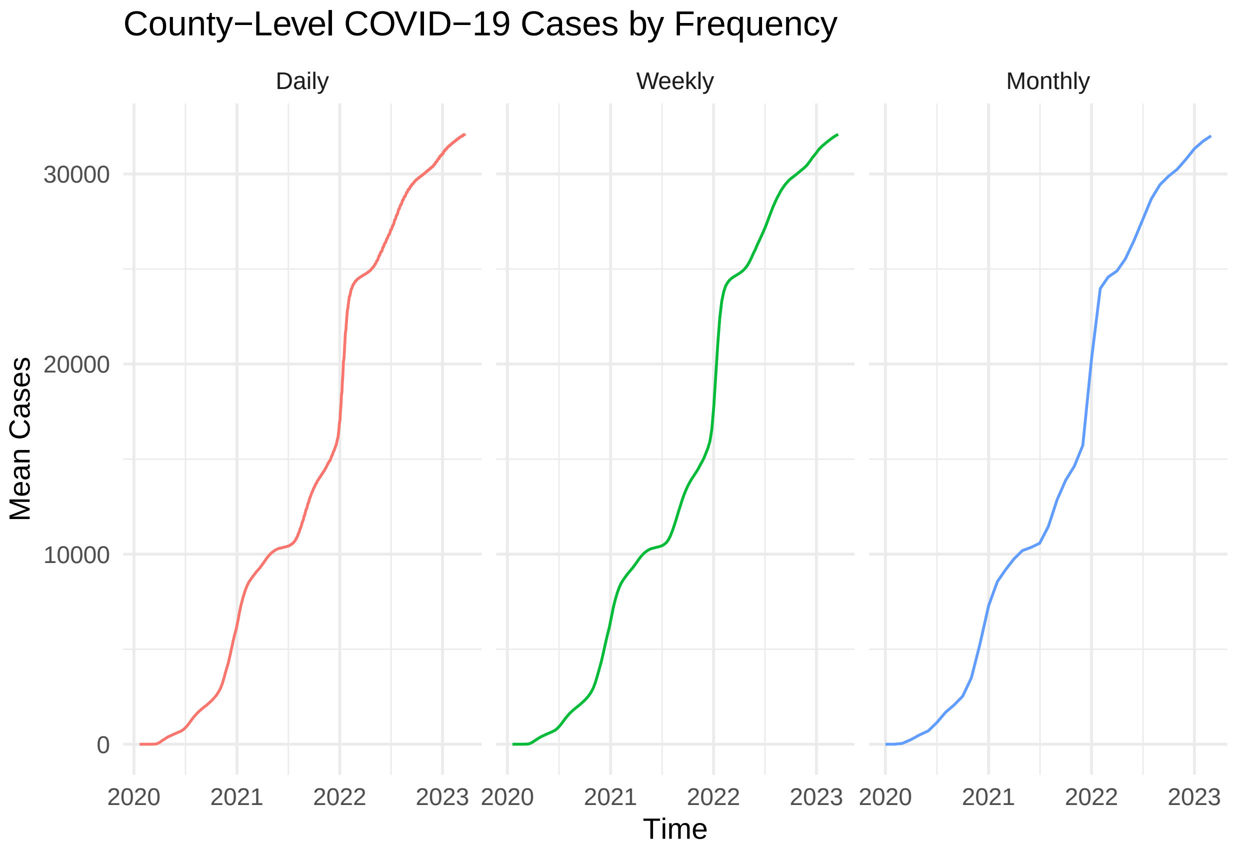
```
county_cases_monthly <- county_cases %>%
  mutate(month = floor_date(date, unit = "month")) %>%
  group_by(month) %>%
  summarise(mean_cases = mean(cases, na.rm = TRUE)) %>%
  ungroup()

county_deaths_monthly <- county_deaths %>%
  mutate(month = floor_date(date, unit = "month")) %>%
  group_by(month) %>%
  summarise(mean_deaths = mean(deaths, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(monthly_change = mean_deaths - lag(mean_deaths))
```

```
state_cases_monthly <- state_cases %>%
  mutate(month = floor_date(date, unit = "month")) %>%
  group_by(month) %>%
  summarise(mean_cases = mean(cases, na.rm = TRUE)) %>%
  ungroup()

state_deaths_monthly <- state_deaths %>%
  mutate(month = floor_date(date, unit = "month")) %>%
  group_by(month) %>%
  summarise(mean_deaths = mean(deaths, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(monthly_change = mean_deaths - lag(mean_deaths))
```

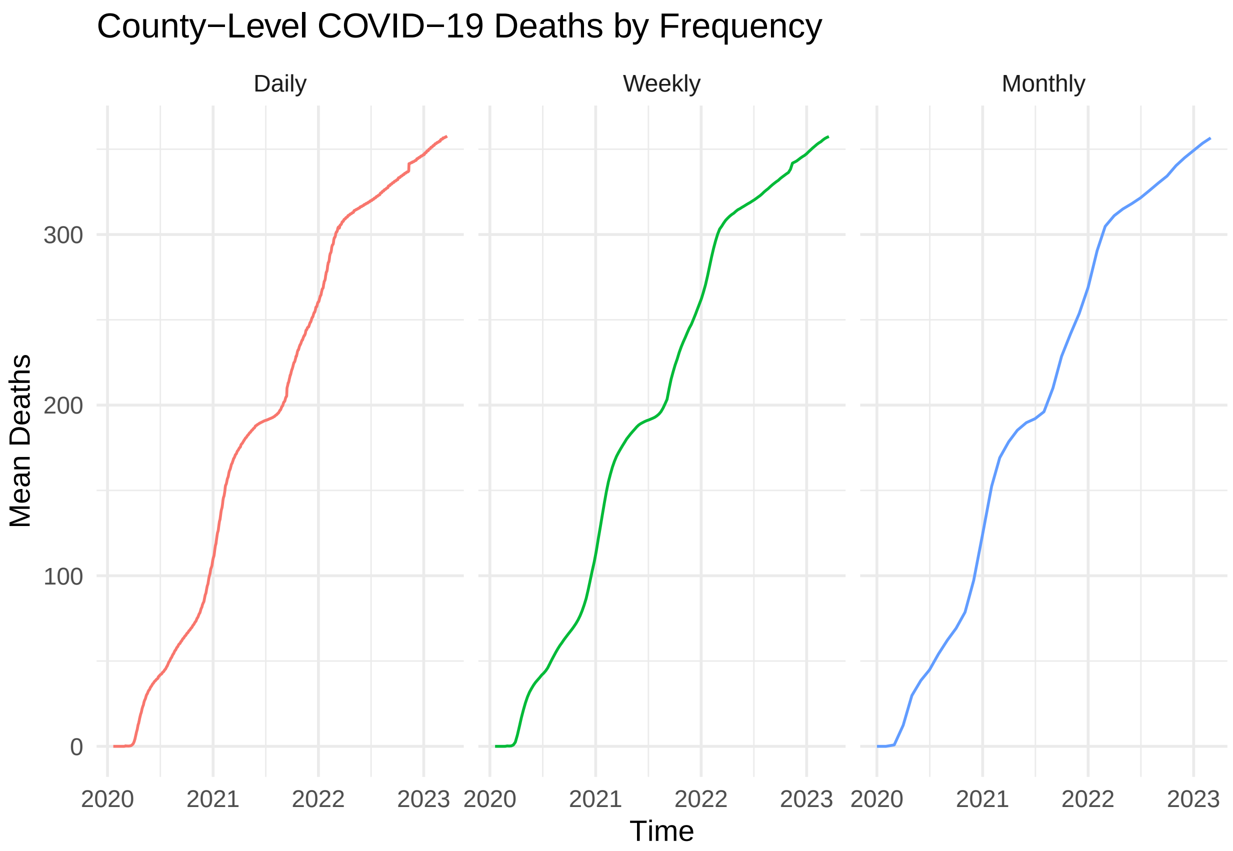
```
US_cases_monthly <- US_cases %>%
  mutate(month = floor_date(date, unit = "month")) %>%
  group_by(month) %>%
  summarise(cases = max(cases, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(monthly_change = cases - lag(cases))

US_deaths_monthly <- US_deaths %>%
  mutate(month = floor_date(date, unit = "month")) %>%
  group_by(month) %>%
  summarise(deaths = max(deaths, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(monthly_change = deaths - lag(deaths))
```

```
county_cases_daily$frequency <- "Daily"
county_cases_weekly$frequency <- "Weekly"
county_cases_monthly$frequency <- "Monthly"

county_cases_daily <- county_cases_daily %>% rename(time = date)
county_cases_weekly <- county_cases_weekly %>% rename(time = week)
county_cases_monthly <- county_cases_monthly %>% rename(time = month)
```

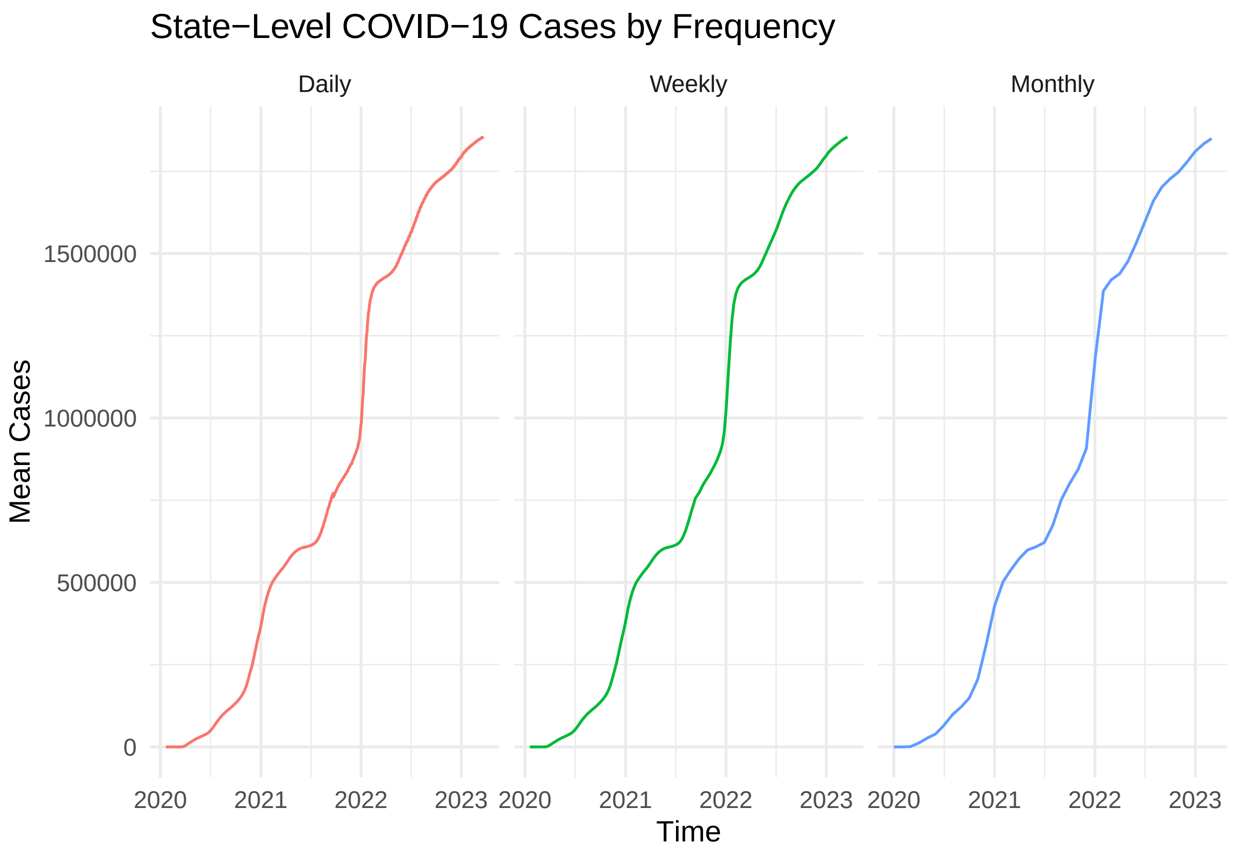
```
combined_county_cases <- bind_rows(county_cases_daily, county_cases_weekly,
↪  county_cases_monthly) %>%
  mutate(frequency = factor(frequency, levels = c("Daily", "Weekly", "Monthly")))

ggplot(data = combined_county_cases) +
  geom_line(mapping = aes(x = time, y = mean_cases, color = frequency), show.legend =
  ↪  FALSE) +
  facet_grid(. ~ frequency, scales = "free_x") +
  labs(title = "County-Level COVID-19 Cases by Frequency", x = "Time", y = "Mean Cases")
  ↪  +
  theme_minimal()
```



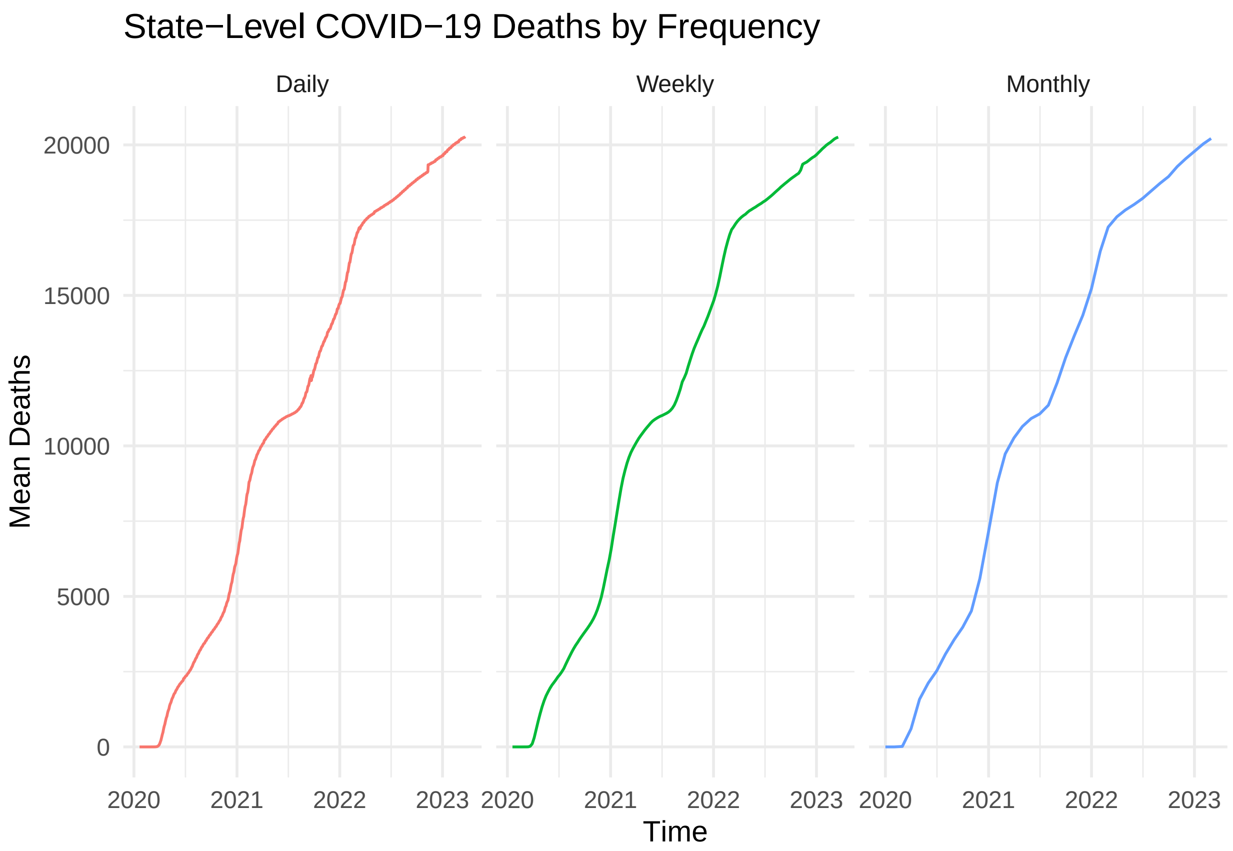
```
county_deaths_daily$frequency <- "Daily"
county_deaths_weekly$frequency <- "Weekly"
county_deaths_monthly$frequency <- "Monthly"

county_deaths_daily <- county_deaths_daily %>% rename(time = date)
county_deaths_weekly <- county_deaths_weekly %>% rename(time = week)
county_deaths_monthly <- county_deaths_monthly %>% rename(time = month)

combined_county_deaths <- bind_rows(county_deaths_daily, county_deaths_weekly,
↪  county_deaths_monthly) %>%
  mutate(frequency = factor(frequency, levels = c("Daily", "Weekly", "Monthly")))

ggplot(data = combined_county_deaths) +
  geom_line(mapping = aes(x = time, y = mean_deaths, color = frequency), show.legend =
  ↪   FALSE) +
  facet_grid(. ~ frequency, scales = "free_x") +
  labs(title = "County-Level COVID-19 Deaths by Frequency", x = "Time", y = "Mean
  ↪   Deaths") +
  theme_minimal()
```



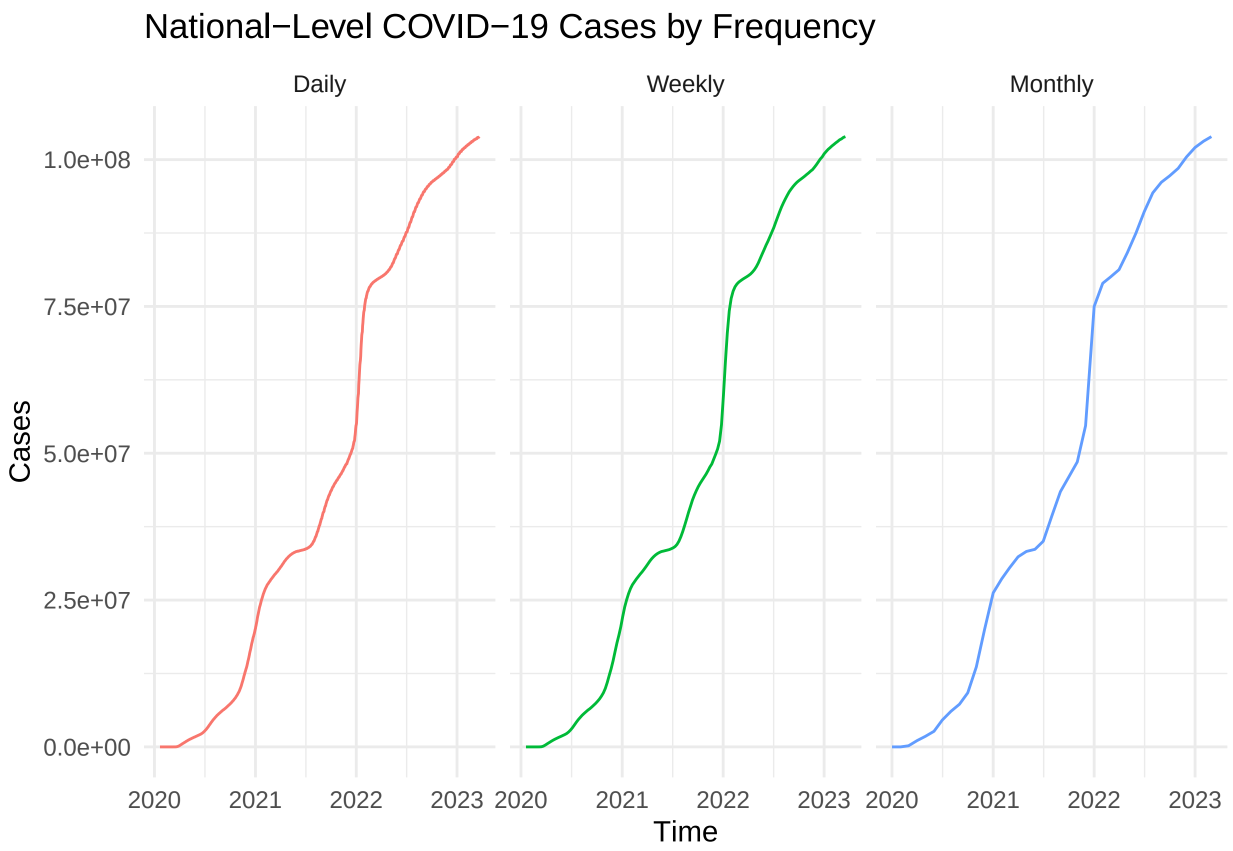
```
state_cases_daily$frequency <- "Daily"
state_cases_weekly$frequency <- "Weekly"
state_cases_monthly$frequency <- "Monthly"

state_cases_daily <- state_cases_daily %>% rename(time = date)
state_cases_weekly <- state_cases_weekly %>% rename(time = week)
state_cases_monthly <- state_cases_monthly %>% rename(time = month)

combined_state_cases <- bind_rows(state_cases_daily, state_cases_weekly,
↪   state_cases_monthly) %>%
  mutate(frequency = factor(frequency, levels = c("Daily", "Weekly", "Monthly")))

ggplot(data = combined_state_cases) +
  geom_line(mapping = aes(x = time, y = mean_cases, color = frequency), show.legend =
  ↪   FALSE) +
  facet_grid(. ~ frequency, scales = "free_x") +
  labs(title = "State-Level COVID-19 Cases by Frequency", x = "Time", y = "Mean Cases") +
  theme_minimal()
```



```
state_deaths_daily$frequency <- "Daily"
state_deaths_weekly$frequency <- "Weekly"
state_deaths_monthly$frequency <- "Monthly"

state_deaths_daily <- state_deaths_daily %>% rename(time = date)
state_deaths_weekly <- state_deaths_weekly %>% rename(time = week)
state_deaths_monthly <- state_deaths_monthly %>% rename(time = month)

combined_state_deaths <- bind_rows(state_deaths_daily, state_deaths_weekly,
↪  state_deaths_monthly) %>%
  mutate(frequency = factor(frequency, levels = c("Daily", "Weekly", "Monthly")))

ggplot(data = combined_state_deaths) +
  geom_line(mapping = aes(x = time, y = mean_deaths, color = frequency), show.legend =
  ↪  FALSE) +
  facet_grid(. ~ frequency, scales = "free_x") +
  labs(title = "State-Level COVID-19 Deaths by Frequency", x = "Time", y = "Mean Deaths")
  ↪  +
  theme_minimal()
```

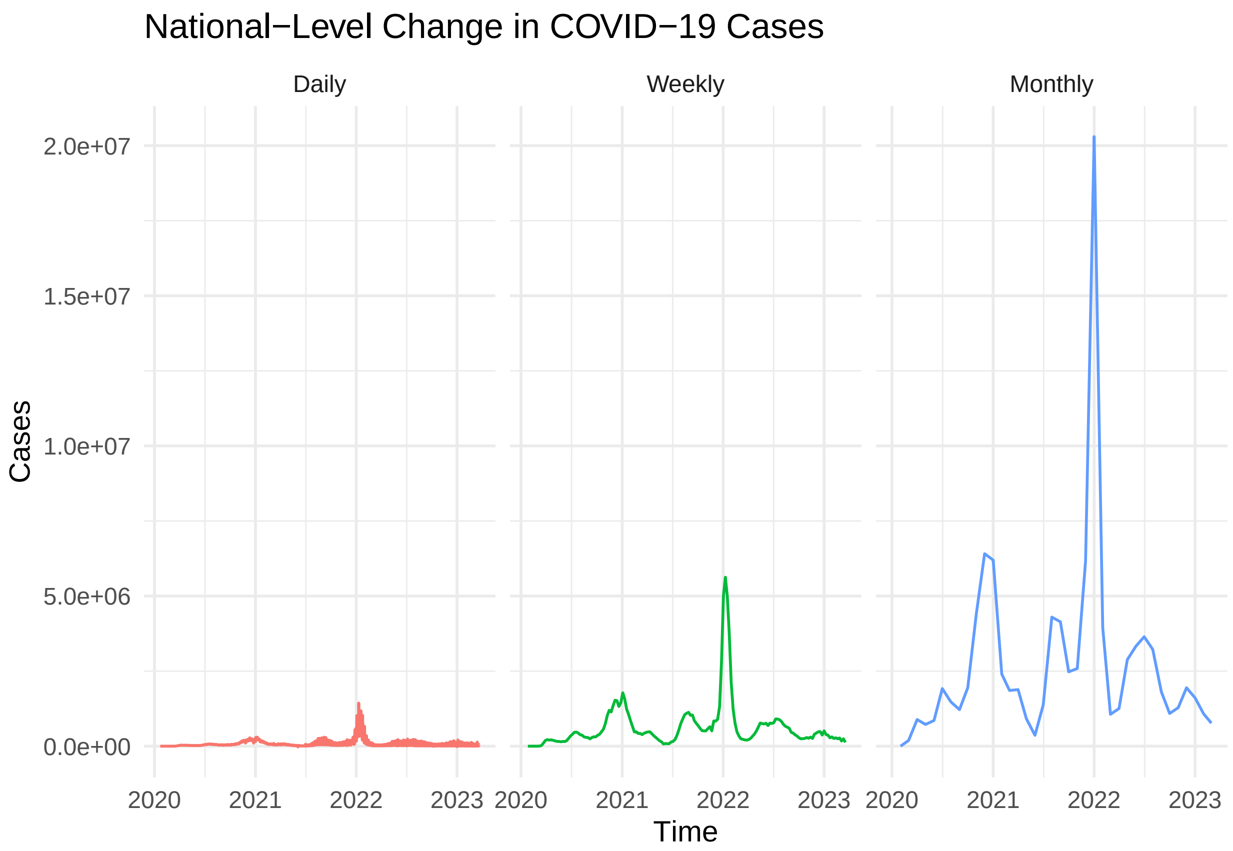


```
US_cases_daily$frequency <- "Daily"
US_cases_weekly$frequency <- "Weekly"
US_cases_monthly$frequency <- "Monthly"

US_cases_daily <- US_cases_daily %>% rename(time = date, change = daily_change)
US_cases_weekly <- US_cases_weekly %>% rename(time = week, change = weekly_change)
US_cases_monthly <- US_cases_monthly %>% rename(time = month, change = monthly_change)

combined_US_cases <- bind_rows(US_cases_daily, US_cases_weekly, US_cases_monthly) %>%
  mutate(frequency = factor(frequency, levels = c("Daily", "Weekly", "Monthly")))

ggplot(data = combined_US_cases) +
  geom_line(mapping = aes(x = time, y = cases, color = frequency), show.legend = FALSE) +
  facet_grid(. ~ frequency, scales = "free_x") +
  labs(title = "National-Level COVID-19 Cases by Frequency", x = "Time", y = "Cases") +
  theme_minimal()
```

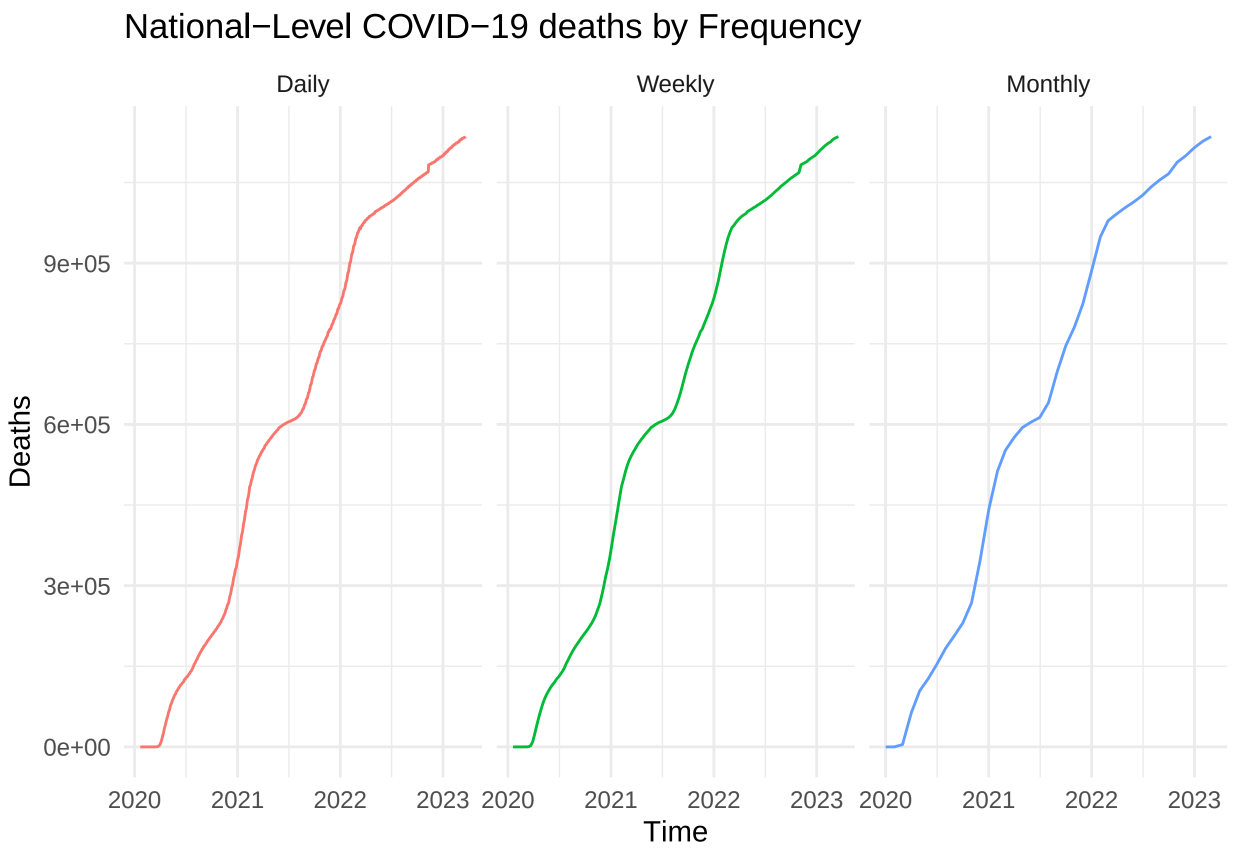


```r
ggplot(data = combined_US_cases) +
  geom_line(mapping = aes(x = time, y = change, color = frequency), show.legend = FALSE)
  ↪  +
  facet_grid(. ~ frequency, scales = "free_x") +
  labs(title = "National-Level Change in COVID-19 Cases", x = "Time", y = "Cases") +
  theme_minimal()
```

```
## Warning: Removed 3 rows containing missing values or values outside the scale range
## (`geom_line()`).
```

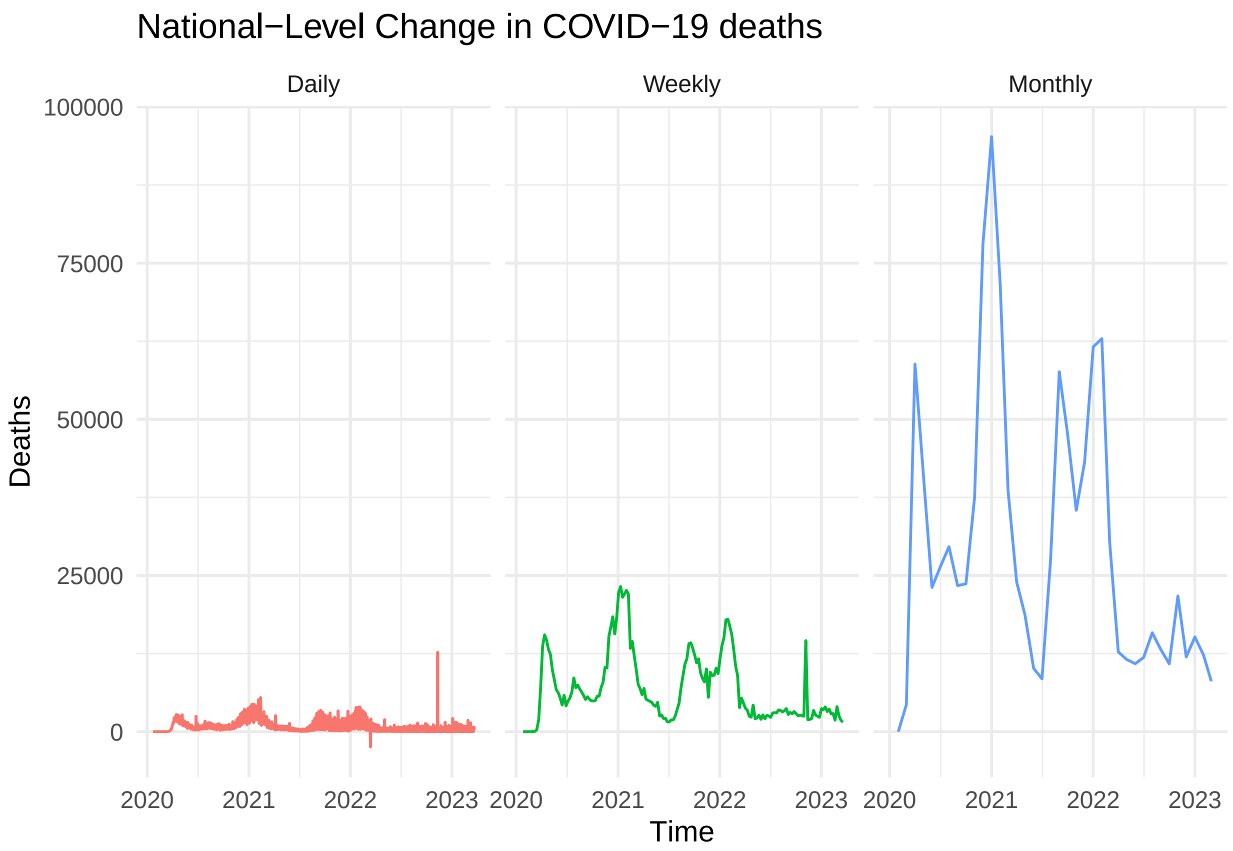


```
US_deaths_daily$frequency <- "Daily"
US_deaths_weekly$frequency <- "Weekly"
US_deaths_monthly$frequency <- "Monthly"

US_deaths_daily <- US_deaths_daily %>% rename(time = date, change = daily_change)
US_deaths_weekly <- US_deaths_weekly %>% rename(time = week, change = weekly_change)
US_deaths_monthly <- US_deaths_monthly %>% rename(time = month, change = monthly_change)

combined_US_deaths <- bind_rows(US_deaths_daily, US_deaths_weekly, US_deaths_monthly) %>%
  mutate(frequency = factor(frequency, levels = c("Daily", "Weekly", "Monthly")))

ggplot(data = combined_US_deaths) +
  geom_line(mapping = aes(x = time, y = deaths, color = frequency), show.legend = FALSE)
  ↪  +
  facet_grid(. ~ frequency, scales = "free_x") +
  labs(title = "National-Level COVID-19 deaths by Frequency", x = "Time", y = "Deaths") +
  theme_minimal()
```



```
ggplot(data = combined_US_deaths) +
  geom_line(mapping = aes(x = time, y = change, color = frequency), show.legend = FALSE)
  ↪  +
  facet_grid(. ~ frequency, scales = "free_x") +
  labs(title = "National-Level Change in COVID-19 deaths", x = "Time", y = "Deaths") +
  theme_minimal()
```

```
## Warning: Removed 3 rows containing missing values or values outside the scale range
## (`geom_line()`).
```



Summary

This analysis explored COVID-19 case and death rates across different temporal resolutions (daily, weekly, and monthly) and geographical levels (county, state, and national). Key findings from the exploratory data analysis revealed distinct patterns of fluctuation in both cases and deaths depending on the level of temporal aggregation. Daily data exhibited significant variability, which was smoothed out in weekly and monthly trends, offering clearer insights into long-term patterns. Geographically, county-level data provided granular insights into local outbreaks, while state and national aggregations highlighted broader trends and peaks in the pandemic trajectory.

The methodology involved cleaning and restructuring datasets to address missing values and inconsistencies, particularly in county-level data. Visualizations such as line plots of mean cases and deaths for each temporal resolution allowed for effective comparison. These insights suggest that the choice of timeframe and geographical level significantly impacts the interpretation of trends, emphasizing the importance of aligning analytical approaches with specific research or policy goals.

The analysis provides actionable insights for public health officials, highlighting the utility of aggregated data for strategic decision-making and the importance of detailed data for localized interventions. However, limitations include the lack of additional demographic variables and the potential impact of unrecorded data. Future work could incorporate more comprehensive datasets to examine disparities and refine predictive modeling for public health planning.