Study of COVID-19 Impact in Scandinavia

Intro

In the context of the COVID-19 pandemic, the Nordic countries provide a unique region for analysis due to their different approaches to combating the virus. My research pays particular attention to the contrast between the strategies of **Sweden** and its neighbors **Finland**, **Denmark**, and **Norway**.

Sweden has chosen a path of minimal restrictions, relying on recommendations and voluntary compliance. Authorities have kept schools, restaurants and shops open, focusing on protecting vulnerable groups and maintaining health care. This approach has sparked much debate and discussion in the international community.

While Sweden took less stringent measures, Finland, Denmark and Norway imposed strict restrictions early in the pandemic. These countries closed schools, universities and public spaces, imposed lockdowns and restrictions on gatherings. Borders were also closed to most travellers, which helped control the spread of the virus. Digital tools for contact tracing and quarantine enforcement played an important role in these countries.

Finland, for example, imposed a two-month lockdown at the start of the pandemic, closing schools, restaurants and banning travel to and from Helsinki and the surrounding area. The rapid rollout of a contact-tracing app and high levels of public trust in government measures also helped compliance.

Denmark has used a strategy of mass testing and contact tracing to identify and isolate cases early. Norway also implemented tough measures early on, including closing borders and introducing local restrictions depending on the infection rate in different regions.

Vaccination strategies also varied. While all countries were aggressive in their vaccination campaigns, the pace and methods varied. For example, Norway and Finland focused on protecting the elderly and vulnerable, while Denmark and Sweden also focused on vaccinating younger people and critical workers.

The aim of my research is to provide an overall analysis of the COVID-19 situation and mortality in the Nordic countries and to compare different approaches and their consequences. I want to find out how different pandemic strategies have affected the incidence and mortality rates and what can be learned from these differences for future epidemiological strategies.

Loading Data

```r
library(tidyverse)
library(dplyr)
library(lubridate)
library(ggplot2)
```

```r
url_in <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covi
cases <- read_csv(str_c(url_in, "time_series_covid19_confirmed_global.csv"))
deaths <- read_csv(str_c(url_in, "time_series_covid19_deaths_global.csv"))
```

Data preprocessing and cleaning

The following steps were performed for data preprocessing and cleaning: From the global data on confirmed COVID-19 cases and deaths, rows for the Nordic countries (**Finland**, **Sweden**, **Denmark**, **Norway**) were selected, unnecessary columns such as geo-coordinates were removed, as they are not required for the analysis. Data for the island territories of Denmark were also excluded to avoid bias in the analysis. Key columns were renamed and dates were converted to date format. The data was converted from wide format to long format, then the cases and deaths data were aggregated by country and date, and a column with the population of each country was added for further analysis. After that, the data was grouped by country, year and month, cumulative and new cases and deaths were calculated, and a new dataframe was created for further analysis. In the end, two dataframes were obtained: one with raw data and one with aggregated data by month.

```r
population <- data.frame(
  Country = c("Finland", "Sweden", "Denmark", "Norway"),
  Population = c(5530719, 10379295, 5831404, 5421241)
)

cases <- cases %>%
  rename(Country = `Country/Region`) %>%
  filter(Country %in% population$Country & is.na(`Province/State`)) %>%
  select(-c(`Province/State`, Lat, Long)) %>%
  pivot_longer(cols = -Country, names_to = "Date", values_to = "Cases") %>%
  mutate(Date = mdy(Date))

deaths <- deaths %>%
  rename(Country = `Country/Region`) %>%
  filter(Country %in% population$Country & is.na(`Province/State`)) %>%
  select(-c(`Province/State`, Lat, Long)) %>%
    pivot_longer(cols = -Country, names_to = "Date", values_to = "Deaths") %>%
    mutate(Date = mdy(Date))

data <- cases %>%
  full_join(deaths, by=join_by("Country", "Date")) %>%
  mutate(Year = year(Date), Month = month(Date)) %>%
  full_join(population, by="Country")

summary(data)
```

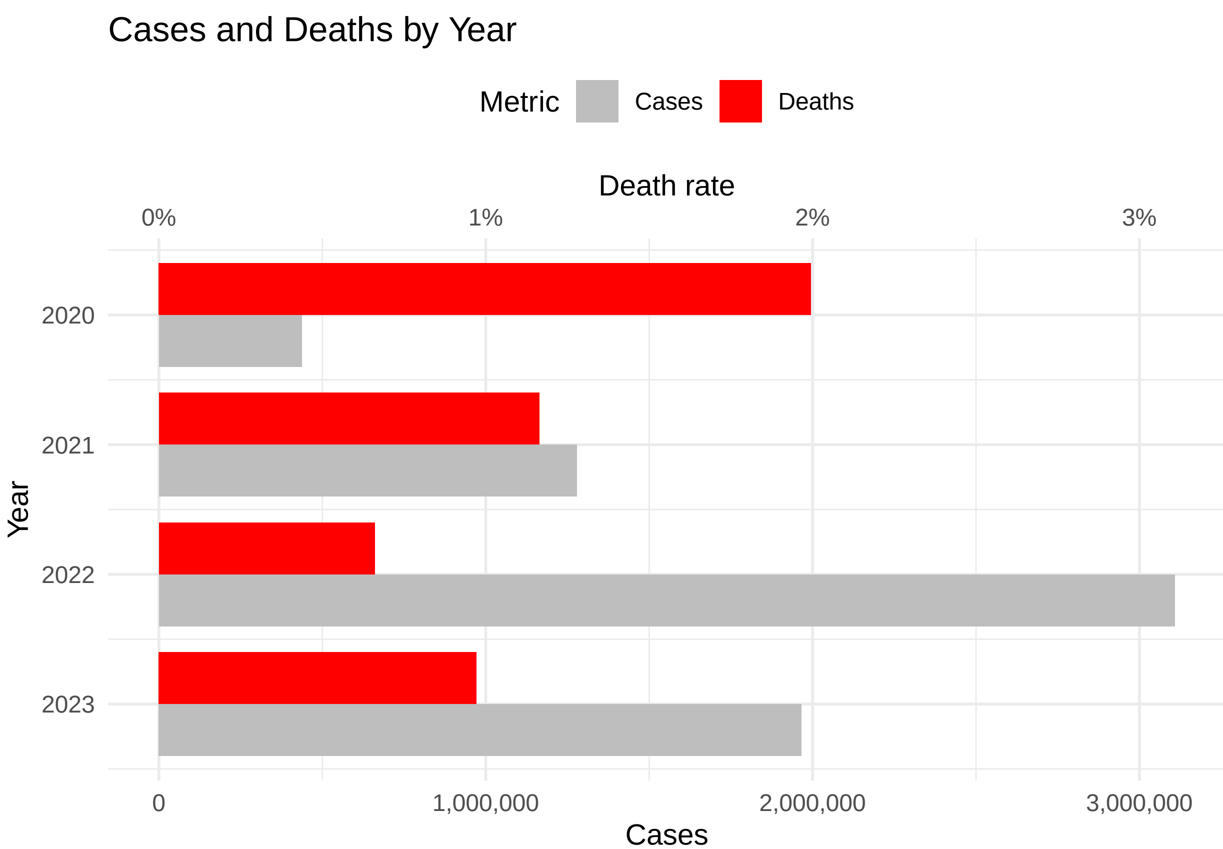
```
##    Country               Date                 Cases             Deaths
##  Length:4572        Min.   :2020-01-22   Min.   :      0   Min.   :    0
##  Class :character   1st Qu.:2020-11-02   1st Qu.:  46982   1st Qu.:  673
##  Mode  :character   Median :2021-08-15   Median : 316438   Median : 2616
##                     Mean   :2021-08-15   Mean   : 894303   Mean   : 5166
##                     3rd Qu.:2022-05-28   3rd Qu.:1446439   3rd Qu.: 6893
##                     Max.   :2023-03-09   Max.   :3404407   Max.   :23777
##       Year          Month          Population
##  Min.   :2020   Min.   : 1.000   Min.   : 5421241
##  1st Qu.:2020   1st Qu.: 3.000   1st Qu.: 5503350
##  Median :2021   Median : 6.000   Median : 5681062
##  Mean   :2021   Mean   : 6.335   Mean   : 6790665
##  3rd Qu.:2022   3rd Qu.: 9.000   3rd Qu.: 6968377
##  Max.   :2023   Max.   :12.000   Max.   :10379295
```

```r
data_grouped <- data %>%
  group_by(Country, Year, Month) %>%
  summarise(
    CasesCum = max(Cases),
    DeathsCum = max(Deaths),
    CasesNew = max(Cases) - min(Cases),
    DeathsNew = max(Deaths) - min(Deaths),
    Population = first(Population),
    .groups = "drop")

summary(data_grouped)
```

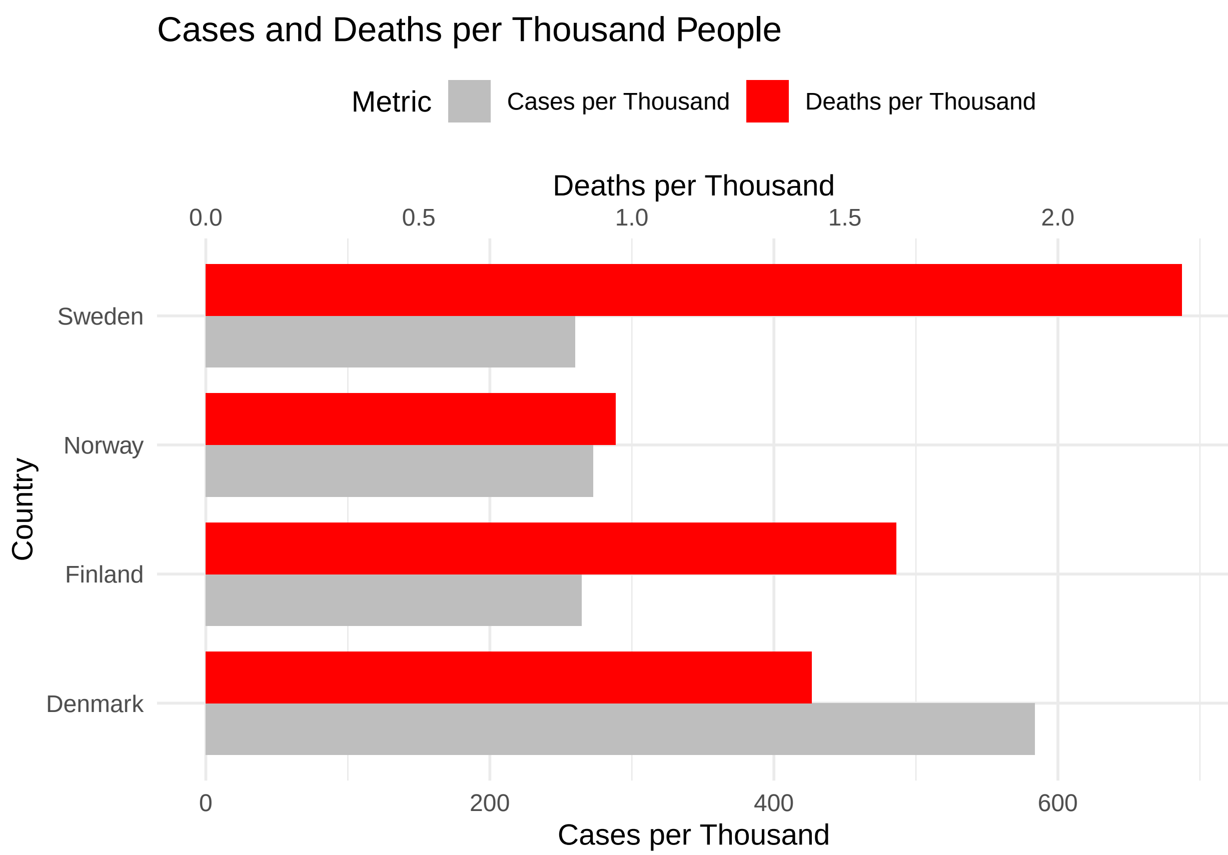
```
##    Country               Year          Month           CasesCum
##  Length:156         Min.   :2020   Min.   : 1.000   Min.   :      0
##  Class :character   1st Qu.:2020   1st Qu.: 3.000   1st Qu.:  48763
##  Mode  :character   Median :2021   Median : 6.000   Median : 352244
##                     Mean   :2021   Mean   : 6.154   Mean   : 932689
##                     3rd Qu.:2022   3rd Qu.: 9.000   3rd Qu.:1460638
##                     Max.   :2023   Max.   :12.000   Max.   :3404407
##    DeathsCum          CasesNew         DeathsNew          Population
##  Min.   :    0.0   Min.   :     0   Min.   :   0.0   Min.   : 5421241
##  1st Qu.:  695.5   1st Qu.:  4299   1st Qu.:  39.0   1st Qu.: 5503350
##  Median : 2685.0   Median : 16240   Median : 138.5   Median : 5681062
##  Mean   : 5336.0   Mean   : 55433   Mean   : 287.9   Mean   : 6790665
##  3rd Qu.: 7121.8   3rd Qu.: 33223   3rd Qu.: 375.2   3rd Qu.: 6968377
##  Max.   :23777.0   Max.   :976903   Max.   :2864.0   Max.   :10379295
```

Analysis through years and countries

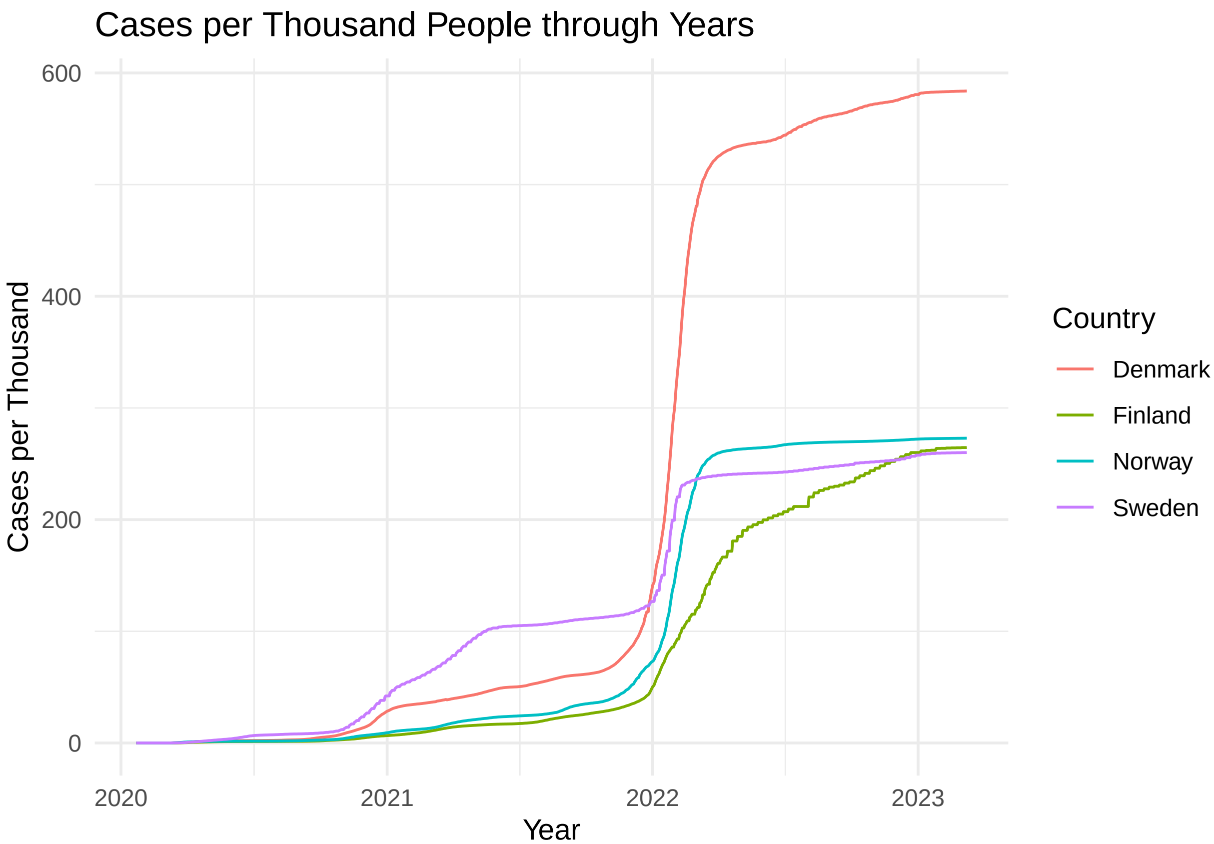
```r
data %>%
  group_by(Year) %>%
  summarise(
    Cases = max(Cases)-min(Cases),
    Deaths = (max(Deaths)-min(Deaths))/Cases,
    .groups = "drop") %>%
  mutate(Deaths = Deaths * 100000000) %>%
  pivot_longer(cols = c("Cases", "Deaths"),
               names_to = "metric", values_to = "value") %>%

ggplot(aes(x = Year, y = value, fill = metric)) +
  geom_bar(stat = "identity", position = "dodge", width = 0.8) +
  scale_y_continuous(
    name = "Cases",
    labels = scales::comma,
    sec.axis = sec_axis(~ . / 100000000, name = "Death rate",
                        labels = scales::percent)) +
  scale_fill_manual(values = c("Cases" = "gray", "Deaths" = "red")) +
  labs(title = "Cases and Deaths by Year",
       x = "Year",
       y = "Value",
       fill = "Metric") +
  theme_minimal() +
  scale_x_reverse() +
  coord_flip() +
  theme(legend.position = "top")
```



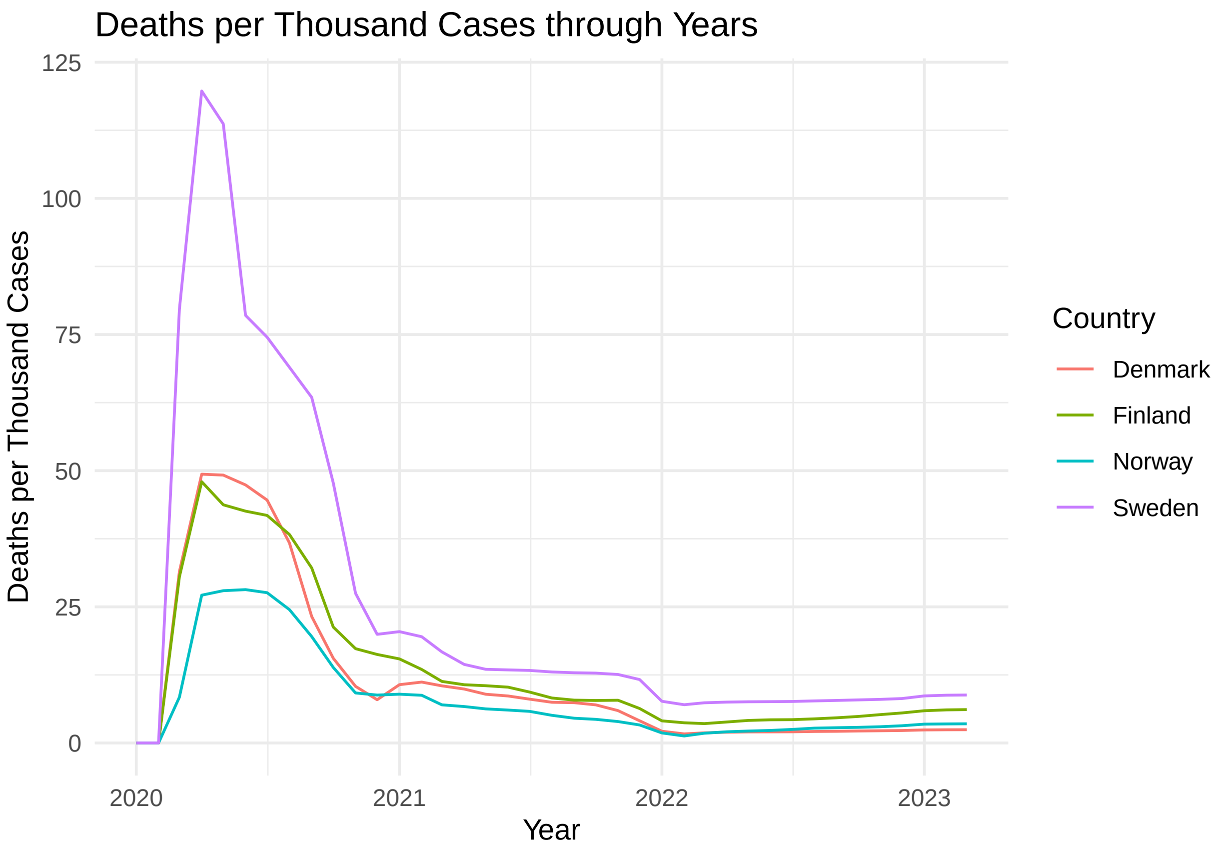

```r
data %>% group_by(Country, Population) %>%
  summarise(Cases = max(Cases), Deaths = max(Deaths), .groups = "drop") %>%
  mutate(`Cases per Thousand` = Cases * 1000 / Population,
         `Deaths per Thousand` = Deaths * 300 * 1000 / Population) %>%
  pivot_longer(cols = c("Cases per Thousand", "Deaths per Thousand"),
               names_to = "metric", values_to = "value") %>%

ggplot(aes(x = Country, y = value, fill = metric)) +
  geom_bar(stat = "identity", position = "dodge", width = 0.8) +
  scale_y_continuous(name = "Cases per Thousand",
    sec.axis = sec_axis(~ . / 300, name = "Deaths per Thousand")) +
  scale_fill_manual(values = c("Cases per Thousand" = "gray",
                               "Deaths per Thousand" = "red")) +
  labs(title = "Cases and Deaths per Thousand People",
       x = "Country",
       y = "Value",
       fill = "Metric") +
  theme_minimal() +
  coord_flip() +
  theme(legend.position = "top")
```




```r
data %>%
  mutate(Cases = Cases * 1000 / Population,
         Deaths = Deaths * 1000 / Population ) %>%
  ggplot() +
  geom_line(aes(x = Date, y = Cases, color=Country)) +
  labs(
    title = "Cases per Thousand People through Years",
       x = "Year",
       y = "Cases per Thousand",
       color = "Country") +
  theme_minimal()
```



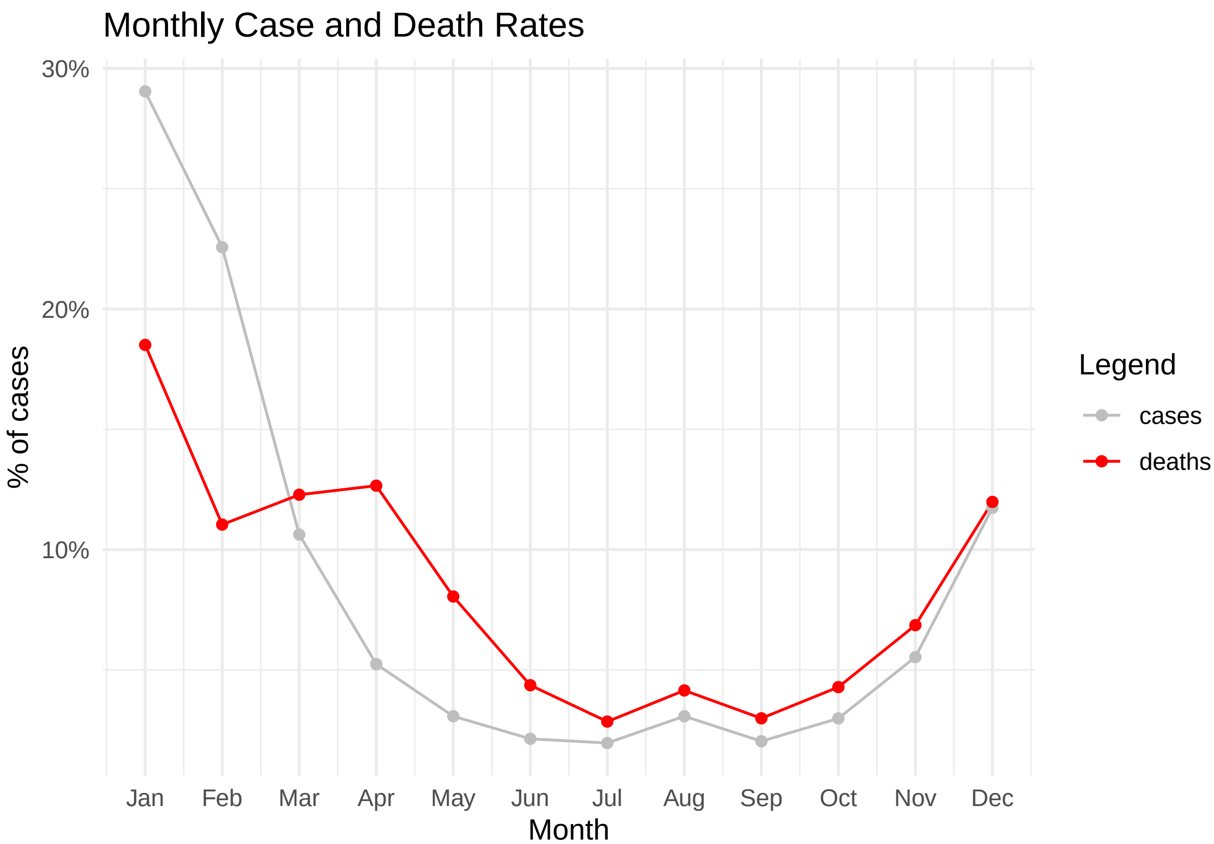

```r
data_grouped %>%
  mutate(DeathsCum = ifelse(CasesCum == 0, 0, DeathsCum * 1000 / CasesCum) ) %>%
  mutate(Date = as.Date(paste(Year, Month, "01", sep = "-"))) %>%

  ggplot(aes(x = Date, y = DeathsCum, color=Country)) +
  geom_line()+
  labs(
    title = "Deaths per Thousand Cases through Years",
       x = "Year",
       y = "Deaths per Thousand Cases",
       color = "Country") +
  theme_minimal()
```



Monthly Trends Analysis

```
data_grouped %>%
  group_by(Month) %>% summarise(Cases = sum(CasesNew),
                                Deaths = sum(DeathsNew), .groups = "drop") %>%
  mutate(Cases = Cases/sum(Cases), Deaths = Deaths/sum(Deaths)) %>%

  ggplot(aes(x = Month)) +
  geom_line(aes(y = Cases, color = "cases")) +
  geom_line(aes(y = Deaths, color = "deaths")) +
  geom_point(aes(y = Cases, color = "cases")) +
  geom_point(aes(y = Deaths, color = "deaths")) +
  scale_color_manual(values = c("cases" = "gray", "deaths" = "red")) +
  scale_y_continuous(labels = scales::percent) +
  labs(
    title = "Monthly Case and Death Rates",
       x = "Month",
```

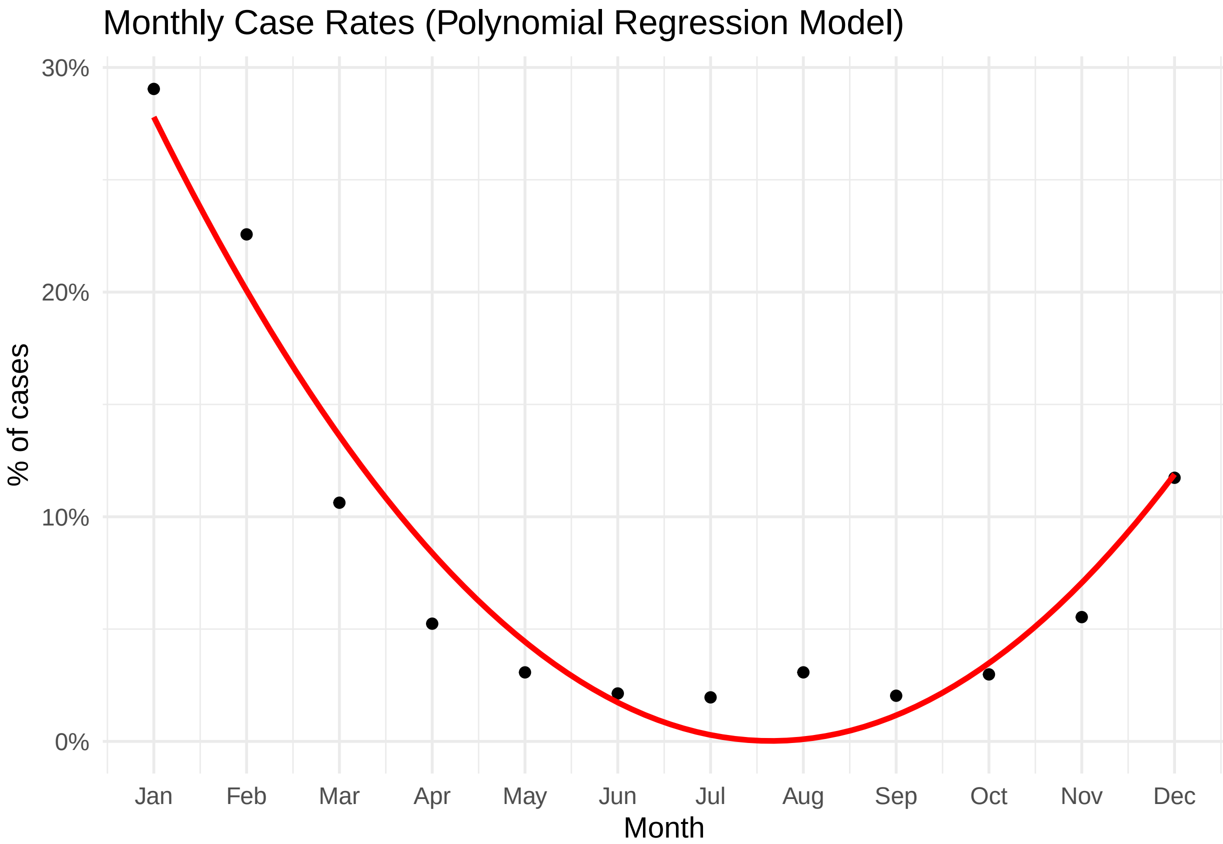
```
      y = "% of cases",
      color = "Legend") +
  theme_minimal() +
  scale_x_continuous(breaks = 1:12, labels = month.abb)
```



Polynomial Regression Model

```
data_montly <- data_grouped %>%
  group_by(Month) %>% summarise(Cases = sum(CasesNew), .groups = "drop") %>%
  mutate(Cases = Cases/sum(Cases))

  ggplot(data_montly, aes(x = Month, y = Cases)) +
  geom_point() +
  geom_smooth(method = "lm", formula = y ~ poly(x, 2), se = FALSE, color = "red") +
  scale_y_continuous(labels = scales::percent) +
  labs(
    title = "Monthly Case Rates (Polynomial Regression Model)",
```

```r
       x = "Month",
       y = "% of cases") +
  theme_minimal() +
  scale_x_continuous(breaks = 1:12, labels = month.abb)
```




```r
summary(lm(Cases ~ poly(Month, 2), data = data_montly))
```

```
##
## Call:
## lm(formula = Cases ~ poly(Month, 2), data = data_montly)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.031499 -0.014020  0.001219  0.013555  0.029776
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.083333   0.006357   13.11 3.61e-07 ***
## poly(Month, 2)1 -0.172871   0.022020   -7.85 2.57e-05 ***
## poly(Month, 2)2  0.229367   0.022020   10.42 2.55e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02202 on 9 degrees of freedom
## Multiple R-squared:  0.9498, Adjusted R-squared:  0.9386
```

```
## F-statistic: 85.06 on 2 and 9 DF,  p-value: 1.428e-06
```