Covid-19 Report

1. Introduciton

This paper analyzes Covid-19 data from the Johns Hopkins GitHub repository. This dataset details the number of Covid-19 cases and deaths recorded by various countries throughout the pandemic. Our focus is on examining how different countries managed the outbreak and exploring the correlation between Covid-19 cases and deaths when grouped by country.

Libraries needed:

```
library(tidyverse)
library(ggplot2)
library(dplyr)
```

Importing the data:

```
url_in <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_cov
file_names <- c("time_series_covid19_confirmed_global.csv","time_series_covid19_deaths_global.csv", "ti
urls <- str_c(url_in, file_names)
uid_lookup_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/

global_cases <- read_csv(urls[1])
global_deaths <- read_csv(urls[2])
uid = read_csv(uid_lookup_url) %>%
  select(-c(Lat, Long_, Combined_Key, code3, iso2, iso3, Admin2))
```

2. Tidying the Data

Here we are cleaning up the dataset.

```
global_cases <- global_cases %>%
  pivot_longer(cols = -c('Province/State',
                         'Country/Region', Lat, Long),
               names_to = "date", values_to = "cases") %>%
  select(-c(Lat, Long))

global_deaths <- global_deaths %>%
  pivot_longer(cols = -c('Province/State',
                         'Country/Region', Lat, Long),
               names_to = "date", values_to = "deaths") %>%
  select(-c(Lat, Long))

global <- global_cases %>%
  full_join(global_deaths) %>%
  rename(Country_Region = 'Country/Region',
```

```
         Province_State = 'Province/State') %>%
  mutate(date = mdy(date))

global <- global %>% filter(cases>0)

global <- global %>%
  unite("Combined_Key",
         c(Province_State, Country_Region),
         sep = ", ",
         na.rm = TRUE,
         remove = FALSE)

global <- global %>%
  left_join(uid, by = c("Province_State", "Country_Region")) %>%
  select(-c(UID, FIPS)) %>%
  select(Country_Region, date, cases, deaths, Population, Combined_Key)

summary(global)
```

```
##  Country_Region          date                cases               deaths
##  Length:306827      Min.   :2020-01-22   Min.   :        1   Min.   :      0
##  Class :character   1st Qu.:2020-12-12   1st Qu.:     1316   1st Qu.:      7
##  Mode  :character   Median :2021-09-16   Median :    20365   Median :    214
##                     Mean   :2021-09-11   Mean   :  1032863   Mean   :  14405
##                     3rd Qu.:2022-06-15   3rd Qu.:   271281   3rd Qu.:   3665
##                     Max.   :2023-03-09   Max.   :103802702   Max.   :1123836
##
##    Population        Combined_Key
##  Min.   :6.700e+01   Length:306827
##  1st Qu.:7.866e+05   Class :character
##  Median :6.948e+06   Mode  :character
##  Mean   :2.890e+07
##  3rd Qu.:2.914e+07
##  Max.   :1.380e+09
##  NA's   :6729
```

3. Visualization and Analysis

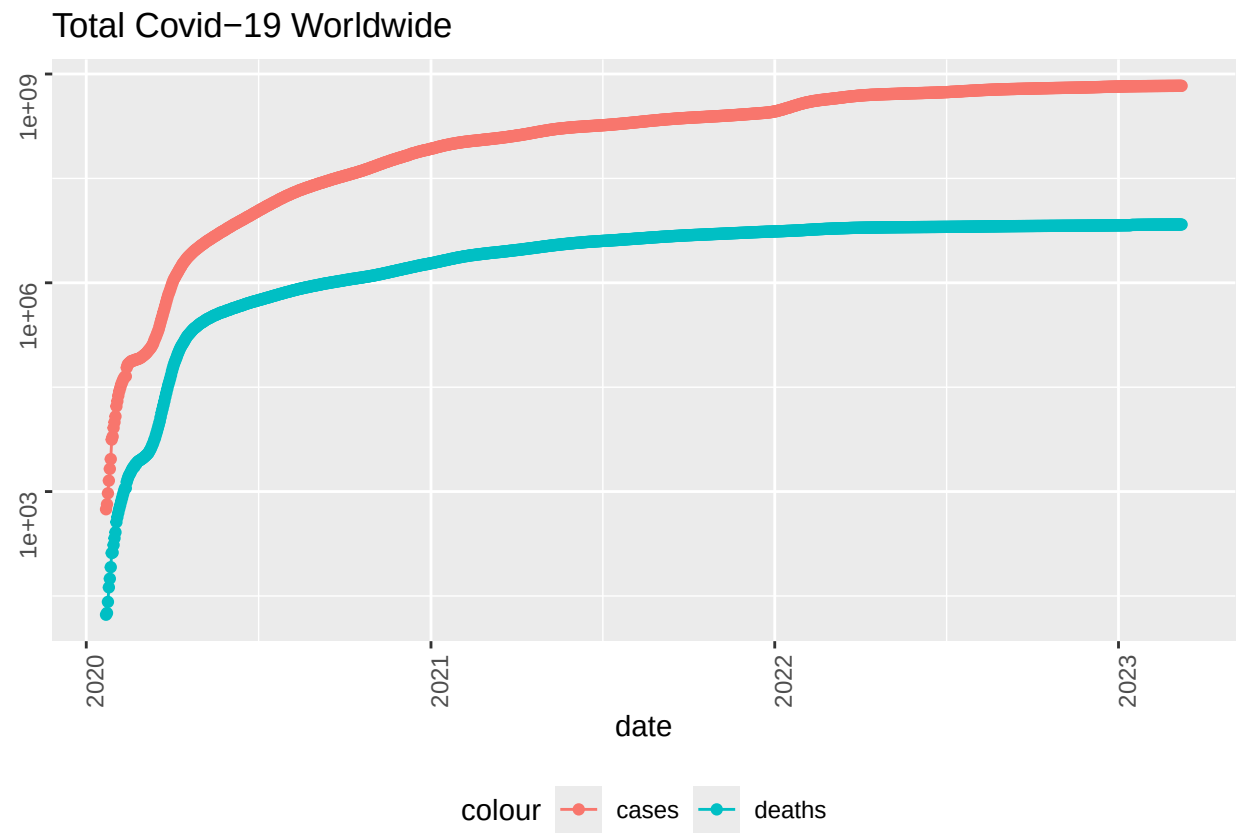
Let start by looking at case and deaths on a worldwide bases.

```
By_Country<-global %>%
  group_by(Country_Region,date) %>%
  summarize(cases=sum(cases),deaths=sum(deaths),
    Population= sum(Population)) %>%
  mutate(deaths_per_mill=deaths *1000000 / Population)%>%
  select(Country_Region,date,cases,deaths,deaths_per_mill,Population) %>%
  ungroup()

Total<-By_Country %>%
  group_by(date)%>%
  summarize(cases=sum(cases),deaths=sum(deaths),
    Population= sum(Population)) %>%
```

```
  mutate(deaths_per_mill=deaths *1000000 / Population)%>%
  select(date,cases,deaths,deaths_per_mill,Population) %>%
  ungroup()
```

```
Total%>%
  filter(cases > 0) %>%
  ggplot(aes(x = date,y = cases))+
  geom_line(aes(color = "cases"))+
  geom_point(aes(color = "cases"))+
  geom_line(aes(y = deaths, color = "deaths"))+
  geom_point(aes(y = deaths, color = "deaths"))+
  scale_y_log10()+
  theme(legend.position="bottom",
    axis.text=element_text(angle=90))+
  labs(title="Total Covid-19 Worldwide",y=NULL)
```



We are filtering deaths to be greater than 10,000. This removes countries that are either too small for meaningful results, or countries that had issues with their reporting.

```
summary_country<-global %>%
 group_by(Country_Region)%>%
 summarise(deaths= max(deaths),cases= max(cases),population=max(Population),
 cases_per_thou=1000* cases/ population,
 deaths_per_thou=1000*deaths / population)%>%
 filter(deaths > 10000,population> 0)
```

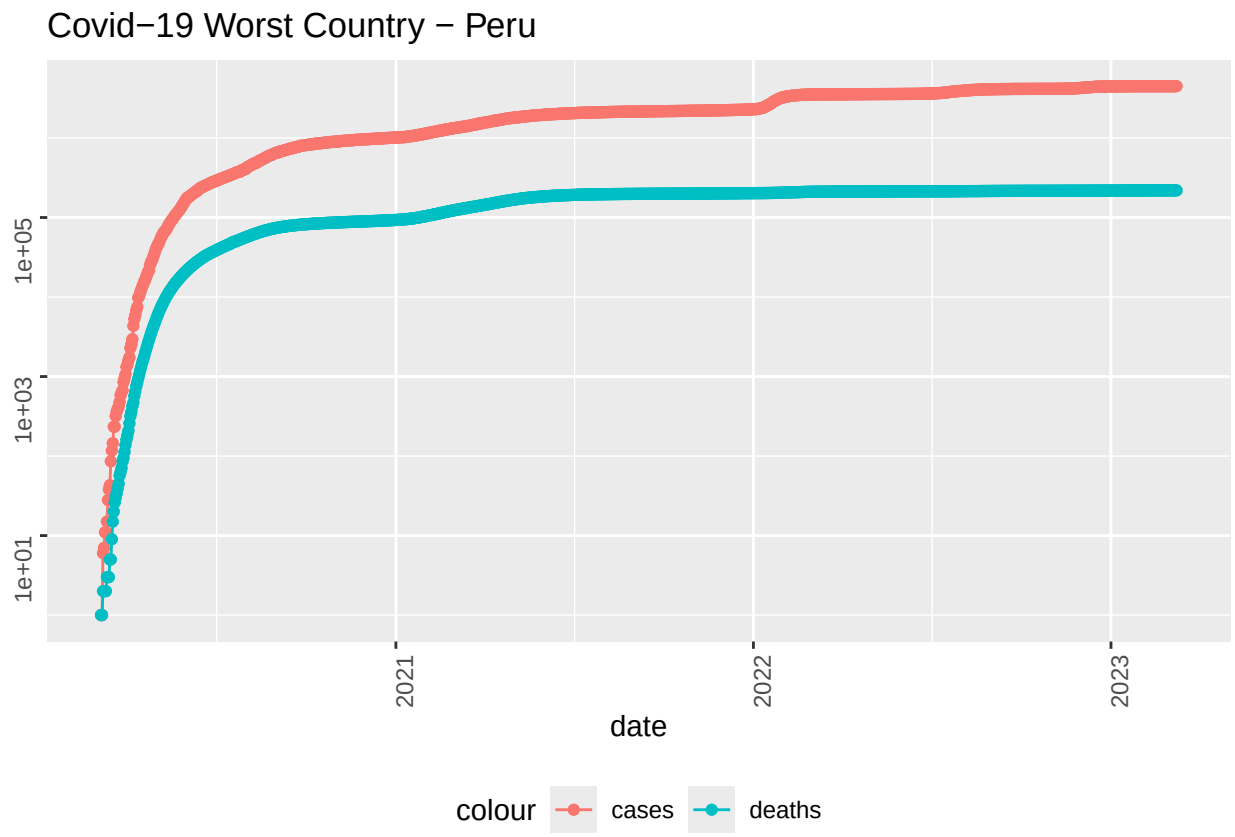
Worst countries (with at least 10,000 deaths):

```
summary_country %>%
  slice_max(deaths_per_thou,n=10) %>%
  select(deaths_per_thou,cases_per_thou, everything())
```

```
## # A tibble: 10 x 6
##    deaths_per_thou cases_per_thou Country_Region        deaths  cases population
##              <dbl>          <dbl> <chr>                 <dbl>  <dbl>      <dbl>
##  1            6.66           136. Peru                  2.20e5 4.49e6   32971846
##  2            5.50           187. Bulgaria              3.82e4 1.30e6    6948445
##  3            5.05           227. Hungary               4.88e4 2.20e6    9660350
##  4            4.96           122. Bosnia and Herzegovi~ 1.63e4 4.02e5    3280815
##  5            4.38           309. Croatia               1.80e4 1.27e6    4105268
##  6            4.25           458. Georgia               1.70e4 1.83e6    3989175
##  7            3.97           431. Czechia               4.25e4 4.62e6   10708982
##  8            3.87           491. Slovakia              2.10e4 2.67e6    5434712
##  9            3.52           174. Romania               6.77e4 3.35e6   19237682
## 10            3.41           315. US                    1.12e6 1.04e8  329466283
```

Looking at Peru's case and death curves.

```
Country<-"Peru"
By_Country%>%
  filter(Country_Region == Country)%>%
  filter(cases >0)%>%
  ggplot(aes(x=date, y=cases))+
  geom_line(aes(color="cases"))+
  geom_point(aes(color="cases"))+
  geom_line(aes(y=deaths,color="deaths"))+
  geom_point(aes(y=deaths,color="deaths"))+
  scale_y_log10()+
  theme(legend.position="bottom",
    axis.text=element_text(angle=90))+
  labs(title="Covid-19 Worst Country - Peru",y=NULL)
```



Best countries (with at least 10,000 deaths):

```
summary_country %>%
  slice_min(deaths_per_thou,n=10) %>%
  select(deaths_per_thou,cases_per_thou, everything())
```

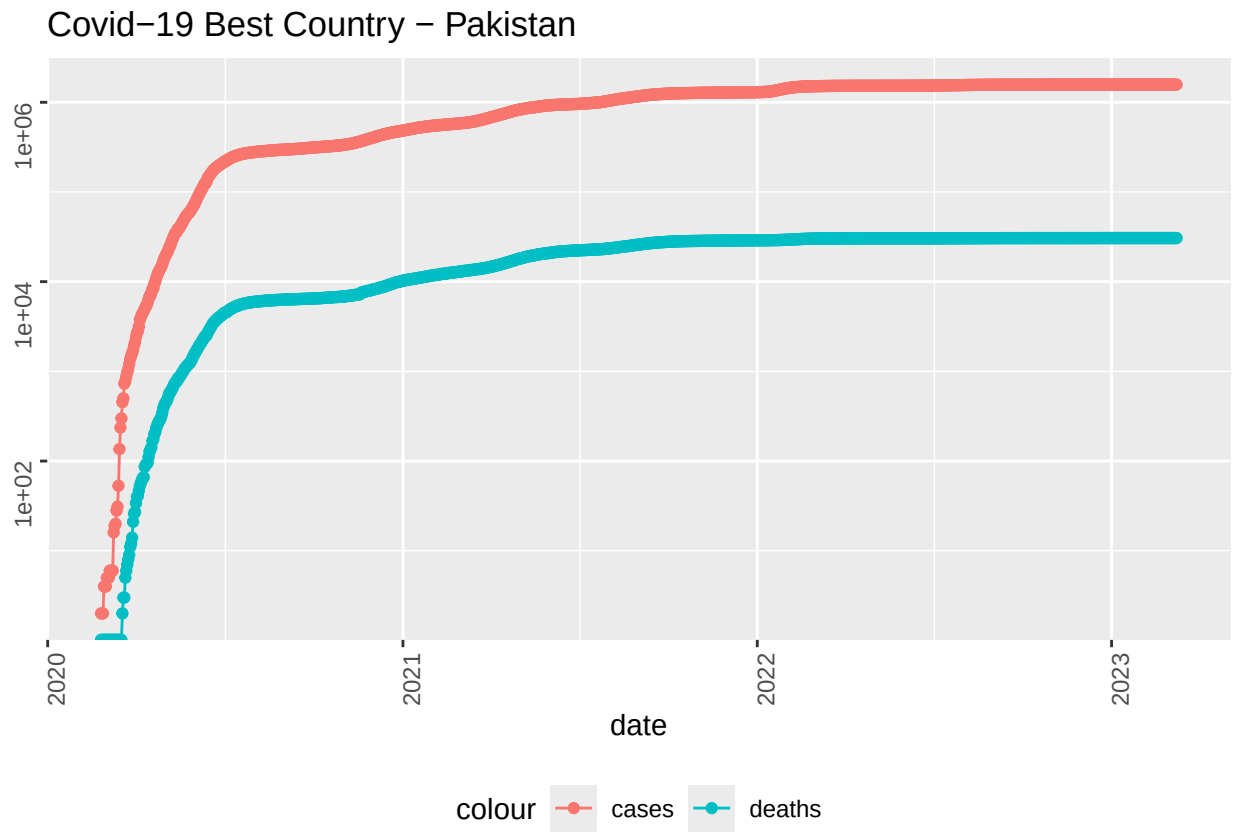
```
## # A tibble: 10 x 6
##    deaths_per_thou cases_per_thou Country_Region deaths    cases population
##              <dbl>          <dbl> <chr>           <dbl>    <dbl>      <dbl>
##  1           0.139           7.14 Pakistan        30644  1577411  220892331
##  2           0.179          12.4  Bangladesh      29445  2037871  164689383
##  3           0.242           5.04 Egypt           24812   515759  102334403
##  4           0.358          11.7  Burma           19490   633950   54409794
##  5           0.385          32.4  India          530779 44690738 1380004385
##  6           0.413          34.4  Nepal           12020  1001154   29136808
##  7           0.441          34.5  Morocco         16296  1272490   36910558
##  8           0.444         118.   Vietnam         43186 11526994   97338583
##  9           0.486          67.7  Thailand        33918  4728182   69799978
## 10           0.577         263.   Japan           72997 33320438  126476458
```

Looking at Pakistan's case and death curves.

```
Country<-"Pakistan"
By_Country%>%
  filter(Country_Region == Country)%>%
  filter(cases >0)%>%
```

```
  ggplot(aes(x=date, y=cases))+
  geom_line(aes(color="cases"))+
  geom_point(aes(color="cases"))+
  geom_line(aes(y=deaths,color="deaths"))+
  geom_point(aes(y=deaths,color="deaths"))+
  scale_y_log10()+
  theme(legend.position="bottom",
    axis.text=element_text(angle=90))+
  labs(title="Covid-19 Best Country - Pakistan",y=NULL)
```

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
## log-10 transformation introduced infinite values.
```



4. Modeling

Lets do a basic model to see the relationship between number of cases and number of deaths.

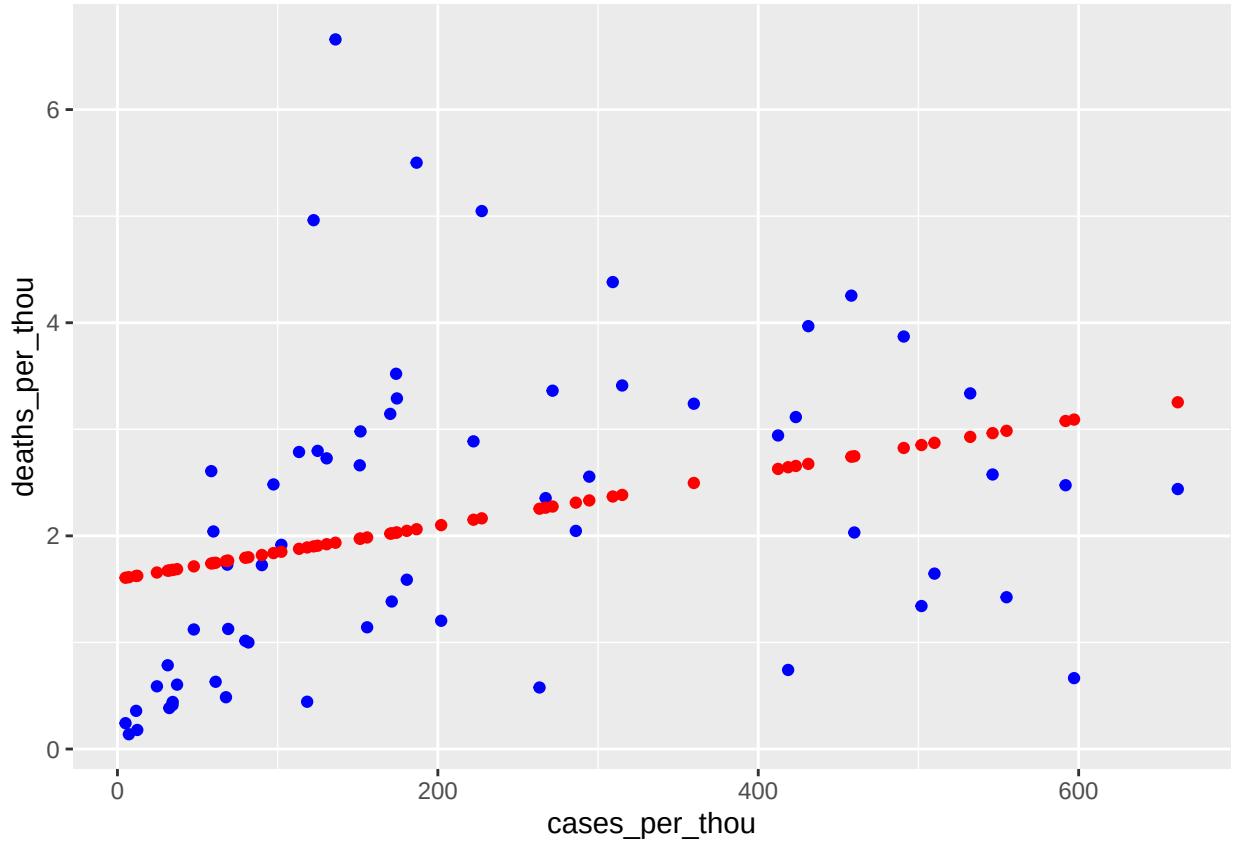
```
mod<-lm(deaths_per_thou ~ cases_per_thou,data=summary_country)
summary(mod)
```

```
##
## Call:
## lm(formula = deaths_per_thou ~ cases_per_thou, data = summary_country)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4264 -1.1002 -0.2651  0.8781  4.7230
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.5941949  0.2785597   5.723  3.4e-07 ***
## cases_per_thou 0.0025071  0.0009701   2.584   0.0122 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.402 on 61 degrees of freedom
## Multiple R-squared:  0.09868,    Adjusted R-squared:  0.08391
## F-statistic: 6.679 on 1 and 61 DF,  p-value: 0.01217
```

```r
Global_tot_w_pred <- summary_country %>% mutate(pred = predict(mod))

Global_tot_w_pred %>% ggplot()+
  geom_point(aes(x = cases_per_thou, y = deaths_per_thou), color = "blue")+
  geom_point(aes(x = cases_per_thou, y = pred), color = "red")
```



5. Conclusion

The analysis reveals a weak linear relationship between the number of cases and deaths, suggesting that the data might better fit a logarithmic curve. Additionally, the shape of the case and death curves showed little variation between the best and worst performing countries, indicating that the disease followed a similar pattern across countries, although there were varying levels of severity.

Bias

It is important to recognize the limitations of the data. Differences in data collection methods across countries may have introduced biases, potentially due to varying resources or classification practices. Furthermore, some countries may have under reported cases to present a more favorable image internationally. Given the political nature of the pandemic, data from certain countries should be viewed with caution.