# Understanding COVID-19

Group F

April 15, 2020

## Download an Additional Dataset on COVID-19 Testing

Save the CSV file from the following link in your working directory before you knit. File name should be **owid-covid-data.csv**.

`https://covid.ourworldindata.org/data/owid-covid-data.csv`

Max Roser, Hannah Ritchie, and Esteban Ortiz-Ospina (2020) - "Coronavirus Source Data". Published online at OurWorldInData.org. Retrieved from: 'https://ourworldindata.org/coronavirus-source-data' [Online Resource]

## Import Packages

```
library(dplyr)
library(ggplot2)
library(scales)
library(lubridate)
library(ggrepel)
# 15 distinct colors
col_vector = c("dodgerblue2", "#E31A1C", "green4", "#6A3D9A", "#FF7F00",
               "black", "gold1", "skyblue2", "palegreen2", "#FDBF6F",
               "gray70", "maroon", "orchid1", "darkturquoise", "darkorange4")
```

## Import COVID-19 Dataset

```
covid19 <- read.csv("COVID19_full_data.csv")
#convert factor to Date class
covid19$date <- as.Date(covid19$date,"%Y-%m-%d")
#convert factor to character
covid19$location <- as.character(covid19$location)
```

# Part 1: Extent of the Pandemic

## 1.1 As of 17 March, there are 15 countries with more than 1000 cases. Which 15 countries are there? Plot the number of total cases over time for each of those 15 countries.

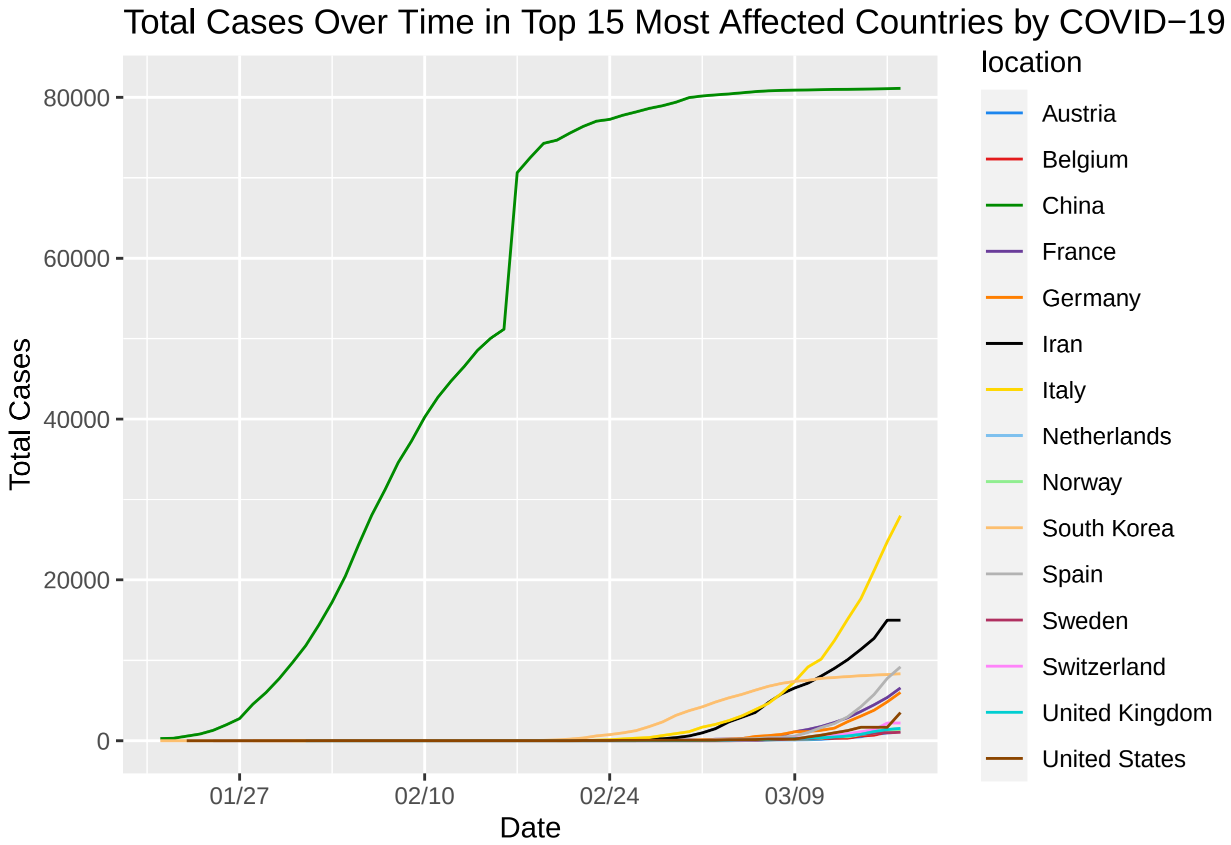
### List countries with more than 1000 total cases on March 17

```r
# We used filter() function to return rows with the following conditions:
# 1. The date is March 17 which is the maximum date in this dataset.
# 2. The number of total cases is above 1000.
# 3. Exclude the 'World' since it is not a country.
countries_1000_cases_17_March <- filter(covid19, date == max(date), total_cases > 1000,
                                        location != "World")
# reorder the countries in the descending order of total cases
countries_desc <- arrange(countries_1000_cases_17_March,desc(total_cases))

# On March 17, only 15 countries had > 1000 cases
# This step is for newer dataset display because more recent data has > 15 countries
countries_names <- countries_1000_cases_17_March$location[1:15]
# print out the top 15 country names
countries_names
```

```
##  [1] "Austria"        "Belgium"        "China"          "France"
##  [5] "Germany"        "Iran"           "Italy"          "Netherlands"
##  [9] "Norway"         "South Korea"    "Spain"          "Sweden"
## [13] "Switzerland"    "United Kingdom" "United States"
```

**Plot the number of total cases over time for each of those 15 countries using geom_line() function of ggplot package. All countries, as distinguished from their different color are plotted together. In this way, we can easily compare one country with another**

```r
# select all the data from that 15 countries
countries_1000_cases <- filter(covid19, location %in% countries_names)

#----------PLOTTING------------
# plot by location (color = location)
plot <- ggplot(countries_1000_cases, aes(x=date, y=total_cases, color=location))
p1 <- plot + geom_line()
# adjust the labels and breaks of X axis to prevent overlapping labels
p2 <- p1 + scale_x_date(labels = date_format("%m/%d"), breaks = date_breaks("2 weeks"))
# add title and annotate the axes
p3 <- p2 + ggtitle("Total Cases Over Time in Top 15 Most Affected Countries by COVID-19") +
  xlab("Date") + ylab("Total Cases")
# use custom colors for the lines
p4 <- p3 + scale_color_manual(values=c(col_vector, nrow(countries_names)))
p4
```



This plot shows that China is the most worst-affected country globally. The total cases in the country increased significantly before March, especially between the 16th and 17th of February where more than 19,000 new cases were added. After March, the trend tends to flatten out.

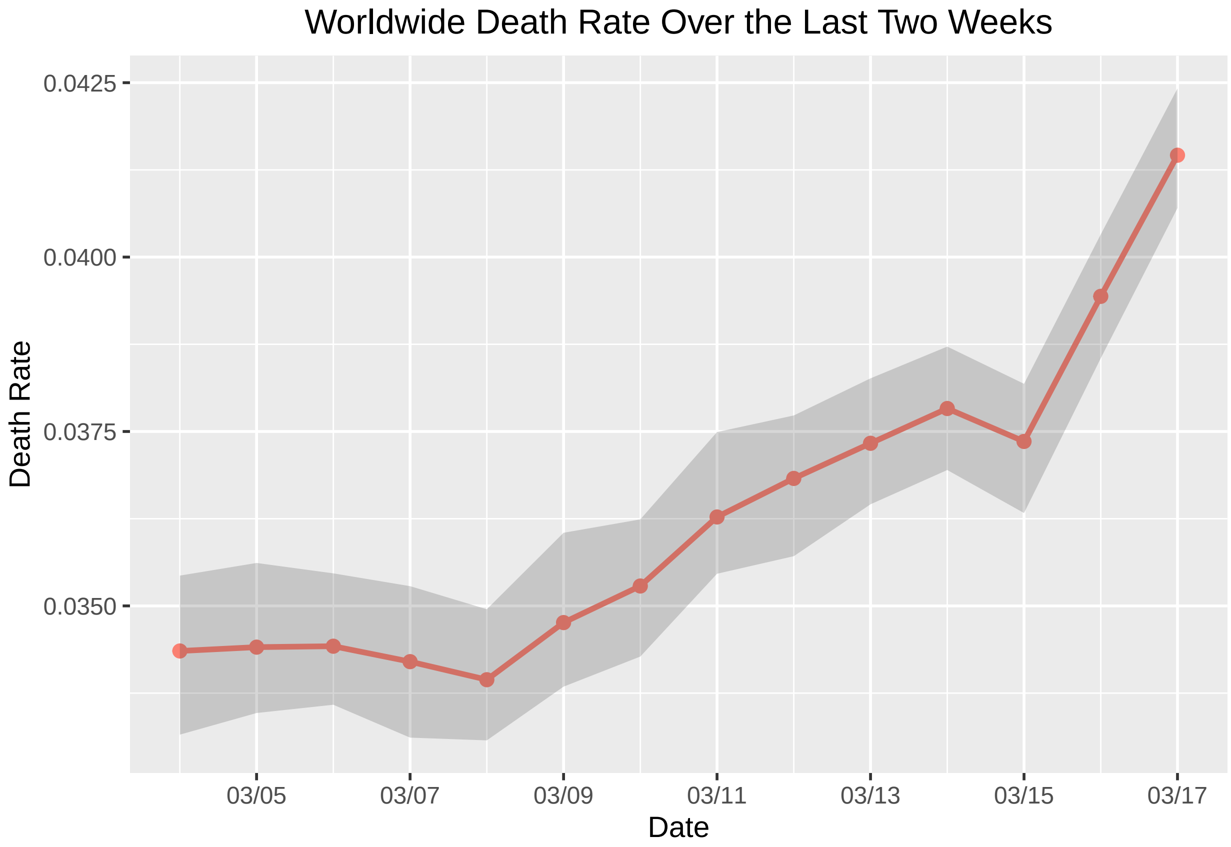
In other countries such as Italy, however, COVID-19 began to break out in March. If we assume that the speed of disease spreading inside each country is the same, the total cases in these 15 countries should be related to their onset of spread, that is the more cases a country has, the earlier the virus may have begun to spread in that country. From the plot, we can see that the outbreak in Italy, which is the second worst-affected country by COVID19, is approximately one month after the outbreak in China. Also, the line for South Korea flattens in the early stage, which may be due to their extensive testing.

### 1.2 For the last 2 weeks (4 March to 17 March), look at the total number of deaths and the total number of cases worldwide. For every day during that period, provide an estimate of the death rate (total deaths divided by total cases), including a confidence interval, for each day in those two weeks. Plot that data.

We implemented bootstrap method to estimate the confidence interval of the death rate daily. Geom_point() shows the data point of death_rate. Geom_ribbon() shows the confidence interval for each estimated death_rate.

```r
# extract world's data in the last 2 weeks (4 March to 17 March)
world <- filter(covid19, date %in% seq.Date(from=max(date)-13,
         to=as.Date(max(date)), by="1 day"), location == "World")

# initialize empty vectors to store the death rate and
# lower and upper values of the confidence interval
death_rate <- c()
lower <- c()
upper <- c()

#------------BOOTSTRAPPING--------------
# loop to extract total death and total cases for each time point
for (i in 1:nrow(world)) {
  death <- world$total_deaths[i]
  case <- world$total_cases[i]
  # obtain the number of infected people who are alive
  survival <- case-death
  # pool the death and survival data
  obs_sample <- c(rep("death", death),rep("survival", survival))
  bootstrap_death <- vector()
    # loop to sample many times
    for (b in 1:100) {
      # sample from the pooled data with replacement
      bootstrap_sample <- sample(obs_sample, length(obs_sample), replace = T)
      # store the number of death cases from sampling into a vector
      bootstrap_death <- c(bootstrap_death, length(subset(bootstrap_sample,
                           bootstrap_sample == "death")))
    }
  #calculate 95% confidence intervals
  lower_ci<-as.numeric(quantile(bootstrap_death, 0.025))
  upper_ci<-as.numeric(quantile(bootstrap_death, 0.975))

  # calculate the death rate (death/case) and store the death rate and confidence intervals
  death_rate <- c(death_rate,death/case)
  lower <-c(lower,lower_ci/case)
  upper <-c(upper,upper_ci/case)
}

# create a dataframe for plotting
df <- data.frame(world$date, death_rate, lower, upper)

#---------PLOTTING--------------
plot <- ggplot(df, aes(x = world.date, y = death_rate))
# plot lines
```

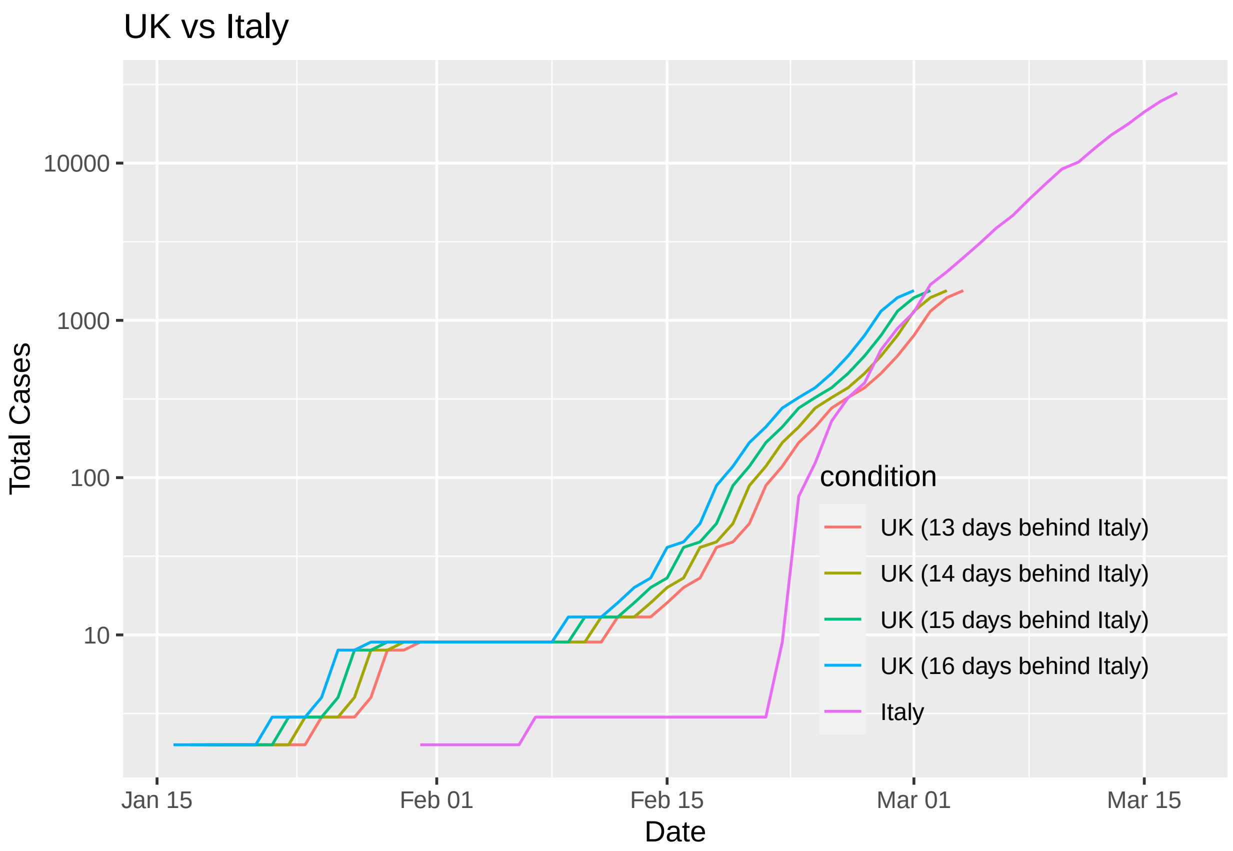
```
p1 <- plot + geom_line(colour = "salmon", size = 1)
# emphasize each data point
p2 <- p1 + geom_point(colour = "salmon", size = 2)
# visualize the confidence interval using geom_ribbon()
p3 <- p2 + geom_ribbon(aes(ymin=lower, ymax=upper), alpha=0.2)
# adjust the labels and breaks of X axis
p4 <- p3 + scale_x_date(labels = date_format("%m/%d"), breaks = date_breaks("2 days"))
# add title and annotate the axes
p5 <- p4 + ggtitle("Worldwide Death Rate Over the Last Two Weeks") +
  # center the plot title using hjust
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab("Date") + ylab("Death Rate")
p5
```



In general, the death rate has been rising since March 4. The confidence interval gradually narrows down because more and more cases were reported, indicating the death rate calculated is getting more acurate. Also, we should be aware of the fact that this is not the true death rate: the reported death cases should have been diagnosed several days earlier. Thus, the total deaths should be divided by the total cases from several days ago, but we do not know the exact interval. In March, there were more new cases reported everyday, hence total deaths are actually divided by a larger denominator, which means that the death rate is underestimated.

**1.3 Plot the case numbers over time for both Italy and the UK. Italy has been the first country in Europe to see a major outbreak of COVID-19. In the UK, media have reported that the UK is around 14 days "behind" Italy. Can you use a plot to check whether this is plausible?**

```r
# select data from UK
UK <- filter(covid19, location == "United Kingdom")
# convert factor to date
UK$date <- as.Date(UK$date)
# add another column called altered_days
# to store the number of days we changed from the actual date
# this column is also for distinguishment
UK$altered_days <- NA
# create a new data frame to store all the altered information
UK_b <- c()

# minus 13, 14, 15, 16 days from the UK date
for (d in 13:16) {
  UK_a <- UK
  UK_a$altered_days <- d
  UK_a$date <- UK$date-d
  # combine all the altered date into one dataframe using rbind()
  UK_b <- rbind(UK_b,UK_a)
}

# select data from Italy
Italy <- filter(covid19, location=="Italy")
# this column is for distinguishment (helpful for plotting)
Italy$altered_days <- "Italy"
# add Italy data to the dataframe
UnI <- rbind(UK_b,Italy)

#-------------PLOTTING---------------
plot <- ggplot(UnI, aes(x = date, y = total_cases, col = altered_days))
p1 <- plot + geom_line()
# change the scale of y axis to logarithmic to make the plot clearer
p2 <- p1 + scale_y_continuous(trans = "log10")
# annotate the axes
p3 <- p2 + labs(title = "UK vs Italy", x = "Date", y = "Total Cases", col = "condition")
# place the legend inside the plot
p4 <- p3 + theme(legend.position = c(.78,.25), legend.background = element_blank())
# labels for the different colors
p5 <- p4+ scale_color_discrete(labels = c("UK (13 days behind Italy)",
                                  "UK (14 days behind Italy)",
                                  "UK (15 days behind Italy)",
                                  "UK (16 days behind Italy)",
                                  "Italy"))
p5
```



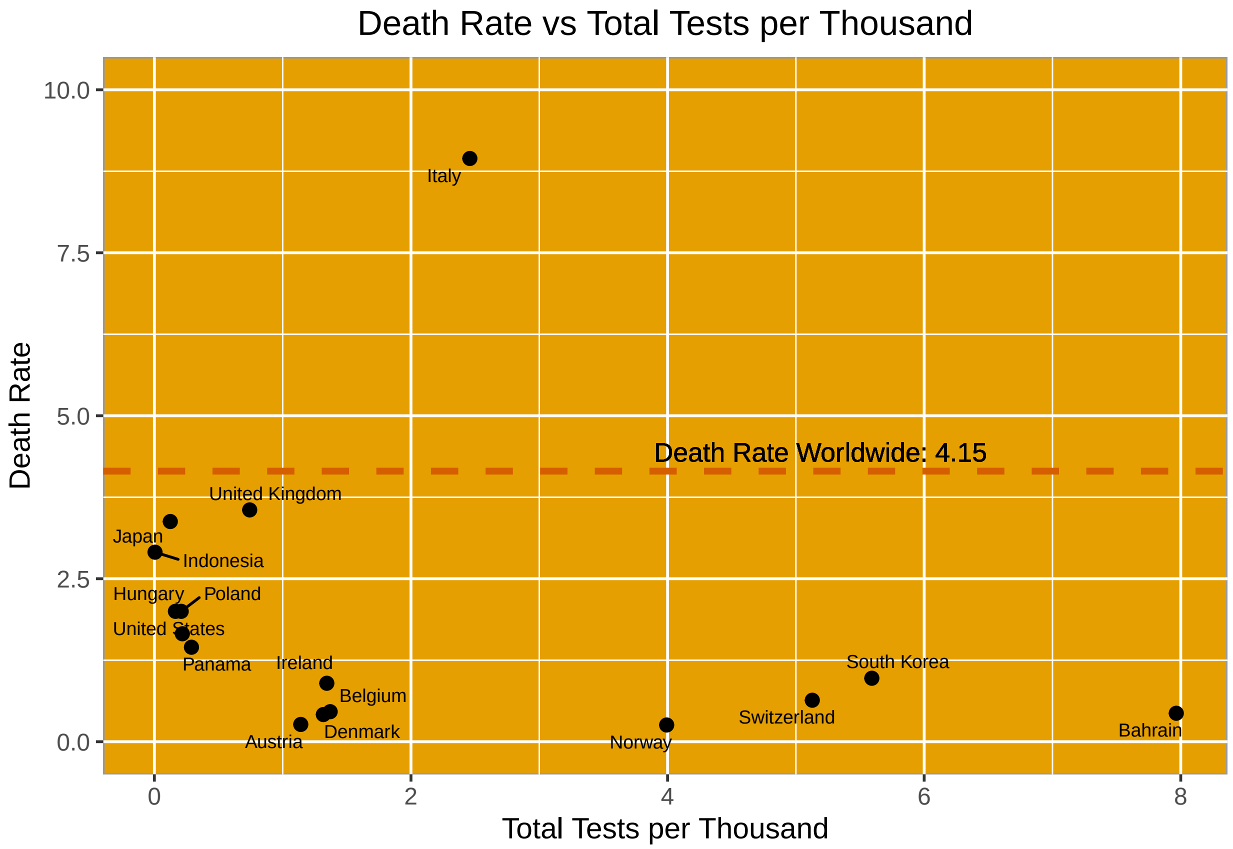
To prove that the UK is around 14 days “behind” Italy, we compared the total cases in Italy with the total cases in the UK 12, 13, 14, 15 and 16 days later. We compared the total cases directly because the population densities of **West Midland (UK)** and **Lombardia (Italy)** where the outbreak started, are similar. The total cases also increased exponentially, so we changed the scale of the y axis to logarithmic in order to make the figure clearer.

From the figure, we can see that there is an overlap between 100 to 1000 cases, which can prove that the UK is around 14 days behind Italy. The curves below 100 total cases may not overlap, but this is due to logarithmic linearization, which makes these relatively small differences in small numbers look large. These small differences (within 100) can be ignored.

## Part 2: Death Rate vs Testing Rate

After seeing the flattened line of South Korea from part 1.1, we were curious about the effect of extensive testing. We found a testing dataset containing some of the countries and wanted to see whether there is a relationship between testing number and death rate. The hypothesis is that the more testing a country does, the more controlled the disease is under and thus the death rate is lower. First, we plot the death rate against total test per thousand people for all the countries that have the test data to get an overview.

```r
latest_date <- max(covid19$date)
# select all the data from the most recent date
covid19_latest <- filter(covid19, date == latest_date)
# calculate the death rates (total death/total cases)*100
# and store them into a new column death_rate
covid19_latest$death_rate <- (covid19_latest$total_deaths/covid19_latest$total_cases)*100


#-----------SPECIFY THE TESTING DATA BELOW-------------
test <- read.csv("owid-covid-data.csv")
# convert from factor to Date
test$date <- as.Date(test$date,"%Y-%m-%d")
# convert from factor to character
test$location <- as.character(test$location)
# extract all the testing data that match the latest date
test_latest <- filter(test, date == latest_date)
# keep only the needed columns
test_latest <- test_latest[,c("location","total_tests_per_thousand")]

# merge the covid19 dataset and testing dataset using left_join()
# the function includes everyting in the first dataset (covid19) and
# look for matches in the second dataset (testing)
testing <-left_join(covid19_latest, test_latest, by = "location")

# calculate world death rate, which will serve as a reference line in the plot
world_death_rate <- round(filter(testing, location == "World")$death, digits = 2)
# exclude NA values
testing <- testing[complete.cases(testing),]

#------------PLOTTING---------------
p <- ggplot(testing, aes(x = total_tests_per_thousand, y = death_rate, label = location))
p1 <- p + geom_point(size = 2)
# add a dashed line to indicate death rate worldwide
p2 <- p1 + geom_hline(yintercept = world_death_rate,
             color = "#D55E00",
             linetype = "dashed",
             size = 1.2)
# explanatory text for the dashed line
p3 <- p2 + geom_text(aes(0, world_death_rate,
                         label = paste("Death Rate Worldwide:", world_death_rate)),
                     hjust = -1.5, vjust = -0.5, size = 3.5)
# scale the y axis
p4 <- p3 + ylim(0,10)
# use geom_text_repel under ggrepel package to prevent overlapping labels
p5 <- p4 + geom_text_repel(size = 2.5)
p6 <- p5 + xlab("Total Tests per Thousand") + ylab("Death Rate") +
          ggtitle("Death Rate vs Total Tests per Thousand") +
          # center the title
          theme(plot.title = element_text(hjust = 0.5))
# change the background color for the plot
p7 <- p6 + theme(panel.background = element_rect(fill = '#E69F00', colour = '#999999'))
p7
```



Generally, we can see a trend that the more tests a country does, the lower the death rate is. Countries that are doing relatively high levels of testing such as Bahrain, South Korea, Switzerland and Norway report lower death rate. On the other hand, the plot shows that countries with limited testing have higher death rate. Italy, however, seems to be an outlier due to its particularly high death rate. It may be that Italy's realtively elderly population affects their death rate.

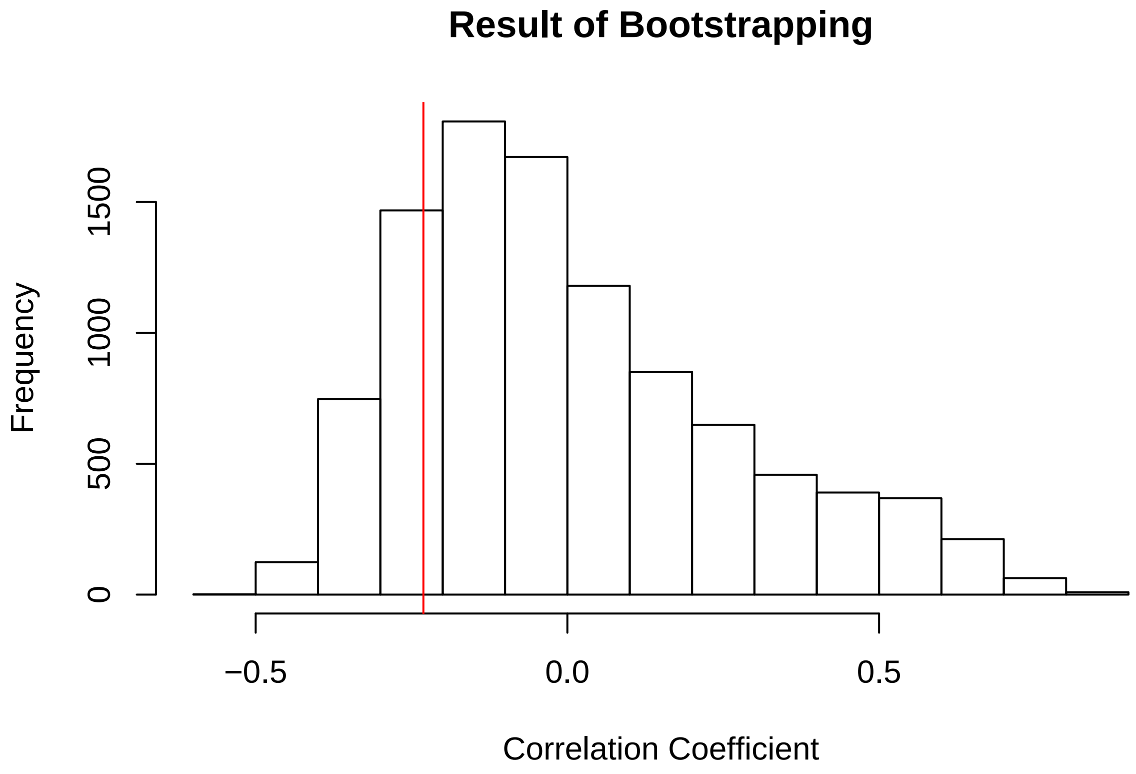
Then, we want to further prove the relationship between death rate and testing. We could not meet the assumptions for correlation, so we decided to do bootstrapping based on correlation coefficient. Though we cannot use the p-value from cor.test directly because the assumption is not met, the correlation coefficient still reflect the relationship between death rate and testing. The closer it is to -1, the stronger the relationship. So we reshuffle the death rate corresponding to test rates and see the posibility of getting a smaller correlation coefficient than the current one.

```
# calculate the correlation coefficient
cor <- cor.test(testing$death_rate, testing$total_tests_per_thousand)$estimate
# create a sample for bootstrapping
sample <- testing

#------------BOOTSTRAPPING-----------
res <- NULL
```

```
# loop to reshuffle death rate and test per thousand people
for (i in 1:10000) {
  #reshuffle
  sample$death_rate <- sample(sample$death_rate)
  a <- cor.test(sample$death_rate, sample$total_tests_per_thousand)
  res <- c(res, a$estimate)
}
# this is a negative relationship, so we use <
p <- length(which(res < cor))/10000
p
```

```
## [1] 0.1787
```

```
# plot the result from bootstrapping
hist(res, xlab="Correlation Coefficient", main="Result of Bootstrapping")
abline(v=cor, col="red")
```



The p value is bigger than 0.05. We cannot reject the null hypothesis, which means that there is not enough evidence to prove that there is a negative relationship between death rate and testing. The reason could be there is indeed no relationship between death rate and testing or it may be because the pandemic has just started and the data on March 17 is not representative enough. We need future data with more countries provided with their testing data to draw valid conclusions.