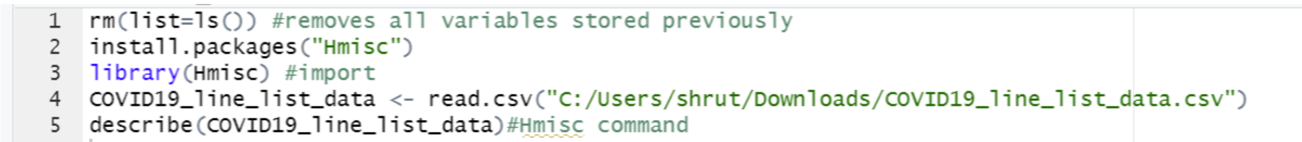
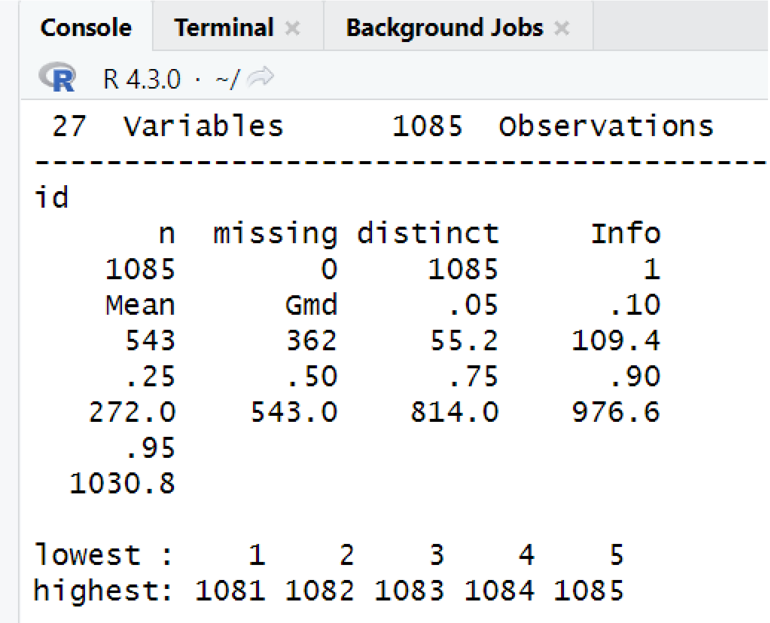
COVID-19 DATA ANALYSIS USING R PROGRAMMING

Step 1: Describing and Getting to know about the dataset:



```
rm(list=ls()) #removes all variables stored previously
install.packages("Hmisc")
library(Hmisc) #import
COVID19_line_list_data <- read.csv("C:/Users/shrut/Downloads/COVID19_line_list_data.csv")
describe(COVID19_line_list_data)#Hmisc command
```



```
 27  Variables      1085  Observations
---------------------------------------
id
       n  missing distinct     Info
    1085        0     1085        1
    Mean      Gmd      .05      .10
     543      362     55.2    109.4
     .25      .50      .75      .90
   272.0    543.0    814.0    976.6
     .95
  1030.8

lowest :    1    2    3    4    5
highest: 1081 1082 1083 1084 1085
```

Based on the provided COVID-19 dataset, here is an analysis of the variables:

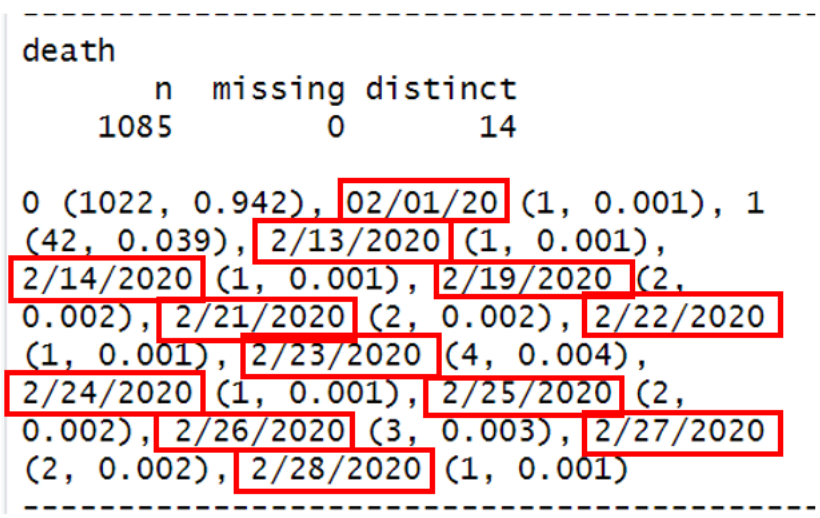
1. **id:** This variable represents the unique identifier for each observation. There are 1085 observations with no missing values. The range of values is from 1 to 1085.
2. **case_in_country:** This variable represents the case number in the country. There are 888 observations with 197 missing values. The range of values is from 1 to 365.
3. **reporting.date:** This variable represents the date of reporting. There are 1084 observations with 1 missing value. The range of dates is from February 1, 2020, to February 28, 2020.
4. **summary:** This variable provides a summary of the case. There are 1080 observations with 5 missing values. Based on the provided COVID-19 dataset, here is an analysis of the variables:
5. **case_in_country:** This variable represents the case number in the country. There are 888 observations with 197 missing values. The range of values is from 1 to 365.

6. **reporting.date:** This variable represents the date of reporting. There are 1084 observations with 1 missing value. The range of dates is from February 1, 2020, to February 28, 2020.
7. **location:** This variable represents the location of the case. There are 1085 observations with no missing values. There are 156 distinct locations.
8. **country:** This variable represents the country of the case. There are 1085 observations with no missing values. There are 38 distinct countries.
9. **gender:** This variable represents the gender of the case. There are 902 observations with 183 missing values. The majority are male (520) and female (382).
10. **age:** This variable represents the age of the case. There are 843 observations with 242 missing values. The average age is 49.48, and the range of values is from 0.25 to 96.
11. **symptom_onset:** This variable represents the date of symptom onset. There are 563 observations with 522 missing values. The range of dates is from January 2, 2020, to February 26, 2020.
12. **If_onset_approximated:** This variable indicates if the symptom onset date is approximate. There are 560 observations with 525 missing values. The sum of approximate onset dates is 24.
13. **hosp_visit_date:** This variable represents the date of hospital visit. There are 507 observations with 578 missing values. The range of dates is from January 1, 2020, to February 28, 2020.
14. **exposure_start:** This variable represents the start date of exposure. There are 128 observations with 957 missing values. The range of dates is from January 3, 2020, to February 21, 2020.
15. **exposure_end:** This variable represents the end date of exposure. There are 341 observations with 744 missing values. The range of dates is from January 2, 2020, to February 25, 2020.
16. **visiting.Wuhan:** This variable indicates if the case visited Wuhan. There are 1085 observations with no missing values. The sum of cases that visited Wuhan is 192.
17. **from.Wuhan:** This variable indicates if the case is from Wuhan. There are 1081 observations with 4 missing values. The sum of cases from Wuhan is 156.
18. **death:** This variable indicates if the case resulted in death. There are 1085 observations with no missing values. There are 14 distinct values indicating different death cases.
19. **recovered:** This variable indicates if the case recovered. There are 1085 observations with no missing values. There are 32 distinct values indicating different recovery cases.
20. **symptom:** This variable represents the symptoms reported by the case. There are 270 observations with 815 missing values. There are 108 distinct symptoms.

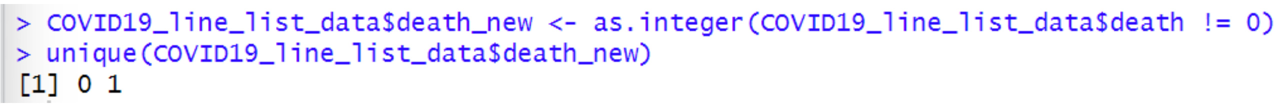
21. **source:** This variable represents the source of the information. There are 1085 observations with no missing values. There are 85 distinct sources.

Step 2: Cleaning the data:

Death variable has **0 -> if dead** and **1 -> if not dead**, but there are few date values too in this column which will be difficult to work with. Hence, we need to clean this column.



```
------------------------------------------
death
       n  missing distinct
    1085        0       14

0 (1022, 0.942), 02/01/20 (1, 0.001), 1
(42, 0.039), 2/13/2020 (1, 0.001),
2/14/2020 (1, 0.001), 2/19/2020 (2,
0.002), 2/21/2020 (2, 0.002), 2/22/2020
(1, 0.001), 2/23/2020 (4, 0.004),
2/24/2020 (1, 0.001), 2/25/2020 (2,
0.002), 2/26/2020 (3, 0.003), 2/27/2020
(2, 0.002), 2/28/2020 (1, 0.001)
------------------------------------------
```



```
> COVID19_line_list_data$death_new <- as.integer(COVID19_line_list_data$death != 0)
> unique(COVID19_line_list_data$death_new)
[1] 0 1
```

Creating a new variable named "death_new" in the "COVID19_line_list_data" dataset. The purpose of this code is to assign a value of 1 to the "death_new" variable if the corresponding "death" variable is not equal to 0, and 0 otherwise.

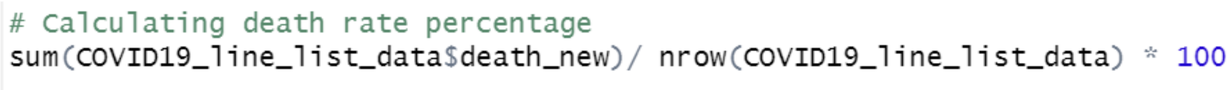
Here's a breakdown of the code:

1. **COVID19_line_list_data$death != 0**: This part compares each value in the "death" variable of the "COVID19_line_list_data" dataset to 0. It returns a logical vector where each element is **TRUE** if the corresponding "death" value is not equal to 0, and **FALSE** otherwise.
2. **as.integer()**: This function converts the logical vector obtained in the previous step to an integer vector. It assigns a value of 1 to **TRUE** elements and a value of 0 to **FALSE** elements.

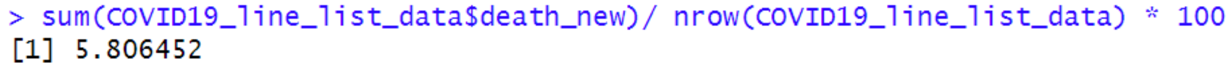
3. **COVID19_line_list_data$death_new <-**: This assigns the resulting integer vector to a new variable called "death_new" within the "COVID19_line_list_data" dataset.

Step 3: Performing Data Analysis:

Calculating the death rate due to COVID-19 in percentage:

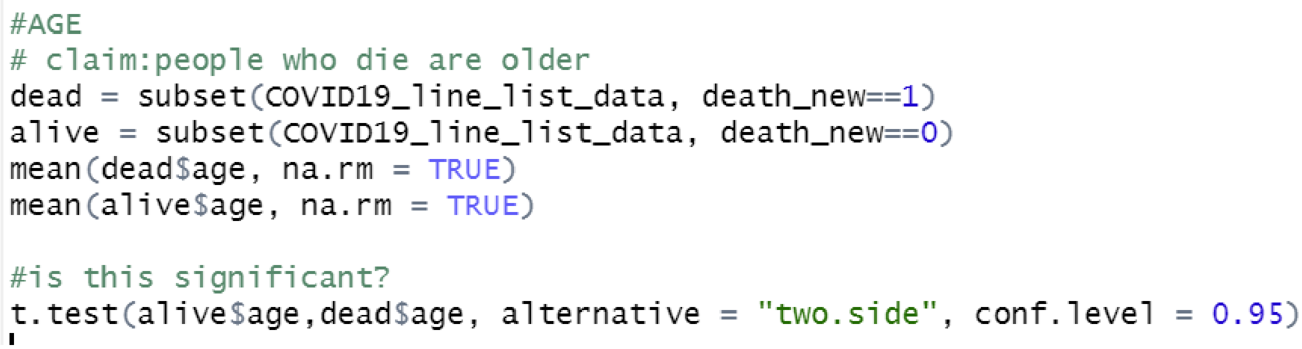


```
# Calculating death rate percentage
sum(COVID19_line_list_data$death_new)/ nrow(COVID19_line_list_data) * 100
```

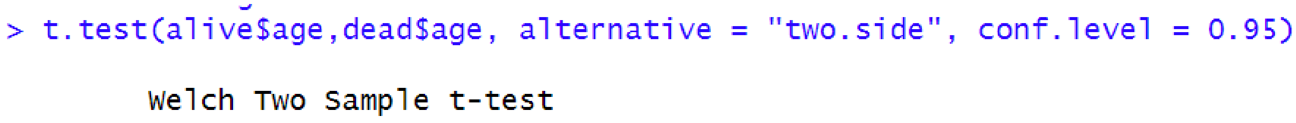


```
> sum(COVID19_line_list_data$death_new)/ nrow(COVID19_line_list_data) * 100
[1] 5.806452
```

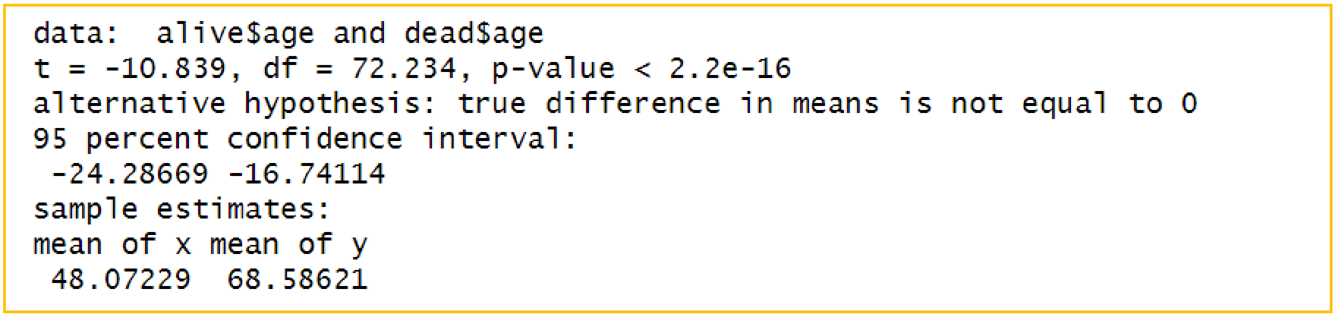
There was a claim that people who die from COVID-19 tend to be older on average. So, let's try to prove this claim statistically:



```
#AGE
# claim:people who die are older
dead = subset(COVID19_line_list_data, death_new==1)
alive = subset(COVID19_line_list_data, death_new==0)
mean(dead$age, na.rm = TRUE)
mean(alive$age, na.rm = TRUE)

#is this significant?
t.test(alive$age,dead$age, alternative = "two.side", conf.level = 0.95)
```



```
> t.test(alive$age,dead$age, alternative = "two.side", conf.level = 0.95)

        Welch Two Sample t-test
```



```
data:  alive$age and dead$age
t = -10.839, df = 72.234, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -24.28669 -16.74114
sample estimates:
mean of x mean of y 
 48.07229  68.58621
```

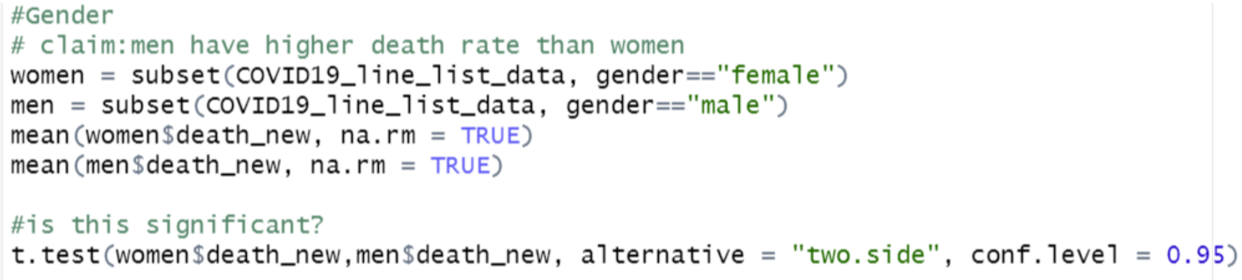
1. **Mean Age Comparison:** The t-test provides the means of the two groups being compared. In this case, the mean age of the alive group is 48.07229, while the mean age of the dead group is 68.58621. This suggests that, on average, individuals who have died from COVID-19 are older than those who are still alive.

2. **Statistical Significance:** The t-test also provides the p-value, which is a measure of the statistical significance of the observed difference in means. In this case, the p-value is less than 2.2e-16 (a very small value), indicating that the observed difference in means is statistically significant. This means that the difference in mean age between the alive and dead groups is unlikely to have occurred by chance alone.

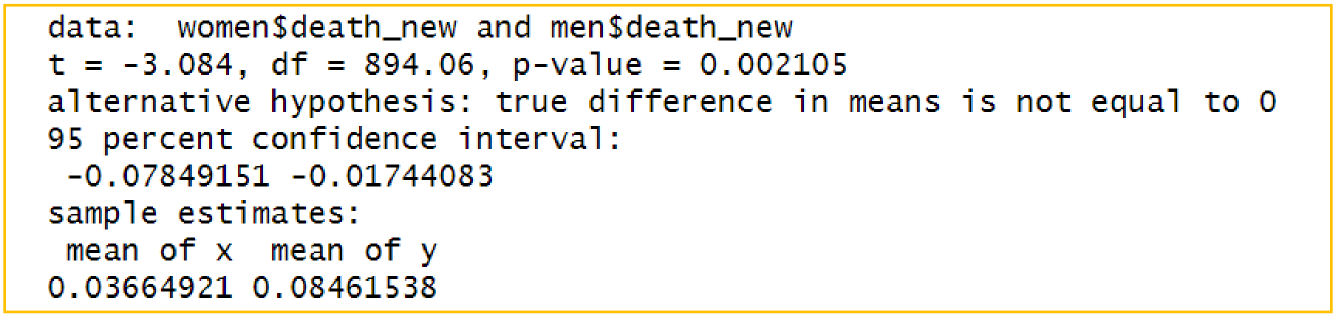
3. **Confidence Interval:** The 95 percent confidence interval (-24.28669, -16.74114) provides a range of values within which the true difference in means is likely to fall. Since the interval does not include zero, it suggests that there is a significant difference between the mean ages of the two groups.

Based on these findings, we can conclude that there is a statistically significant difference in the mean age between individuals who have died from COVID-19 and those who are still alive. **The data support the claim that people who die from COVID-19 tend to be older on average.**

Similarly, there was another claim that men have a higher death rate compared to women. So, let's try to prove this claim statistically:



```
#Gender
# claim:men have higher death rate than women
women = subset(COVID19_line_list_data, gender=="female")
men = subset(COVID19_line_list_data, gender=="male")
mean(women$death_new, na.rm = TRUE)
mean(men$death_new, na.rm = TRUE)

#is this significant?
t.test(women$death_new,men$death_new, alternative = "two.side", conf.level = 0.95)
```



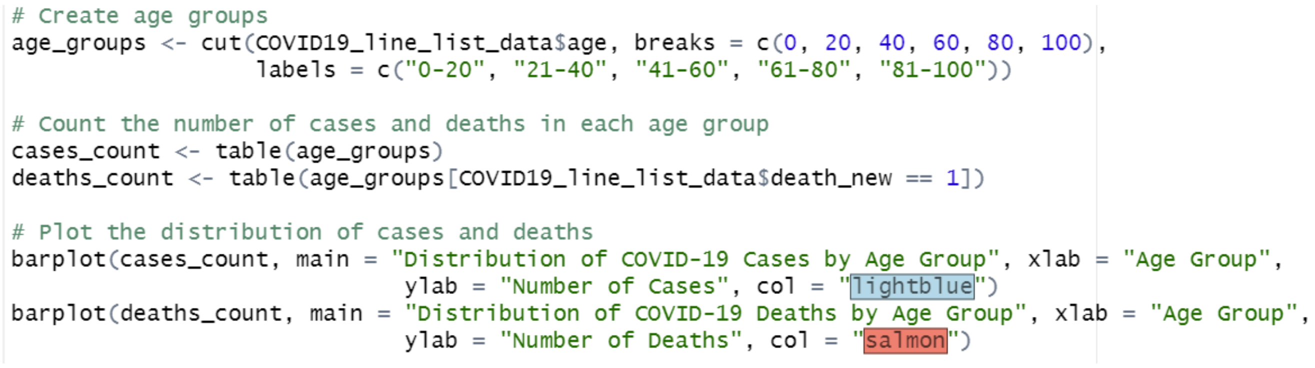
```
data:  women$death_new and men$death_new
t = -3.084, df = 894.06, p-value = 0.002105
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.07849151 -0.01744083
sample estimates:
 mean of x  mean of y
0.03664921 0.08461538
```

1. **Mean Death Rate Comparison:** The t-test provides the means of the two groups being compared. In this case, the mean death rate for women is 0.03664921, while the mean death rate for men is 0.08461538. This suggests that, on average, men have a higher death rate from COVID-19 compared to women.

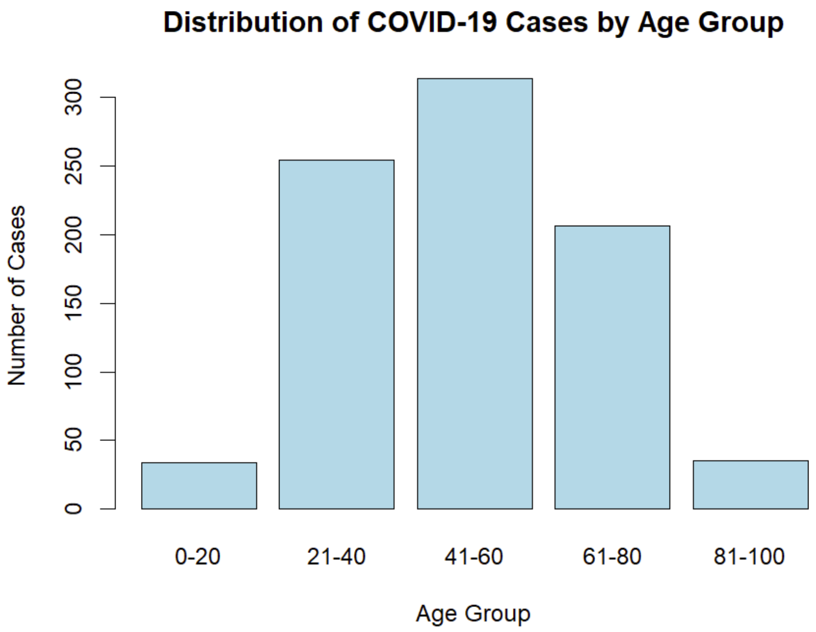
2. **Statistical Significance:** The t-test also provides the p-value, which is a measure of the statistical significance of the observed difference in means. In this case, the p-value is 0.002105, which is less than the commonly used threshold of 0.05. This indicates that the observed difference in death rates between men and women is statistically significant. The difference is unlikely to have occurred by chance alone.
3. **Confidence Interval:** The 95 percent confidence interval (-0.07849151, -0.01744083) provides a range of values within which the true difference in means is likely to fall. Since the interval does not include zero, it suggests that there is a significant difference in death rates between men and women.

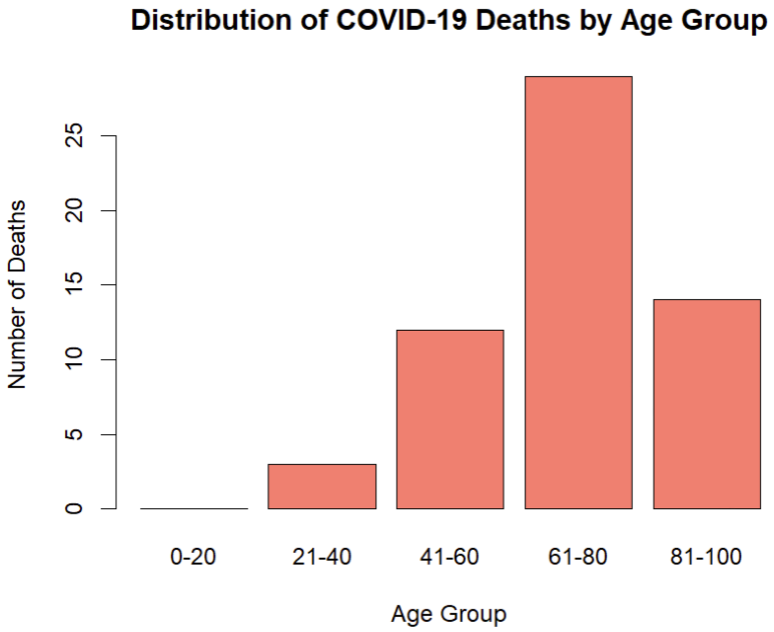
Based on these findings, we can conclude that there is a statistically significant difference in the death rates between men and women with regards to COVID-19. **The data support the claim that men have a higher death rate compared to women.**

Creating age groups and plotting bar-graphs to show the distribution of COVID-19 cases and Covid-19 Deaths by Age Groups:

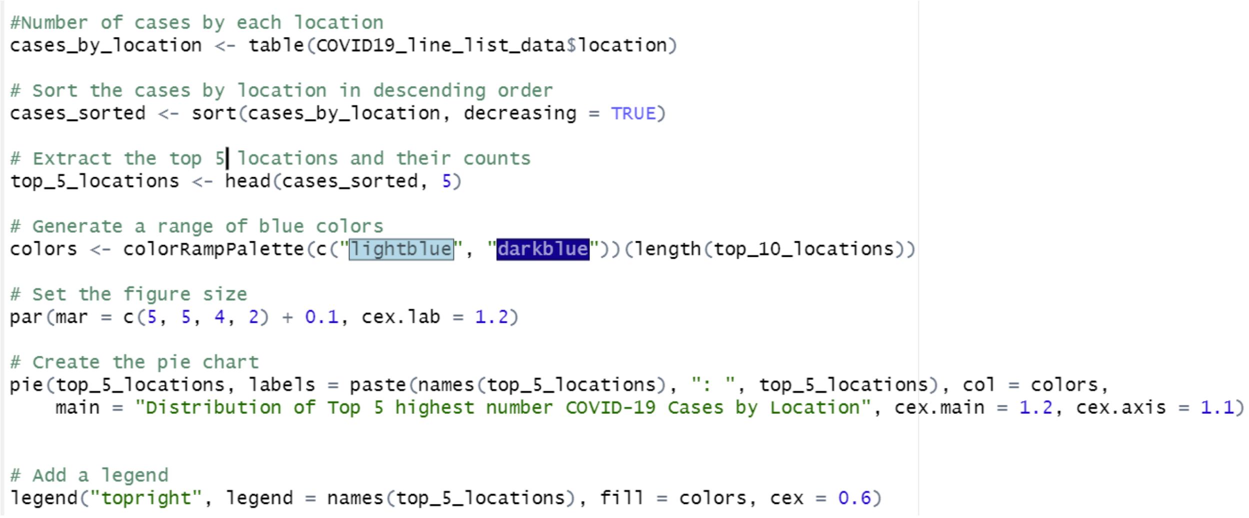


```
# Create age groups
age_groups <- cut(COVID19_line_list_data$age, breaks = c(0, 20, 40, 60, 80, 100),
                  labels = c("0-20", "21-40", "41-60", "61-80", "81-100"))

# Count the number of cases and deaths in each age group
cases_count <- table(age_groups)
deaths_count <- table(age_groups[COVID19_line_list_data$death_new == 1])

# Plot the distribution of cases and deaths
barplot(cases_count, main = "Distribution of COVID-19 Cases by Age Group", xlab = "Age Group",
                             ylab = "Number of Cases", col = "lightblue")
barplot(deaths_count, main = "Distribution of COVID-19 Deaths by Age Group", xlab = "Age Group",
                             ylab = "Number of Deaths", col = "salmon")
```

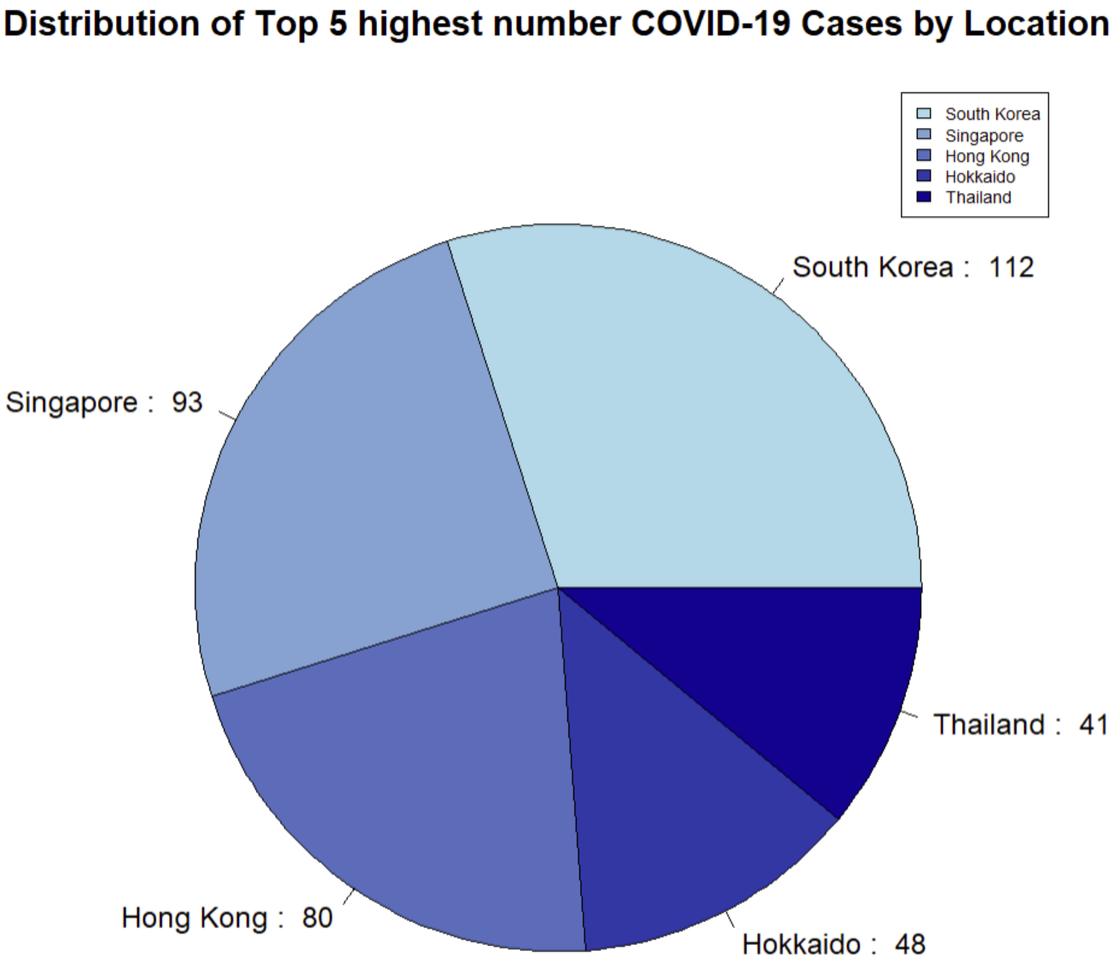




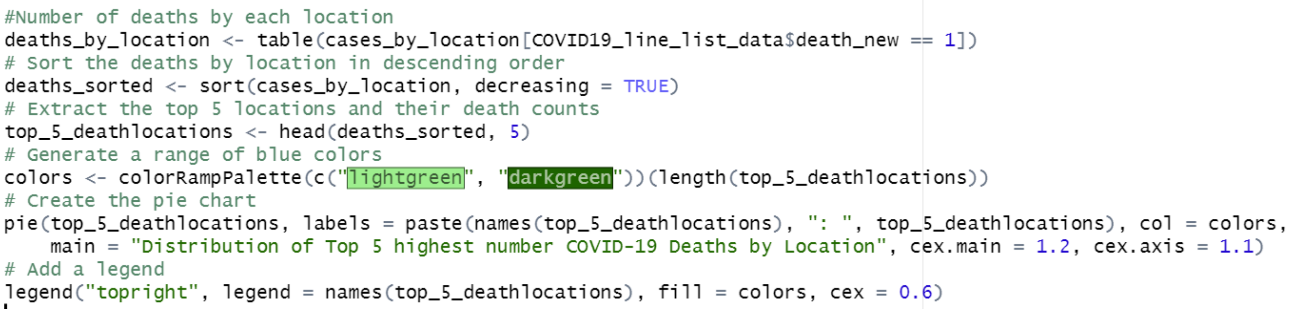
Plotting Pie-Chart to show the Top-5 most contributing Locations of COVID-19 Cases:



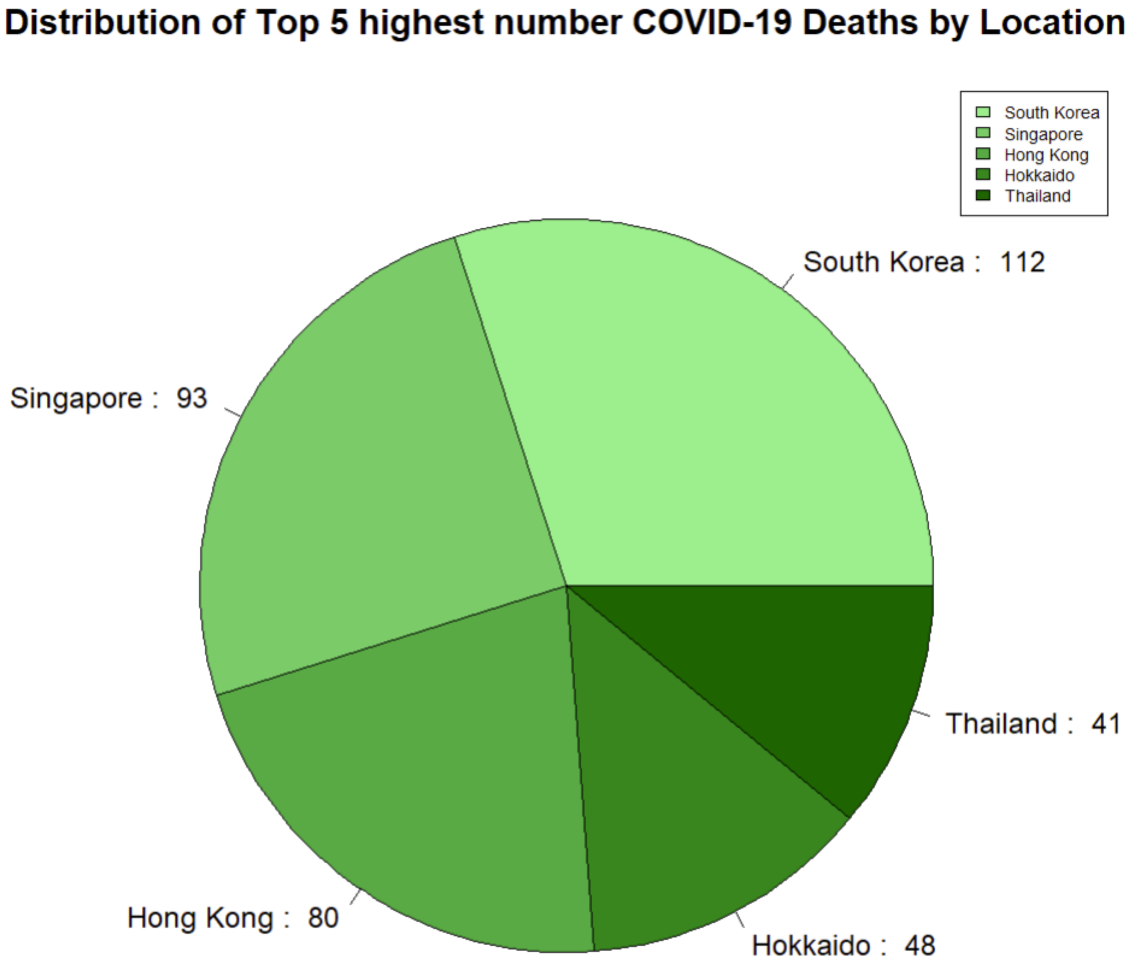
```
#Number of cases by each location
cases_by_location <- table(COVID19_line_list_data$location)

# Sort the cases by location in descending order
cases_sorted <- sort(cases_by_location, decreasing = TRUE)

# Extract the top 5 locations and their counts
top_5_locations <- head(cases_sorted, 5)

# Generate a range of blue colors
colors <- colorRampPalette(c("lightblue", "darkblue"))(length(top_10_locations))

# Set the figure size
par(mar = c(5, 5, 4, 2) + 0.1, cex.lab = 1.2)

# Create the pie chart
pie(top_5_locations, labels = paste(names(top_5_locations), ": ", top_5_locations), col = colors,
    main = "Distribution of Top 5 highest number COVID-19 Cases by Location", cex.main = 1.2, cex.axis = 1.1)


# Add a legend
legend("topright", legend = names(top_5_locations), fill = colors, cex = 0.6)
```



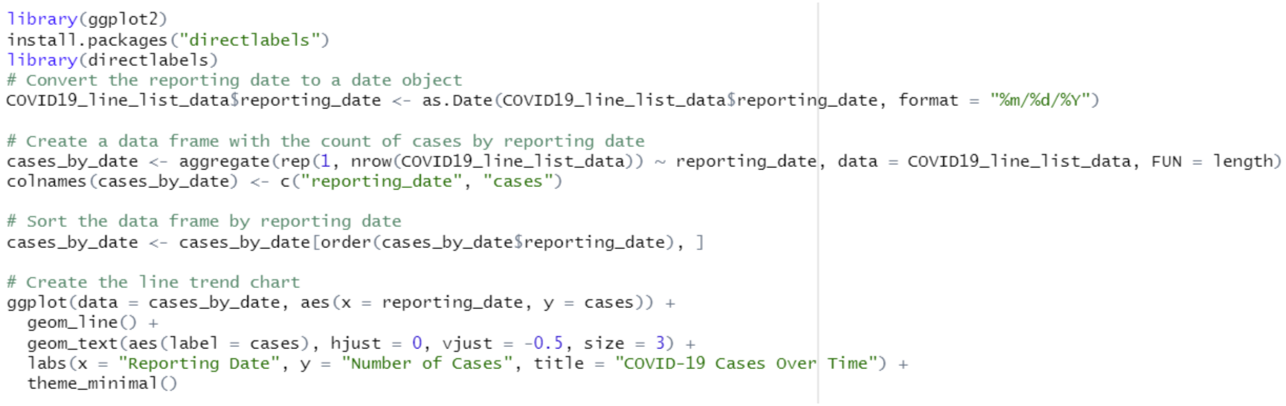
Plotting Pie-Chart to show the Top-5 most contributing Locations of COVID-19 Deaths:



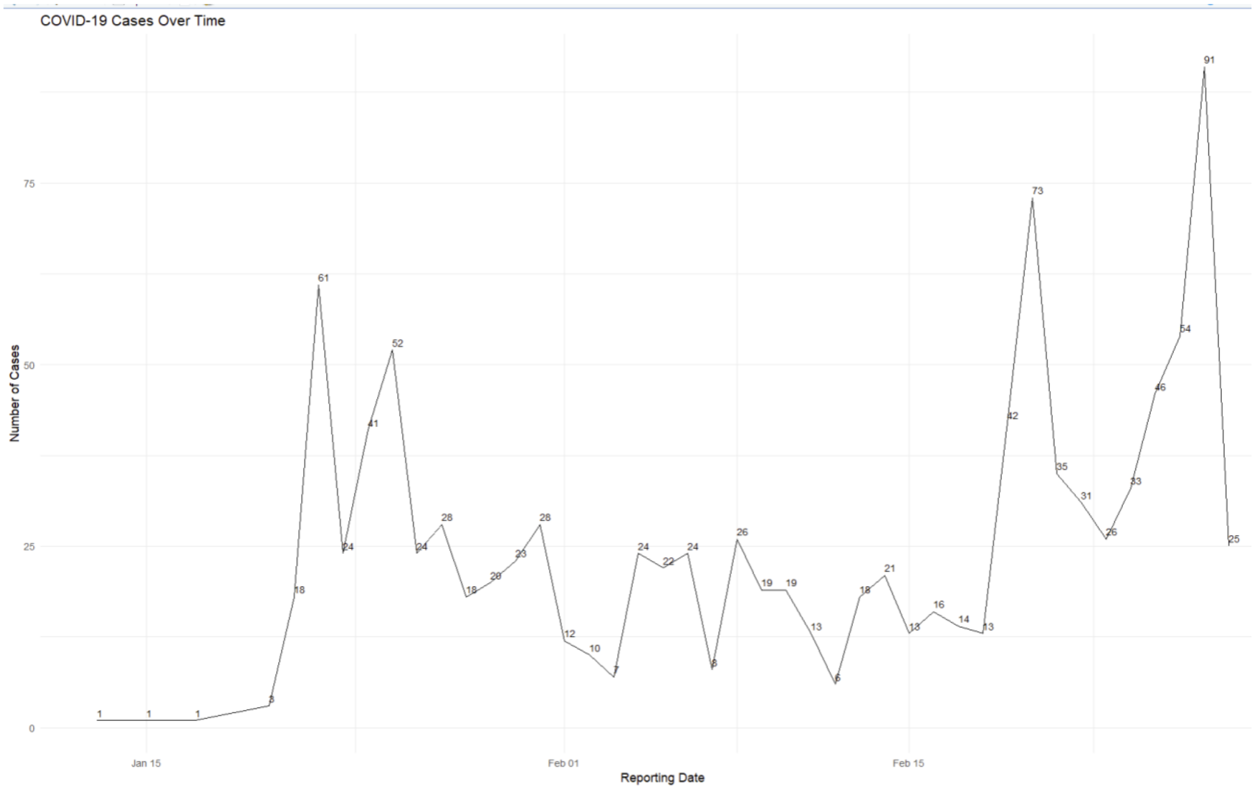
```
#Number of deaths by each location
deaths_by_location <- table(cases_by_location[COVID19_line_list_data$death_new == 1])
# Sort the deaths by location in descending order
deaths_sorted <- sort(cases_by_location, decreasing = TRUE)
# Extract the top 5 locations and their death counts
top_5_deathlocations <- head(deaths_sorted, 5)
# Generate a range of blue colors
colors <- colorRampPalette(c("lightgreen", "darkgreen"))(length(top_5_deathlocations))
# Create the pie chart
pie(top_5_deathlocations, labels = paste(names(top_5_deathlocations), ": ", top_5_deathlocations), col = colors,
    main = "Distribution of Top 5 highest number COVID-19 Deaths by Location", cex.main = 1.2, cex.axis = 1.1)
# Add a legend
legend("topright", legend = names(top_5_deathlocations), fill = colors, cex = 0.6)
```



Plotting a Line-trend chart to show the Trend of Covid-19 cases over time:



```
library(ggplot2)
install.packages("directlabels")
library(directlabels)
# Convert the reporting date to a date object
COVID19_line_list_data$reporting_date <- as.Date(COVID19_line_list_data$reporting_date, format = "%m/%d/%Y")

# Create a data frame with the count of cases by reporting date
cases_by_date <- aggregate(rep(1, nrow(COVID19_line_list_data)) ~ reporting_date, data = COVID19_line_list_data, FUN = length)
colnames(cases_by_date) <- c("reporting_date", "cases")

# Sort the data frame by reporting date
cases_by_date <- cases_by_date[order(cases_by_date$reporting_date), ]

# Create the line trend chart
ggplot(data = cases_by_date, aes(x = reporting_date, y = cases)) +
  geom_line() +
  geom_text(aes(label = cases), hjust = 0, vjust = -0.5, size = 3) +
  labs(x = "Reporting Date", y = "Number of Cases", title = "COVID-19 Cases Over Time") +
  theme_minimal()
```


COVID-19 Cases Over Time
Number of Cases
Reporting Date
Jan 15
Feb 01
Feb 15