MAT2001(Lab) Project Slot:L25+L26

Using R to analyze COVID-19 Data- An approach to use R in the real world

R B R Rudhreesh Kumaar 20BLC1103

Abstract. This semester I am taking a statistics lab class where we utilize R and statistics in order to analyze various datasets. So I thought it would be interesting to apply some of the techniques that I have learned on a COVID-19 dataset. I used this page on Kaggle and found a CSV dataset that you can get in the references below. We have data for about 1,000 cases of COVID-19. We can see the age of each infected person, the fender, whether they recovered or died, among many other things. I am further going to do a t-test to verify to common claims related to covid-19.

PROBLEM STATEMENT

1. The media claims that older people are more likely to die than younger people from COVID-19. Is this true?
2. The media claims that men are more likely to die than women from COVID-19. Is this true?

NEED AND IMPORTANCE

By analyzing a sufficiently large data (in order to remove statistical errors) we can use our existing hypothesis testing methods to know which groups of population are more vulnerable to covid-19,which groups have higher mortality rates when exposed to covid-19. After finding this we can use this information to employ more resources and time to protect these groups from covid-19 in order to minimize the total infected cases and death rate. At a crisis like this the information is crucial.

EXPERIMENT AND ANALYSIS

The general layout of this project is as follows:

Defining the problem – The first and the most critical step is to outline the questions we want to address through data analytics and the possible solutions you want to achieve at the end.

Collecting data – Data collection is a very crucial step and not as easy as it seems. The process requires time and effort. No dataset contains data as you expect it to be and involves searching, arrangements, re-arrangements, and final assembly.

Cleaning data – we want our results to be consistent, so we must ensure that data cleaning has been done correctly. In essence, data cleaning removes unnecessary and duplicate data from the collection of data.

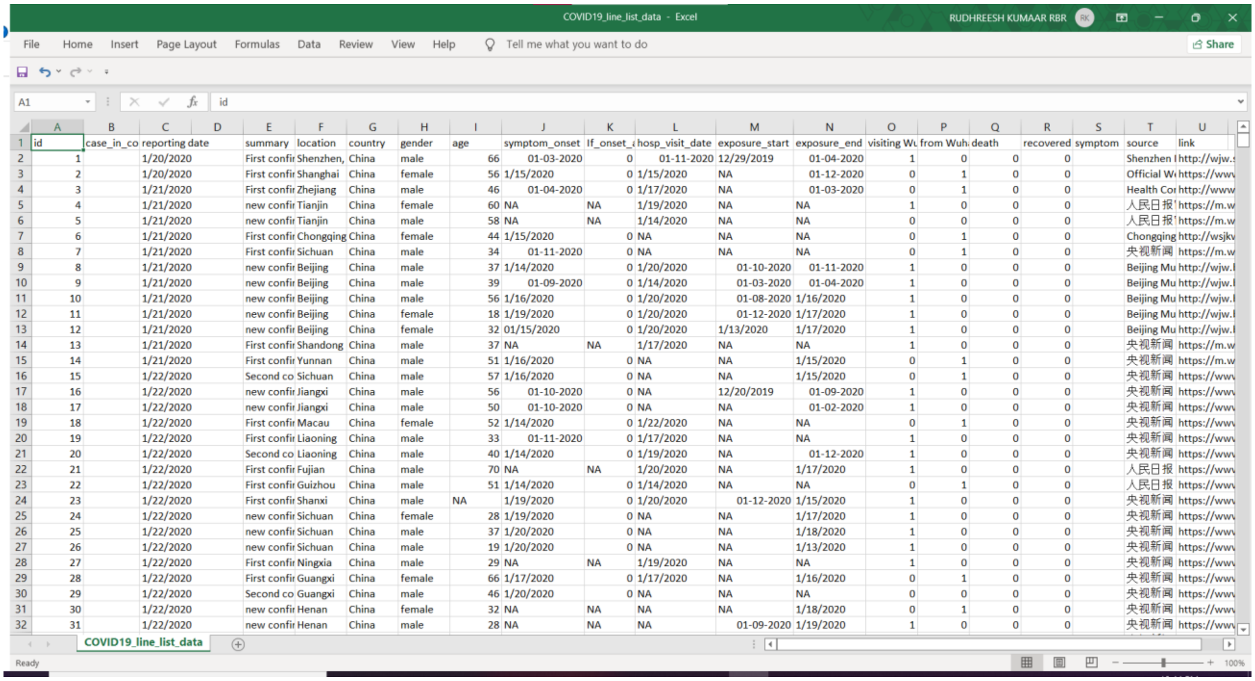
Analyzing the data – At this stage, we have to detect trends and patterns in the data collection, group them accordingly, and understand the behavior of data

Finally we can use the t.test command to gauge our confidence and to see if we can trust our means. In this case, we will use a 95% confidence interval.

Now the detail explanation is as follows:

I used this page on Kaggle and found a CSV dataset that you can get here in reference.

Let us take a look at the data:

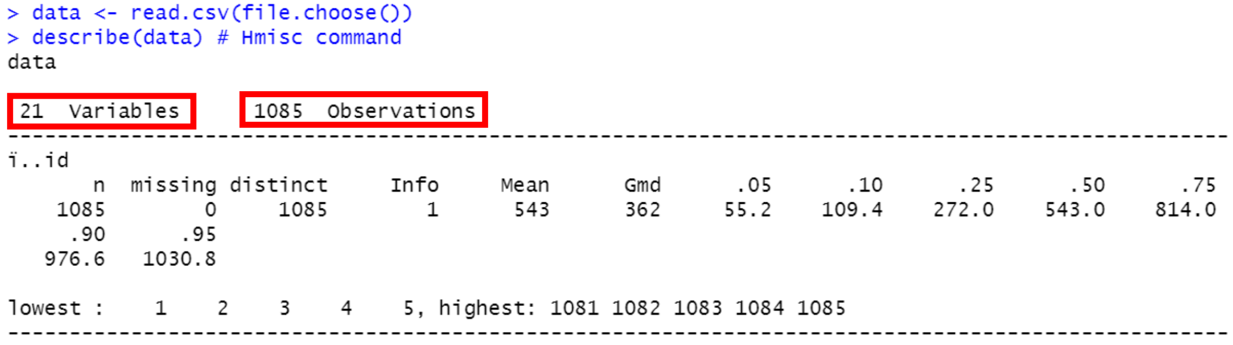


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	id	case_in_co	reporting date		summary	location	country	gender	age	symptom_onset	If_onset_a	hosp_visit_date	exposure_start	exposure_end	visiting Wu	from Wuh	death	recovered	symptom	source	link
2	1		1/20/2020		First confir	Shenzhen,	China	male	66	01-03-2020	0	01-11-2020	12/29/2019	01-04-2020	1	0	0	0		Shenzhen I	http://wjw.s
3	2		1/20/2020		First confir	Shanghai	China	female	56	1/15/2020	0	1/15/2020	NA	01-12-2020	0	1	0	0		Official W	https://www
4	3		1/21/2020		First confir	Zhejiang	China	male	46	01-04-2020	0	1/17/2020	NA	01-03-2020	0	1	0	0		Health Cor	http://www
5	4		1/21/2020		new confir	Tianjin	China	female	60	NA	NA	1/19/2020	NA	NA	1	0	0	0		人民日报	https://m.w
6	5		1/21/2020		new confir	Tianjin	China	male	58	NA	NA	1/14/2020	NA	NA	0	0	0	0		人民日报	https://m.w
7	6		1/21/2020		First confir	Chongqing	China	female	44	1/15/2020	0	NA	NA	NA	0	1	0	0		Chongqing	http://wsjkw
8	7		1/21/2020		First confir	Sichuan	China	male	34	01-11-2020	0	NA	NA	NA	0	1	0	0		央视新闻	https://m.w
9	8		1/21/2020		new confir	Beijing	China	male	37	1/14/2020	0	1/20/2020	01-10-2020	01-11-2020	1	0	0	0		Beijing Mu	http://wjw.l
10	9		1/21/2020		new confir	Beijing	China	male	39	01-09-2020	0	1/14/2020	01-03-2020	01-04-2020	1	0	0	0		Beijing Mu	http://wjw.l
11	10		1/21/2020		new confir	Beijing	China	male	56	1/16/2020	0	1/20/2020	01-08-2020	1/16/2020	1	0	0	0		Beijing Mu	http://wjw.l
12	11		1/21/2020		new confir	Beijing	China	female	18	1/19/2020	0	1/20/2020	01-12-2020	1/17/2020	1	0	0	0		Beijing Mu	http://wjw.l
13	12		1/21/2020		new confir	Beijing	China	female	32	01/15/2020	0	1/20/2020	1/13/2020	1/17/2020	1	0	0	0		Beijing Mu	http://wjw.l
14	13		1/21/2020		First confir	Shandong	China	male	37	NA	NA	1/17/2020	NA	NA	1	0	0	0		央视新闻	https://m.w
15	14		1/21/2020		First confir	Yunnan	China	male	51	1/16/2020	0	NA	NA	1/15/2020	0	1	0	0		央视新闻	https://m.w
16	15		1/22/2020		Second co	Sichuan	China	male	57	1/16/2020	0	NA	NA	1/15/2020	0	1	0	0		央视新闻	https://www
17	16		1/22/2020		new confir	Jiangxi	China	male	56	01-10-2020	0	NA	12/20/2019	01-09-2020	1	0	0	0		央视新闻	https://www
18	17		1/22/2020		new confir	Jiangxi	China	male	50	01-10-2020	0	NA	NA	01-02-2020	1	0	0	0		央视新闻	https://www
19	18		1/22/2020		First confir	Macau	China	female	52	1/14/2020	0	1/22/2020	NA	NA	0	1	0	0		央视新闻	https://www
20	19		1/22/2020		First confir	Liaoning	China	male	33	01-11-2020	0	1/17/2020	NA	NA	1	0	0	0		央视新闻	https://www
21	20		1/22/2020		Second co	Liaoning	China	male	40	1/14/2020	0	1/19/2020	NA	01-12-2020	1	0	0	0		央视新闻	https://www
22	21		1/22/2020		First confir	Fujian	China	male	70	NA	NA	1/20/2020	NA	1/17/2020	1	0	0	0		人民日报	https://www
23	22		1/22/2020		First confir	Guizhou	China	male	51	1/14/2020	0	1/14/2020	NA	NA	0	1	0	0		人民日报	https://www
24	23		1/22/2020		First confir	Shanxi	China	male	NA	1/19/2020	0	1/20/2020	01-12-2020	1/15/2020	1	0	0	0		央视新闻	https://www
25	24		1/22/2020		new confir	Sichuan	China	female	28	1/19/2020	0	NA	NA	1/17/2020	1	0	0	0		央视新闻	https://www
26	25		1/22/2020		new confir	Sichuan	China	male	37	1/20/2020	0	NA	NA	1/18/2020	1	0	0	0		央视新闻	https://www
27	26		1/22/2020		new confir	Sichuan	China	male	19	1/20/2020	0	NA	NA	1/13/2020	1	0	0	0		央视新闻	https://www
28	27		1/22/2020		First confir	Ningxia	China	male	29	NA	NA	1/19/2020	NA	NA	1	0	0	0		央视新闻	https://www
29	28		1/22/2020		First confir	Guangxi	China	female	66	1/17/2020	0	1/17/2020	NA	1/16/2020	0	1	0	0		央视新闻	https://www
30	29		1/22/2020		Second co	Guangxi	China	male	46	1/20/2020	0	NA	NA	NA	0	0	0	0		央视新闻	https://www
31	30		1/22/2020		new confir	Henan	China	female	32	NA	NA	NA	NA	1/18/2020	0	1	0	0		央视新闻	https://www
32	31		1/22/2020		new confir	Henan	China	male	28	NA	NA	NA	01-09-2020	1/19/2020	1	0	0	0		央视新闻	https://www

We have data for about 1,000 cases of COVID-19. We can see the age of each infected person, the fender, whether they recovered or died, among many other things. In RStudio, to import the dataset, we will go to File->Import Dataset->From Text(base). We will copy this line into our main R script, which I will save as code.R in the same folder as our CSV file. For convenience, I will rename the data frame variable to “data.” It is usually a good practice to start your R scripts with this line which just removes all of the data and all the variables you've previously loaded I will also clear all existing variables, import a library called Hmisc, and use it’s describe function to better understand our data.

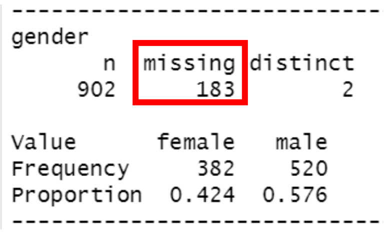
we can use a describe command that we imported from Hmisc so we'll just say describe data and we will run this so after running this we'll have a lot of output in our console but if we scroll up we can see some informative things about our data.

- For instance we have 21 columns and 1085 observations.



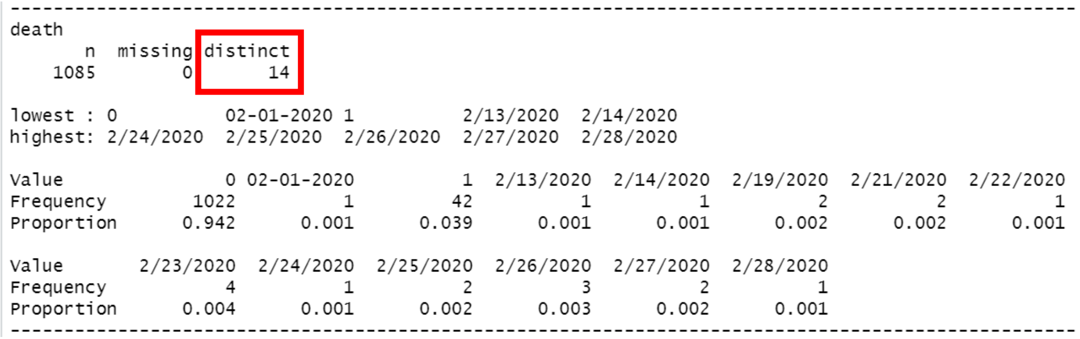
```
> data <- read.csv(file.choose())
> describe(data) # Hmisc command
data

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

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

- if we scroll further down and we see that 183 entries are missing gender but the others aren't.



```
----------------------------
gender
       n  missing distinct
     902      183        2

Value      female   male
Frequency     382    520
Proportion  0.424  0.576
----------------------------
```

- If we look at the death section, we see that there are 14 distinct values.



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

lowest : 0          02-01-2020 1          2/13/2020  2/14/2020
highest: 2/24/2020  2/25/2020  2/26/2020  2/27/2020  2/28/2020

Value               0 02-01-2020          1  2/13/2020  2/14/2020  2/19/2020  2/21/2020  2/22/2020
Frequency        1022          1         42          1          1          2          2          1
Proportion      0.942      0.001      0.039      0.001      0.001      0.002      0.002      0.001

Value       2/23/2020  2/24/2020  2/25/2020  2/26/2020  2/27/2020  2/28/2020
Frequency           4          1          2          3          2          1
Proportion      0.004      0.001      0.002      0.003      0.002      0.001
---------------------------------------------------------------------------------------------------
```

This may seem a little bit weird, but the death column either has a 0 (no death), 1 (no death), or simply the date of the patient’s death. This is difficult to work with because we want all 0s and 1s. We fix this by adding a death_dummy column to our dataset, which only contains the values 0 and 1. We also calculated the death rate of our dataset, which after running turns out to be 5.8%. For the first part, we will analyze the age of the people who have died and of those who did not. Then we use this to proceed to the next step .

FINDINGS AND RESULTS

AGE

The media claims that older people are more likely to die than younger people from COVID-19. Is this true? Let us check with our dataset. First, we will subset our dataset into patients who are alive and patients who have died and compare the mean ages. This code will do it for us:

Code:

```
# AGE

dead = subset(data, death_dummy == 1)

alive = subset(data, death_dummy == 0)

mean(dead$age, na.rm=TRUE)

mean(alive$age, na.rm=TRUE)
```

Notice, the na.rm=TRUE means to skip rows that have NA (or no value) for a specific column (age in this case). After running this, we get that the mean of those who survived is 48 years, while the mean for those who have died is about 68.6 years old.

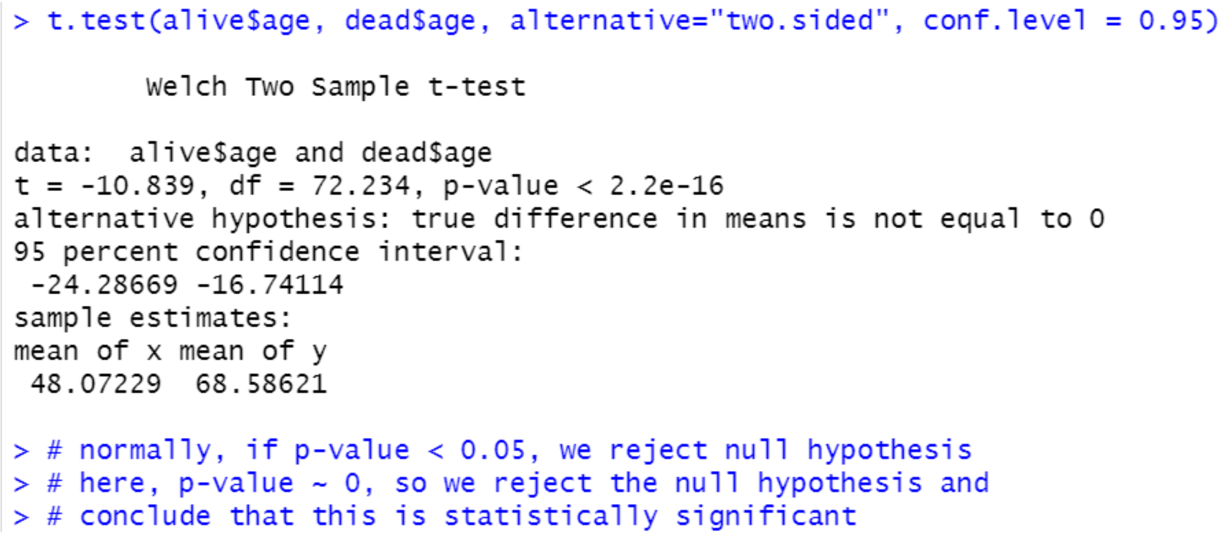
Okay, so the data does indeed show that those who die are older in our sample. But is this true universally for the population? How confident are we that this is true?We can use the t.test command to gauge our confidence and to see if we can trust our means. In this case, we will use a 95% confidence interval. This code will do it for us:

Code:

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

This simple command is quite powerful. Notice, we gave it both the ages of the alive patients as well as the age of the patients who have died. Let’s analyze the output.

OUTPUT:



```
> t.test(alive$age, dead$age, alternative="two.sided", 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 

> # normally, if p-value < 0.05, we reject null hypothesis
> # here, p-value ~ 0, so we reject the null hypothesis and
> # conclude that this is statistically significant
```

Looking at the confidence interval, we can say with 95% certainty, that the age difference between patients who have died and those who haven't is from 16.7 to 24.3 years. Now, look at the p-value. It is almost 0. This means that there is ~0% chance that we obtained such extreme result randomly from this sample under the null hypothesis (which is that the ages of the two groups are equal). For this reason, we can reasonably reject the null hypothesis (under the conventional significance level of 0.05) and say that people who have died from COVID-19 are indeed older than those who did not. Now, let's look at gender!

GENDER

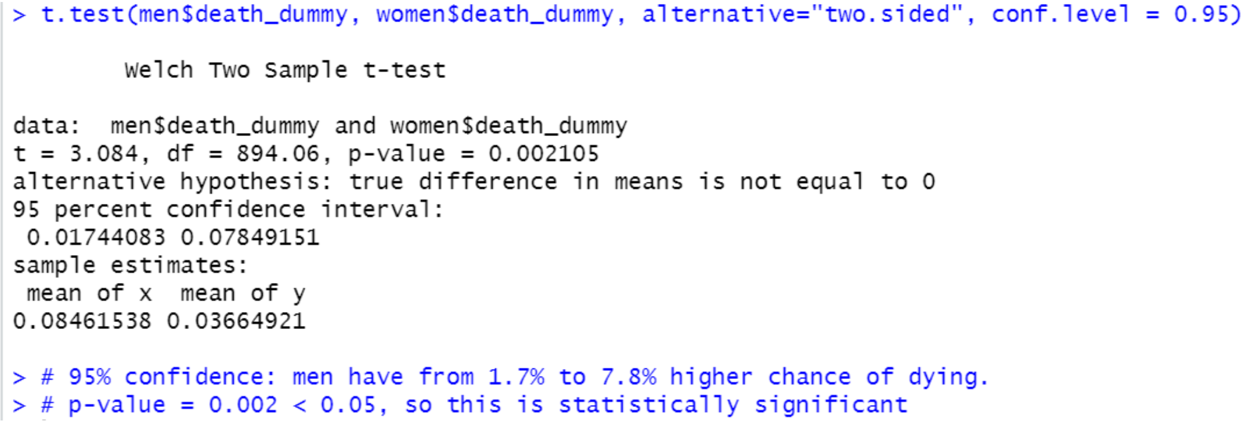
This will be very similar. We want to see if the death rate is similar for men and women. Let us again split our data and perform the t-test. This code will do it for us:

Code:

```
# GENDER

men = subset(data, gender == "male")

women = subset(data, gender == "female")

mean(men$death_dummy, na.rm=TRUE)

mean(women$death_dummy, na.rm=TRUE)

t.test(men$death_dummy, women$death_dummy, alternative="two.sided", conf.level = 0.95)
```

We subset our original data into two sets. After calculating the means, we see that men in this dataset have a death rate of 8.5% as opposed to 3.7% in women. Well, this is unexpected. Again, can we trust this data? Here is the t.test output:

OUTPUT:



```
> t.test(men$death_dummy, women$death_dummy, alternative="two.sided", conf.level = 0.95)

        Welch Two Sample t-test

data:  men$death_dummy and women$death_dummy
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:
 0.01744083 0.07849151
sample estimates:
 mean of x  mean of y 
0.08461538 0.03664921 

> # 95% confidence: men have from 1.7% to 7.8% higher chance of dying.
> # p-value = 0.002 < 0.05, so this is statistically significant
```

Our confidence interval of 95% shows that mean will have from 1.7% to 7.8% higher death rate than women. A p-value of 0.002 signifies that we can reject the null hypothesis that men and women have the same death rate, since 0.002 < 0.05.). For this reason, we can reasonably reject the null hypothesis (under the conventional significance level of 0.05) and say that men are more likely than to die from COVID-19 .

CODE

```
rm(list=ls())
#this removes all variables stored previously
library(Hmisc)
# import the files

data <- read.csv("C:/Users/intel-vsc/Desktop/vit/r studio/COVID19_line_list_data.csv")

describe(data) # Hmisc command
# cleaned up death column

data$death_dummy <- as.integer(data$death != 0)
# death rate
sum(data$death_dummy) / nrow(data)

# AGE
# claim: people who die are older
dead = subset(data, death_dummy == 1)
alive = subset(data, death_dummy == 0)
mean(dead$age, na.rm = TRUE)
mean(alive$age, na.rm = TRUE)
# is this statistically significant?
t.test(alive$age, dead$age, alternative="two.sided", conf.level = 0.95)
# normally, if p-value < 0.05, we reject null hypothesis
# here, p-value ~ 0,
#so we reject the null hypothesis and
# conclude that this is statistically significant
# GENDER

# claim: gender has no effect

men = subset(data, gender == "male")

women = subset(data, gender == "female")

mean(men$death_dummy, na.rm = TRUE) #8.5%!

mean(women$death_dummy, na.rm = TRUE) #3.7%

# is this statistically significant?

t.test(men$death_dummy, women$death_dummy, alternative="two.sided", conf.level = 0.95)

# 99% confidence: men have from 0.01% to 7.8% higher chance

# of dying.

# p-value = 0.002 < 0.05, so this is statistically

# significant
```

CONCLUSION

As you can see, R helps us perform statistical analysis on important datasets quite easily. Thank you for reading!

References: data set: https://www.kaggle.com/sudalairajkumar/novel-corona-virus-2019-dataset/version/25?select=COVID19_line_list_data.csv

APPENDIX

Link of my code file: MAT2001_Project.R