Master of Data Science (Semester 1 – 2020/2021)

Faculty of Computer Science & Information Technology

WQD7006 MACHINE LEARNING FOR DATA SCIENCE

Assignment Title:

Coronaviruses (COVID-19) Prediction based on Symptoms

Student Details:

Mun Siu Hou

17218313

17218313@siswa.um.edu.my

TABLE OF CONTENTS

1.0 Introduction

Coronavirus disease (COVID-19) is an infectious disease caused by a newly discovered coronavirus. Most people infected with the COVID-19 virus will experience mild to moderate respiratory illness and recover without requiring special treatment. Older people, and those with underlying medical problems like cardiovascular disease, diabetes, chronic respiratory disease, and cancer are more likely to develop serious illness. The best way to prevent and slow down transmission is be well informed about the COVID-19 virus, the disease it causes and how it spreads. Protect yourself and others from infection by washing your hands or using an alcohol-based rub frequently and not touching your face. The COVID-19 virus spreads primarily through droplets of saliva or discharge from the nose when an infected person coughs or sneezes, so it is important that you also practice respiratory etiquette (for example, by coughing into a flexed elbow). At this time, there are no specific vaccines or treatments for COVID-19. Therefore, it is a need to provide a symptoms self-checker system for us to determine whether we are infected with this disease or not. This self-checker system not only allow the user to do a pre-testing but also helping the health care workers to reduce their workload.

2.0 Objective

1. To provide help for the user to check whether is infecting with the coronavirus disease or not.
2. To help to reduce the workload of the heath care workers where those medical supplies are limited.

3.0 Methodology

There will be several stages for this assignment such as data preparation, data cleaning, data visualization and etc. Each stage will be discussed in each section. Figure 1 showed the general flow of this project.

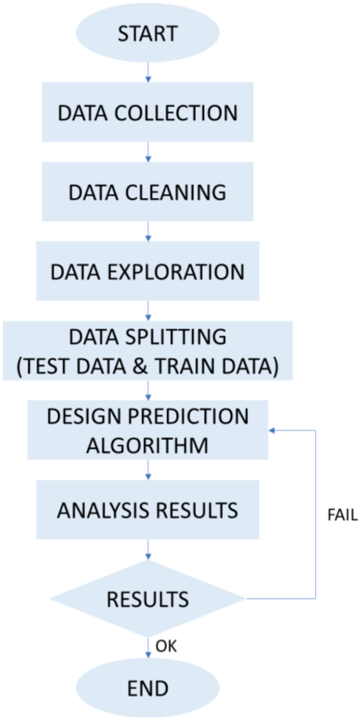


Figure 1: General flow of the Project

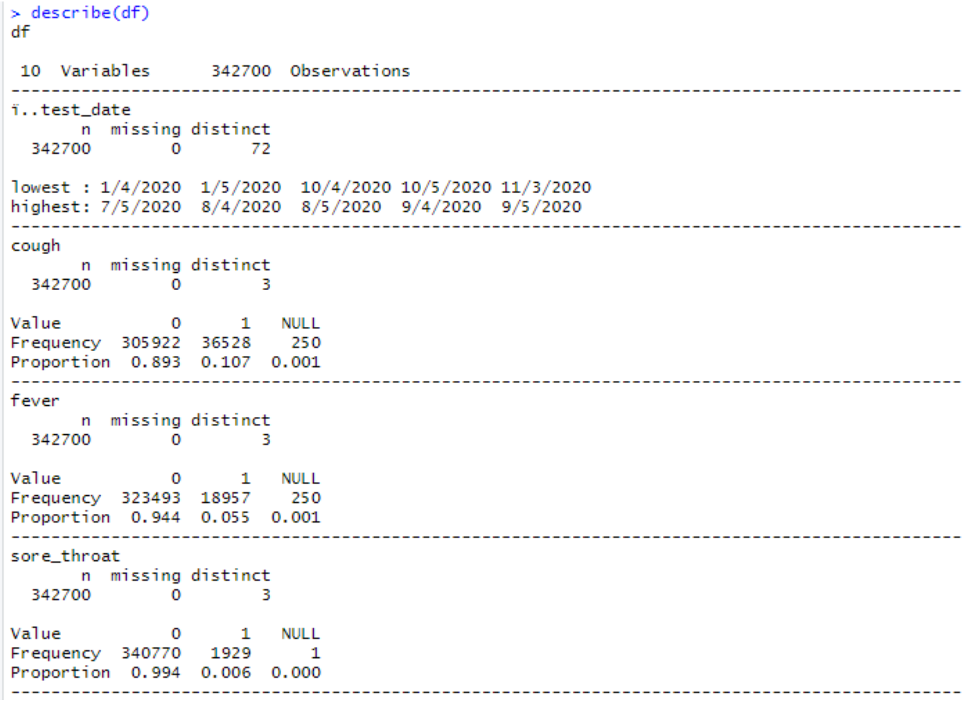
First, this assignment starts with data collection. The data that this project looking for is patient data with symptoms and detecting results. Next, this assignment will continue with data cleaning, data exploration and data splitting. After that, will research about suitable algorithms to do prediction. Next, will go through training and testing process and then only go for analysis on results. If the results are not as expecting, will redesign a better algorithm.

3.1 Data Collection Stage

First, this assignment will be using the data set from the website. This website is from open data from government Israel. Therefore, it required to translate the language before we access the website. Because of the privacy issue, a lot of hospital will not be sharing their patient information and symptoms details.

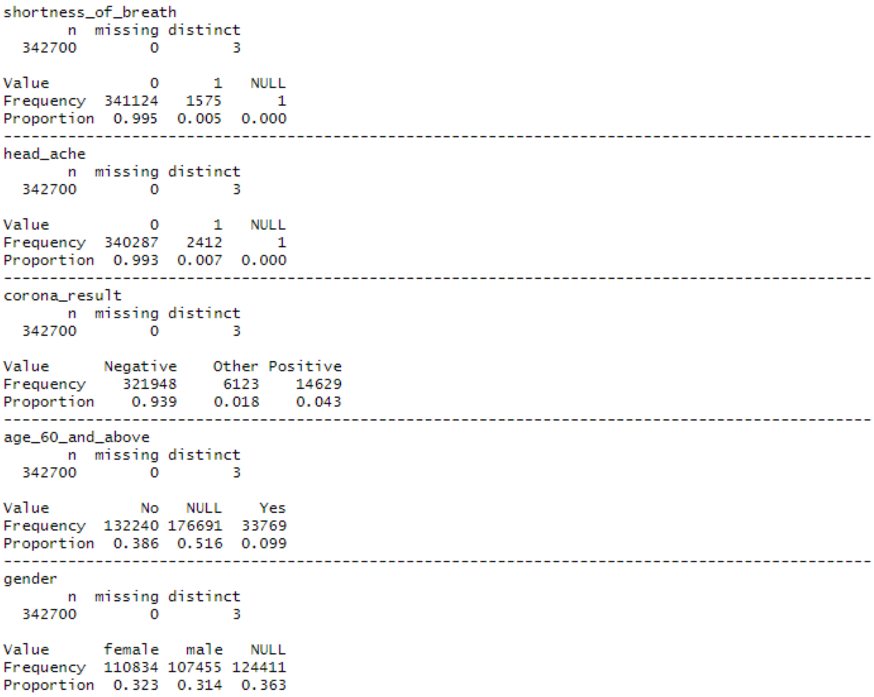
3.2 Data Cleaning Stage

After downloading the dataset from the website, this project will need to do the data cleaning process to make sure the data are ready to be used. Before the data cleaning process, this project also has to make sure that what type of data will be using later. Therefore, at first, this project will check the data shape, structure, and their features. Next, this project will go through each column to check for the missing data. If the missing data is only a very small portion of the whole dataset, this project will directly remove it. Besides for the symptoms of the patients, this dataset also included the activities details for the patients whereby it also recorded the patient’s records on 14 days traveling history and get contact history.



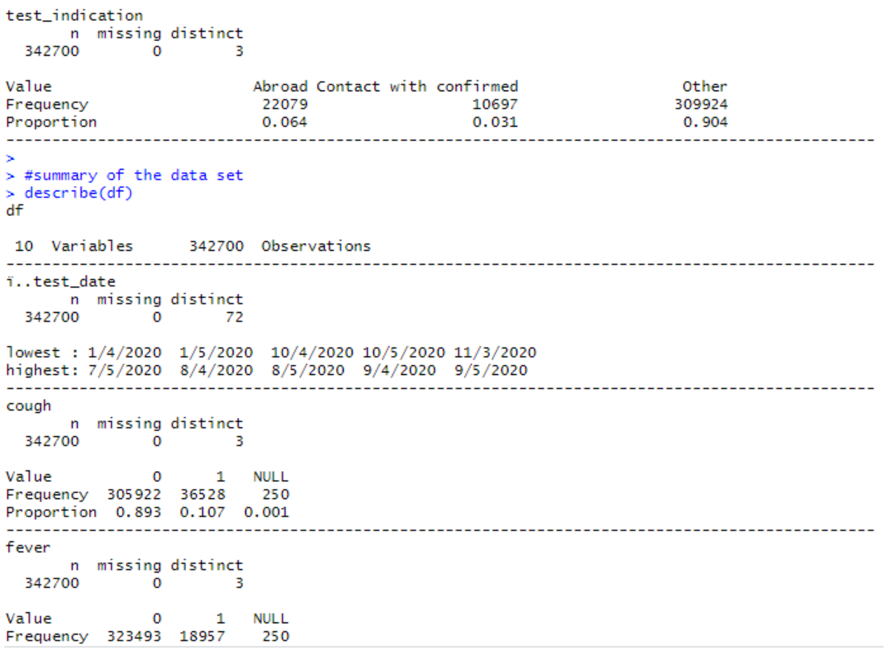
```
> describe(df)
df

 10  Variables      342700  Observations
--------------------------------------------------------------------------------------------------
ï..test_date
       n  missing distinct
  342700        0       72

lowest : 1/4/2020  1/5/2020  10/4/2020 10/5/2020 11/3/2020
highest: 7/5/2020  8/4/2020  8/5/2020  9/4/2020  9/5/2020
--------------------------------------------------------------------------------------------------
cough
       n  missing distinct
  342700        0        3

Value           0      1    NULL
Frequency  305922  36528    250
Proportion  0.893  0.107  0.001
--------------------------------------------------------------------------------------------------
fever
       n  missing distinct
  342700        0        3

Value           0      1    NULL
Frequency  323493  18957    250
Proportion  0.944  0.055  0.001
--------------------------------------------------------------------------------------------------
sore_throat
       n  missing distinct
  342700        0        3

Value           0      1    NULL
Frequency  340770   1929      1
Proportion  0.994  0.006  0.000
--------------------------------------------------------------------------------------------------
```

Figure 2: Result for data description



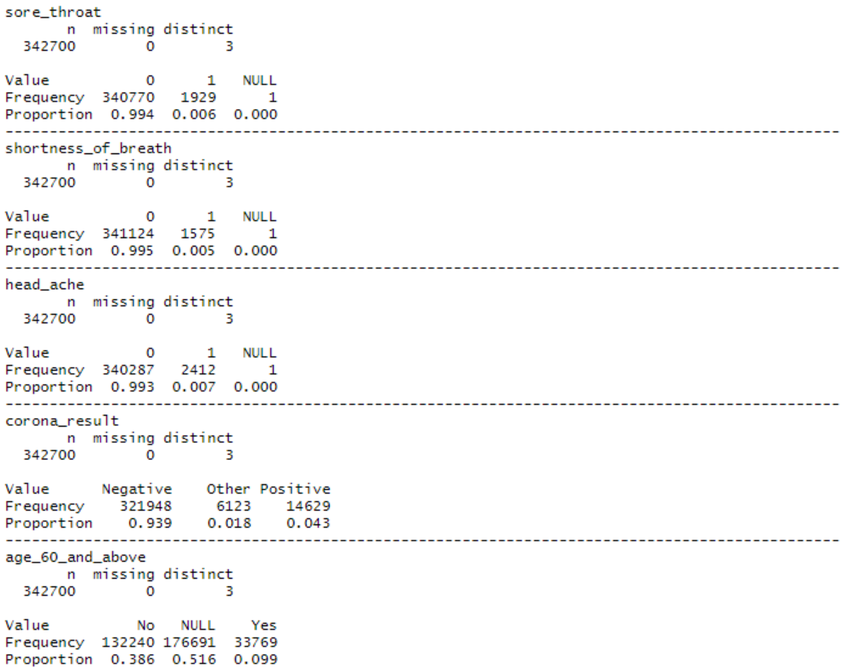
```
shortness_of_breath
       n  missing distinct
  342700        0        3

Value           0      1    NULL
Frequency  341124   1575      1
Proportion  0.995  0.005  0.000
-----------------------------------------------------------------------------------------------
head_ache
       n  missing distinct
  342700        0        3

Value           0      1    NULL
Frequency  340287   2412      1
Proportion  0.993  0.007  0.000
-----------------------------------------------------------------------------------------------
corona_result
       n  missing distinct
  342700        0        3

Value      Negative    Other Positive
Frequency    321948     6123    14629
Proportion    0.939    0.018    0.043
-----------------------------------------------------------------------------------------------
age_60_and_above
       n  missing distinct
  342700        0        3

Value          No   NULL    Yes
Frequency  132240 176691  33769
Proportion  0.386  0.516  0.099
-----------------------------------------------------------------------------------------------
gender
       n  missing distinct
  342700        0        3

Value      female   male   NULL
Frequency  110834 107455 124411
Proportion  0.323  0.314  0.363
```

Figure 3: Result for data description



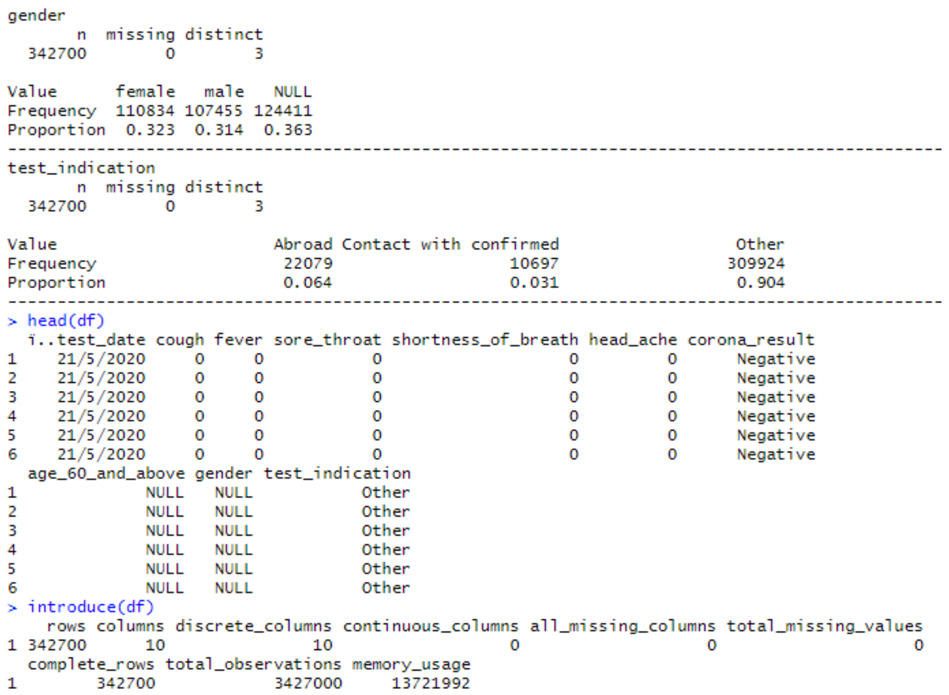
```
test_indication
       n  missing distinct
  342700        0        3

Value                        Abroad Contact with confirmed                    Other
Frequency                     22079                10697                     309924
Proportion                    0.064                0.031                      0.904
------------------------------------------------------------------------------------------
>
> #summary of the data set
> describe(df)
df

 10  Variables      342700  Observations
------------------------------------------------------------------------------------------
i..test_date
       n  missing distinct
  342700        0       72

lowest : 1/4/2020  1/5/2020  10/4/2020 10/5/2020 11/3/2020
highest: 7/5/2020  8/4/2020  8/5/2020  9/4/2020  9/5/2020
------------------------------------------------------------------------------------------
cough
       n  missing distinct
  342700        0        3

Value           0      1   NULL
Frequency  305922  36528    250
Proportion  0.893  0.107  0.001
------------------------------------------------------------------------------------------
fever
       n  missing distinct
  342700        0        3

Value           0      1   NULL
Frequency  323493  18957    250
```

Figure 4: Result for data description



```
sore_throat
       n  missing distinct
  342700        0        3

Value           0      1   NULL
Frequency  340770   1929      1
Proportion  0.994  0.006  0.000
------------------------------------------------------------------------------------------
shortness_of_breath
       n  missing distinct
  342700        0        3

Value           0      1   NULL
Frequency  341124   1575      1
Proportion  0.995  0.005  0.000
------------------------------------------------------------------------------------------
head_ache
       n  missing distinct
  342700        0        3

Value           0      1   NULL
Frequency  340287   2412      1
Proportion  0.993  0.007  0.000
------------------------------------------------------------------------------------------
corona_result
       n  missing distinct
  342700        0        3

Value      Negative    Other Positive
Frequency    321948     6123    14629
Proportion    0.939    0.018    0.043
------------------------------------------------------------------------------------------
age_60_and_above
       n  missing distinct
  342700        0        3

Value          No   NULL    Yes
Frequency  132240 176691  33769
Proportion  0.386  0.516  0.099
```

Figure 5: Result for data description



```
gender
       n  missing distinct
  342700        0        3

Value      female   male   NULL
Frequency  110834 107455 124411
Proportion  0.323  0.314  0.363
-----------------------------------------------------------------------------------------------
test_indication
       n  missing distinct
  342700        0        3

Value                      Abroad Contact with confirmed                  Other
Frequency                   22079                  10697                 309924
Proportion                  0.064                  0.031                  0.904
-----------------------------------------------------------------------------------------------
> head(df)
  i..test_date cough fever sore_throat shortness_of_breath head_ache corona_result
1    21/5/2020     0     0           0                   0         0      Negative
2    21/5/2020     0     0           0                   0         0      Negative
3    21/5/2020     0     0           0                   0         0      Negative
4    21/5/2020     0     0           0                   0         0      Negative
5    21/5/2020     0     0           0                   0         0      Negative
6    21/5/2020     0     0           0                   0         0      Negative
  age_60_and_above gender test_indication
1             NULL   NULL           Other
2             NULL   NULL           Other
3             NULL   NULL           Other
4             NULL   NULL           Other
5             NULL   NULL           Other
6             NULL   NULL           Other
> introduce(df)
    rows columns discrete_columns continuous_columns all_missing_columns total_missing_values
1 342700      10               10                  0                   0                    0
  complete_rows total_observations memory_usage
1        342700            3427000     13721992
```

Figure 6: Result for data description

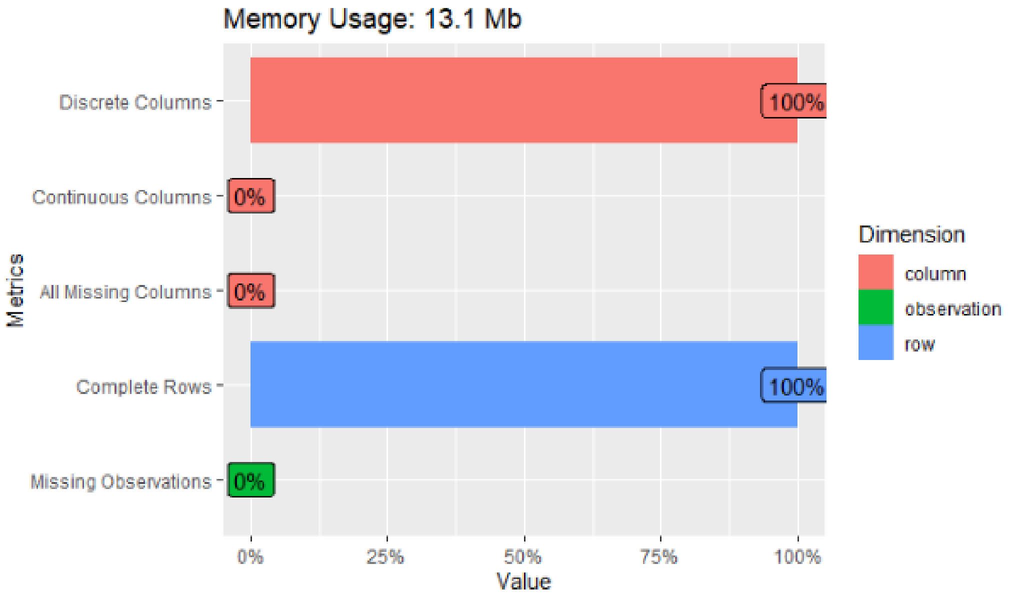
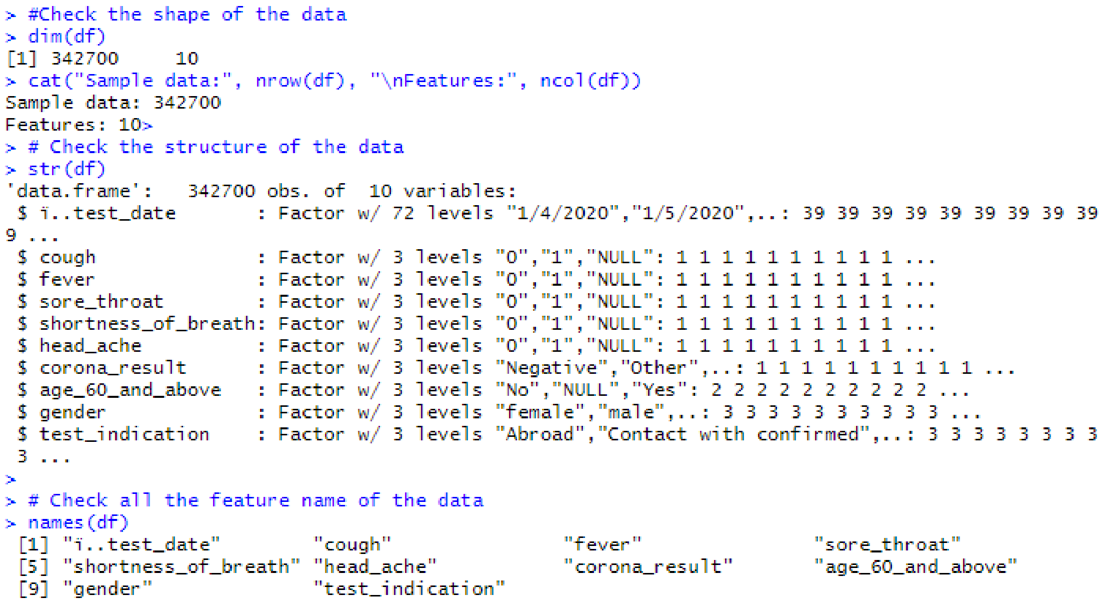


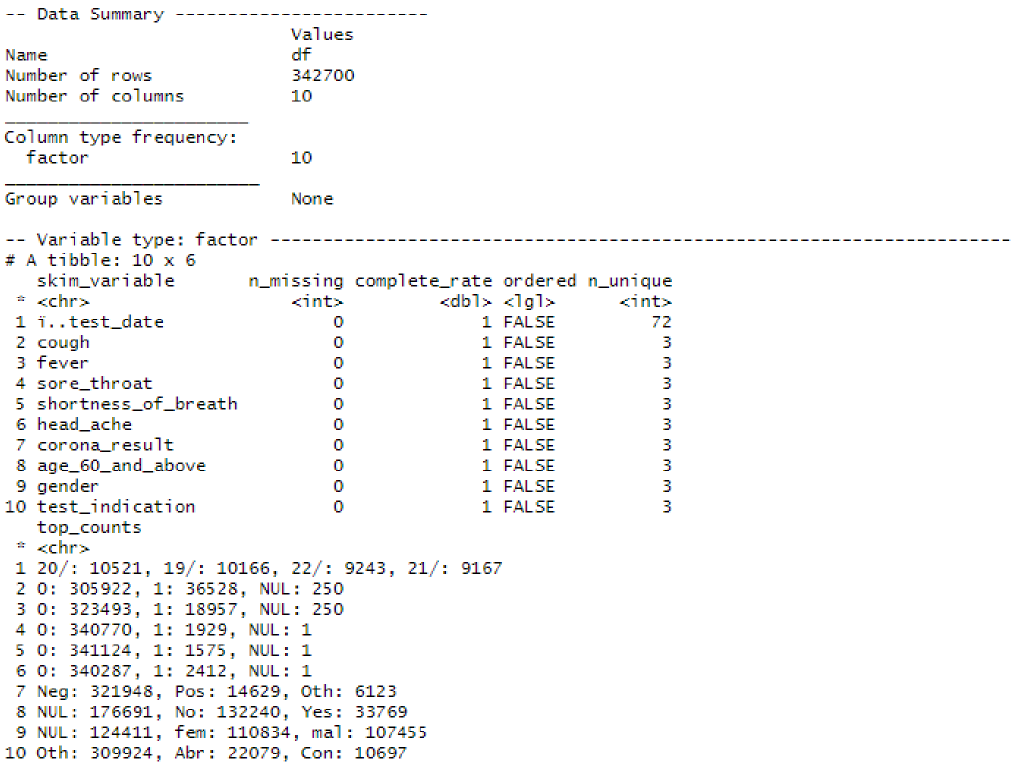
Figure 7: Plot of the dataset

This dataset does not have any missing column because all the missing columns are considered as NULL value.



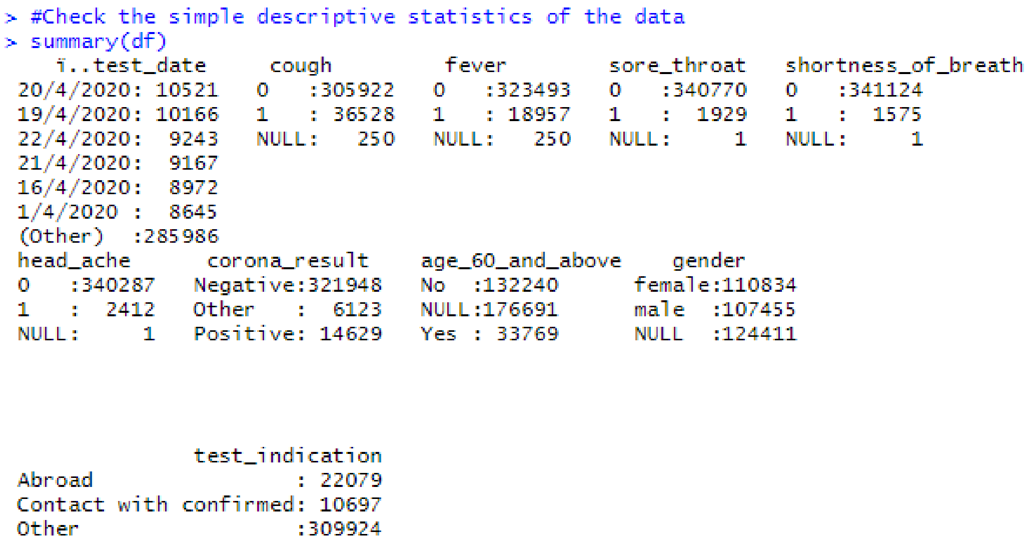
```
> #Check the shape of the data
> dim(df)
[1] 342700     10
> cat("Sample data:", nrow(df), "\nFeatures:", ncol(df))
Sample data: 342700
Features: 10>
> # Check the structure of the data
> str(df)
'data.frame':   342700 obs. of  10 variables:
 $ i..test_date        : Factor w/ 72 levels "1/4/2020","1/5/2020",..: 39 39 39 39 39 39 39 39 39
9 ...
 $ cough               : Factor w/ 3 levels "0","1","NULL": 1 1 1 1 1 1 1 1 1 1 ...
 $ fever               : Factor w/ 3 levels "0","1","NULL": 1 1 1 1 1 1 1 1 1 1 ...
 $ sore_throat         : Factor w/ 3 levels "0","1","NULL": 1 1 1 1 1 1 1 1 1 1 ...
 $ shortness_of_breath: Factor w/ 3 levels "0","1","NULL": 1 1 1 1 1 1 1 1 1 1 ...
 $ head_ache           : Factor w/ 3 levels "0","1","NULL": 1 1 1 1 1 1 1 1 1 1 ...
 $ corona_result       : Factor w/ 3 levels "Negative","Other",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ age_60_and_above    : Factor w/ 3 levels "No","NULL","Yes": 2 2 2 2 2 2 2 2 2 2 ...
 $ gender              : Factor w/ 3 levels "female","male",..: 3 3 3 3 3 3 3 3 3 3 ...
 $ test_indication     : Factor w/ 3 levels "Abroad","Contact with confirmed",..: 3 3 3 3 3 3 3 3
3 ...
>
> # Check all the feature name of the data
> names(df)
 [1] "i..test_date"        "cough"               "fever"               "sore_throat"
 [5] "shortness_of_breath" "head_ache"           "corona_result"       "age_60_and_above"
 [9] "gender"              "test_indication"
```

Figure 8: Check the structure of the dataset



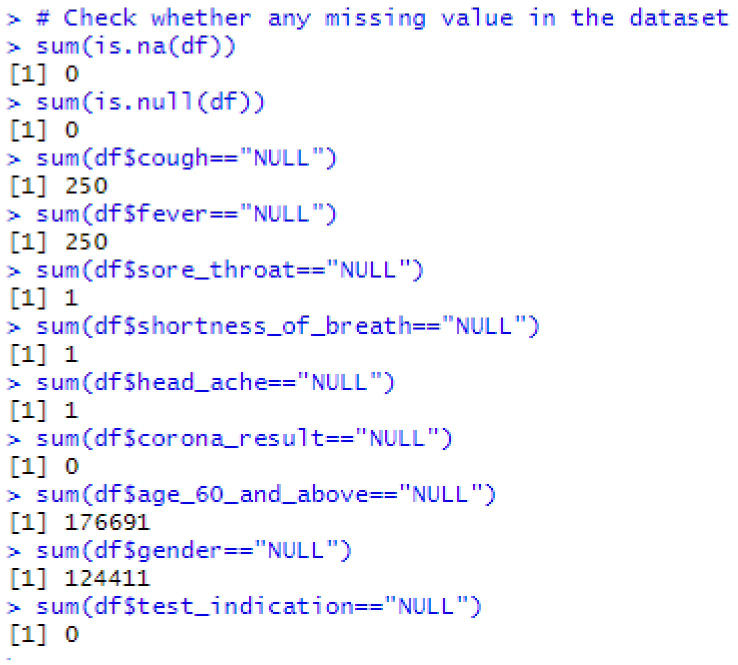
```
-- Data Summary ------------------------
                           Values
Name                       df
Number of rows             342700
Number of columns          10
________________________
Column type frequency:
  factor                   10
________________________
Group variables            None

-- Variable type: factor ----------------------------------------------------------------------
# A tibble: 10 x 6
   skim_variable       n_missing complete_rate ordered n_unique
 * <chr>                   <int>         <dbl> <lgl>      <int>
 1 i..test_date                0             1 FALSE         72
 2 cough                       0             1 FALSE          3
 3 fever                       0             1 FALSE          3
 4 sore_throat                 0             1 FALSE          3
 5 shortness_of_breath         0             1 FALSE          3
 6 head_ache                   0             1 FALSE          3
 7 corona_result               0             1 FALSE          3
 8 age_60_and_above            0             1 FALSE          3
 9 gender                      0             1 FALSE          3
10 test_indication             0             1 FALSE          3
   top_counts
 * <chr>
 1 20/: 10521, 19/: 10166, 22/: 9243, 21/: 9167
 2 0: 305922, 1: 36528, NUL: 250
 3 0: 323493, 1: 18957, NUL: 250
 4 0: 340770, 1: 1929, NUL: 1
 5 0: 341124, 1: 1575, NUL: 1
 6 0: 340287, 1: 2412, NUL: 1
 7 Neg: 321948, Pos: 14629, Oth: 6123
 8 NUL: 176691, No: 132240, Yes: 33769
 9 NUL: 124411, fem: 110834, mal: 107455
10 Oth: 309924, Abr: 22079, Con: 10697
```

Figure 9: Data Summary



```
> #Check the simple descriptive statistics of the data
> summary(df)
   ï..test_date     cough          fever          sore_throat    shortness_of_breath
 20/4/2020: 10521   0    :305922   0    :323493   0    :340770   0    :341124
 19/4/2020: 10166   1    : 36528   1    : 18957   1    :  1929   1    :  1575
 22/4/2020:  9243   NULL:    250   NULL:    250   NULL:      1   NULL:      1
 21/4/2020:  9167
 16/4/2020:  8972
 1/4/2020 :  8645
 (Other)  :285986
 head_ache      corona_result      age_60_and_above     gender
 0    :340287   Negative:321948    No   :132240       female:110834
 1    :  2412   Other    :  6123   NULL:176691        male  :107455
 NULL:      1   Positive: 14629    Yes  : 33769       NULL  :124411



                test_indication
 Abroad                 : 22079
 Contact with confirmed: 10697
 Other                  :309924
```

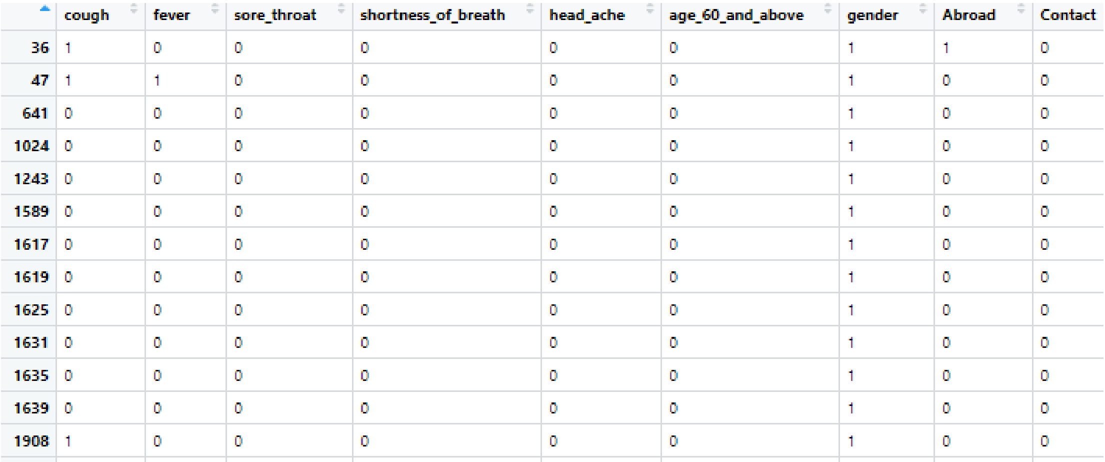
Figure 10: Simplify version of descriptive statistics for the dataset



```
> # Check whether any missing value in the dataset
> sum(is.na(df))
[1] 0
> sum(is.null(df))
[1] 0
> sum(df$cough=="NULL")
[1] 250
> sum(df$fever=="NULL")
[1] 250
> sum(df$sore_throat=="NULL")
[1] 1
> sum(df$shortness_of_breath=="NULL")
[1] 1
> sum(df$head_ache=="NULL")
[1] 1
> sum(df$corona_result=="NULL")
[1] 0
> sum(df$age_60_and_above=="NULL")
[1] 176691
> sum(df$gender=="NULL")
[1] 124411
> sum(df$test_indication=="NULL")
[1] 0
```

Figure 11: Check Missing value

After proceeding with all the data cleaning process, the clean dataset will be looking like figure 12.



	cough	fever	sore_throat	shortness_of_breath	head_ache	age_60_and_above	gender	Abroad	Contact
36	1	0	0	0	0	0	1	1	0
47	1	1	0	0	0	0	1	0	0
641	0	0	0	0	0	0	1	0	0
1024	0	0	0	0	0	0	1	0	0
1243	0	0	0	0	0	0	1	0	0
1589	0	0	0	0	0	0	1	0	0
1617	0	0	0	0	0	0	1	0	0
1619	0	0	0	0	0	0	1	0	0
1625	0	0	0	0	0	0	1	0	0
1631	0	0	0	0	0	0	1	0	0
1635	0	0	0	0	0	0	1	0	0
1639	0	0	0	0	0	0	1	0	0
1908	1	0	0	0	0	0	1	0	0

Figure 12: Clean data

3.3 Data Exploration Stage

In this data exploration stage, this project will do some analysis on the data which include some comparison between the symptoms and the results.

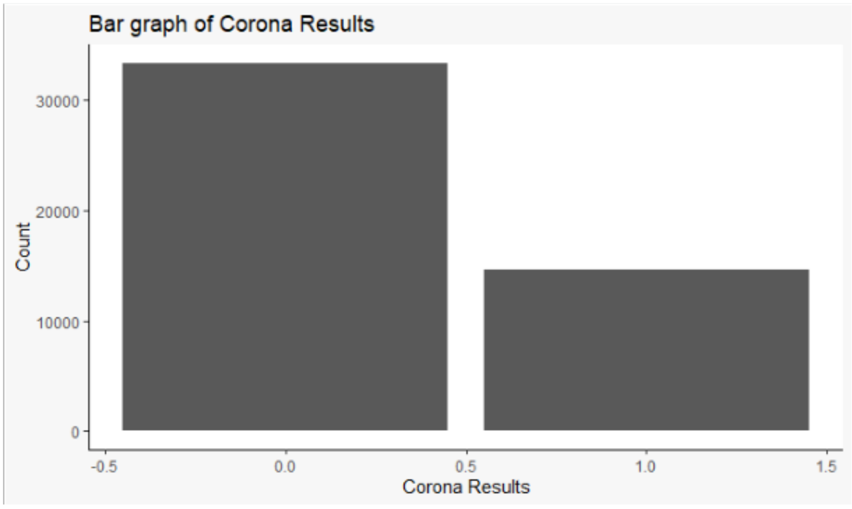


Figure 13: Overall corona result of this dataset

From figure 13, the dataset showed the patient that does not infect with the virus is lesser than who infected.

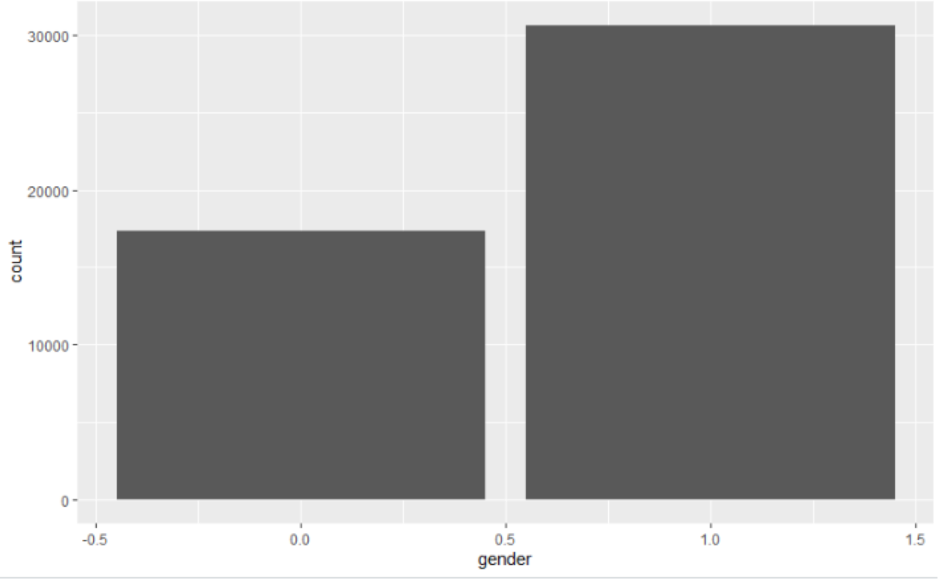


Figure 14: Comparison between gender and corona result

From figure 14, this dataset has more male patient who infected this corona virus than female patient.

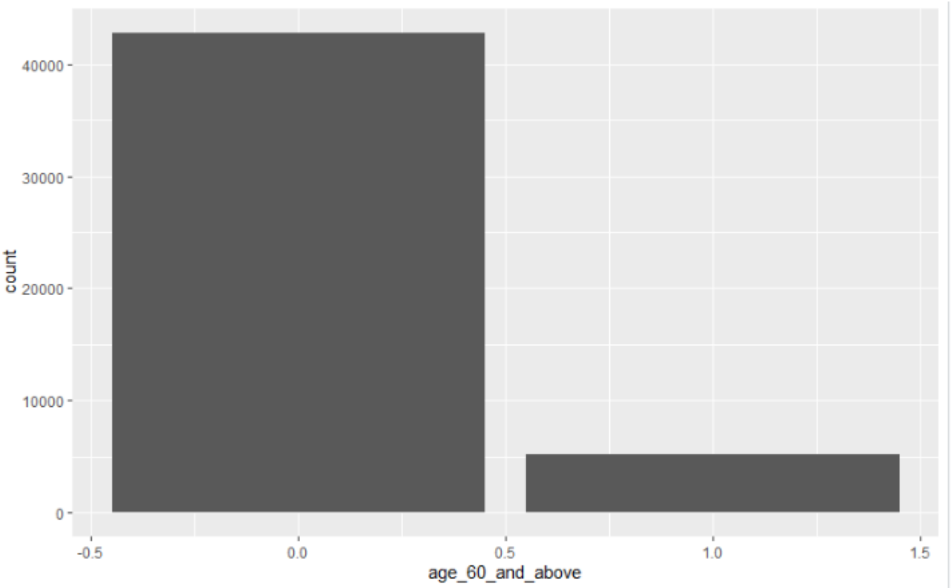


Figure 15: Comparison between age and corona result

Figure 15 showed that generally there are more young patient who has the corona virus than the older one.

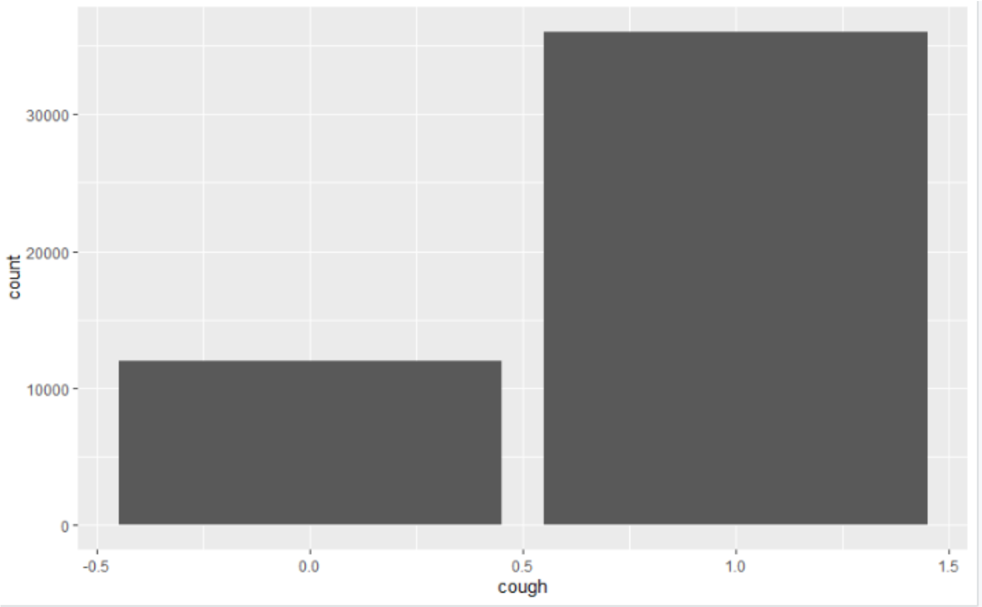


Figure 16: Comparison between cough and corona result

Figure 16 showed the patient who having symptom like cough have higher chances to get infected with this corona virus.

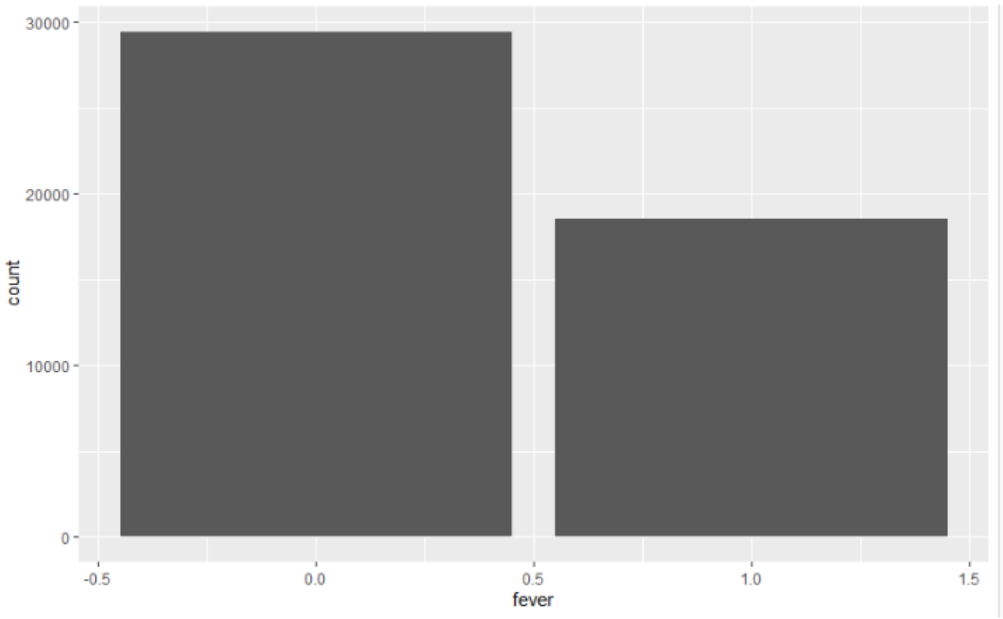


Figure 17: Comparison between fever and corona result

Figure 17 showed the patient who having symptom like fever are not necessary to get infected with this corona virus.

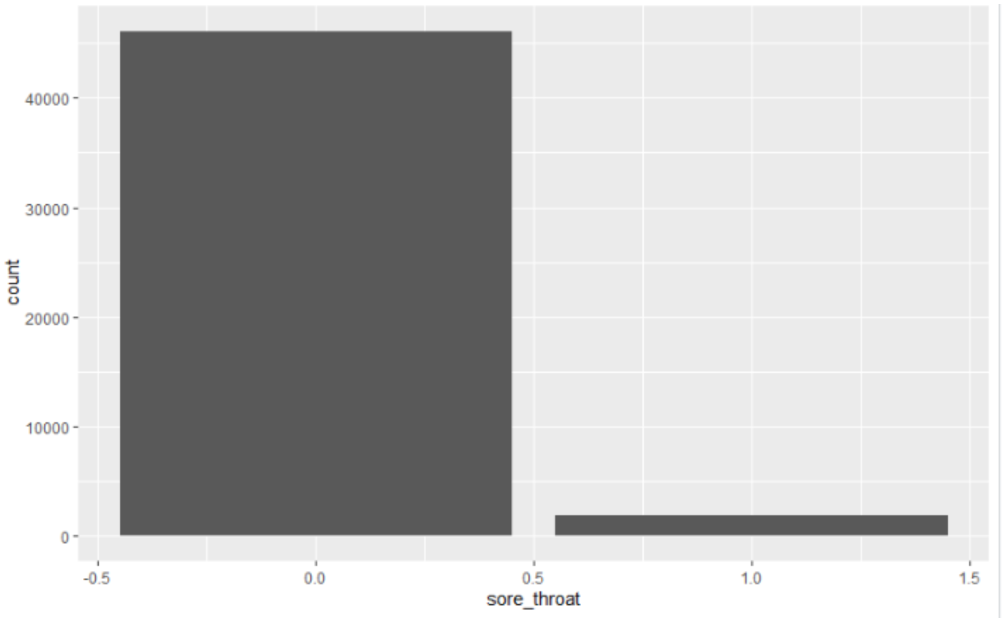


Figure 18: Comparison between sore throat and corona result

Figure 18 showed the patient who having symptom like sore throat are not necessary to get infected with this corona virus.

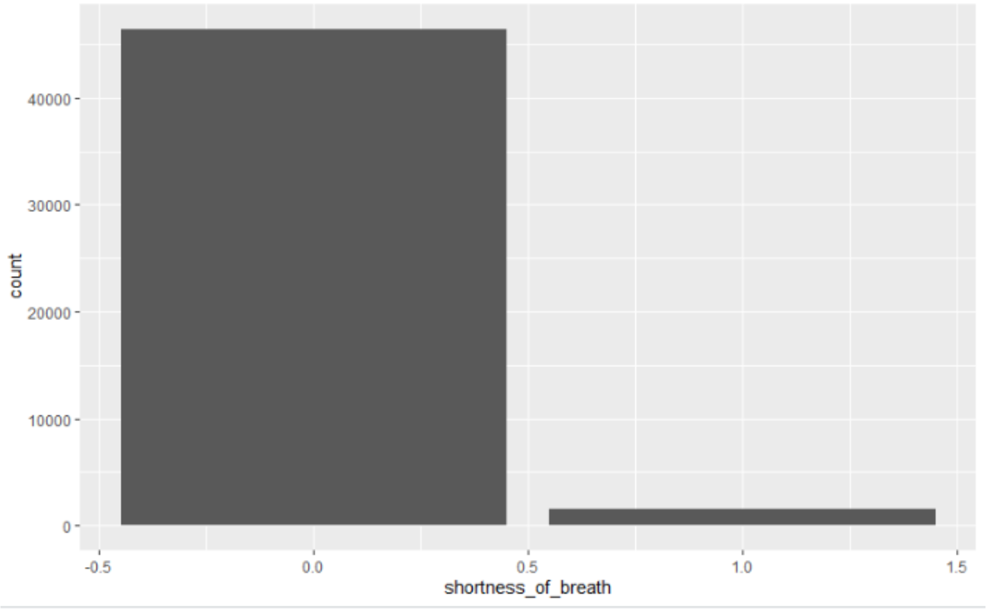


Figure 19: Comparison between shortness of breath and corona result

Figure 19 showed the patient who having symptom like shortness of breath are not necessary to get infected with this corona virus.

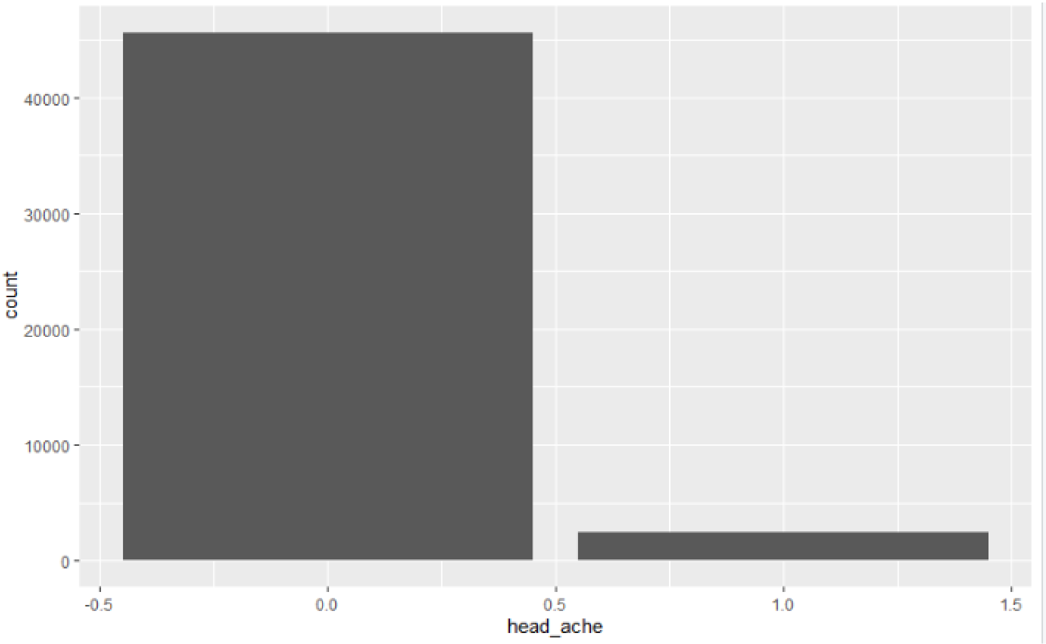


Figure 20: Comparison between headache and corona result

Figure 20 showed the patient who having symptom like headache are not necessary to get infected with this corona virus.

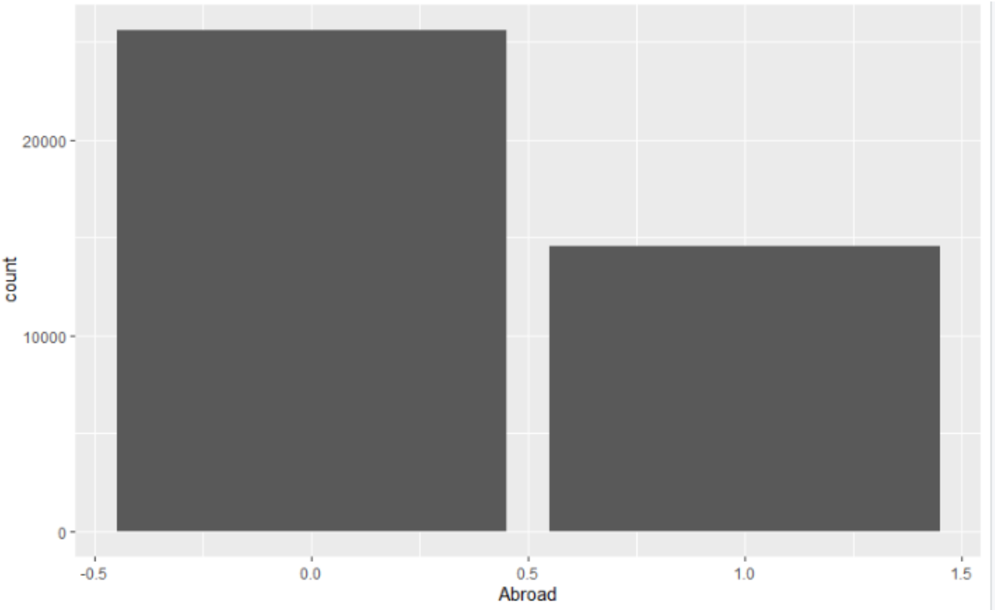


Figure 21: Comparison between abroad and corona result

Figure 21 showed the patient who having recent activity like abroad are not necessary to get infected with this corona virus.

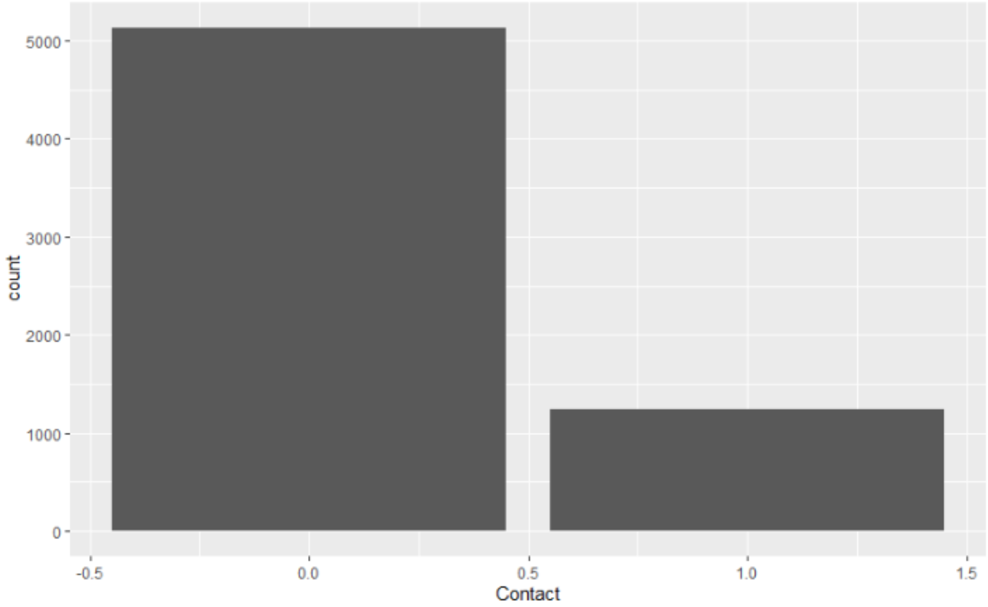


Figure 22: Comparison between contact and corona result

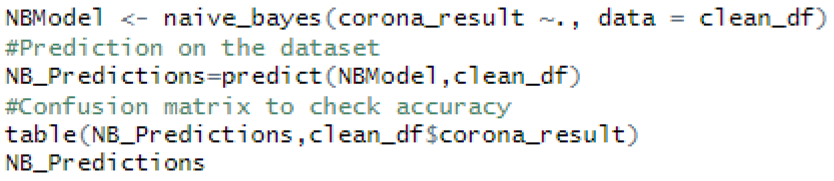
Figure 22 showed the patient who having recent activity like close contact with patient are not necessary to get infected with this corona virus.

3.4 Data Modelling stage

For this data modelling stage, this project will be using two different algorithms as the prediction algorithm which are Naive Bayes Classification and Decision Tree Algorithm. After that, this project will compare the results for both algorithms and select the best algorithm.

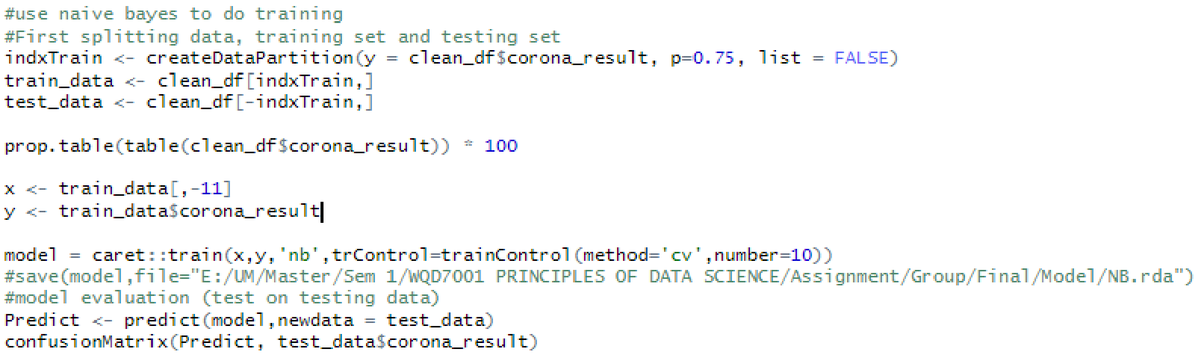
3.4.1 Naïve Bayes Classification

For this naïve bayes classification, there are two different methods to apply in R language. The first one is without any splitting data where just use the naïve bayes function from the library and train the data. And the second one required to split the data into two which representing train data and test data. After the model training the train data will use the test data to do prediction accuracy testing.



```
NBModel <- naive_bayes(corona_result ~., data = clean_df)
#Prediction on the dataset
NB_Predictions=predict(NBModel,clean_df)
#Confusion matrix to check accuracy
table(NB_Predictions,clean_df$corona_result)
NB_Predictions
```

Figure 23: Naïve Bayes Function (Without splitting data)

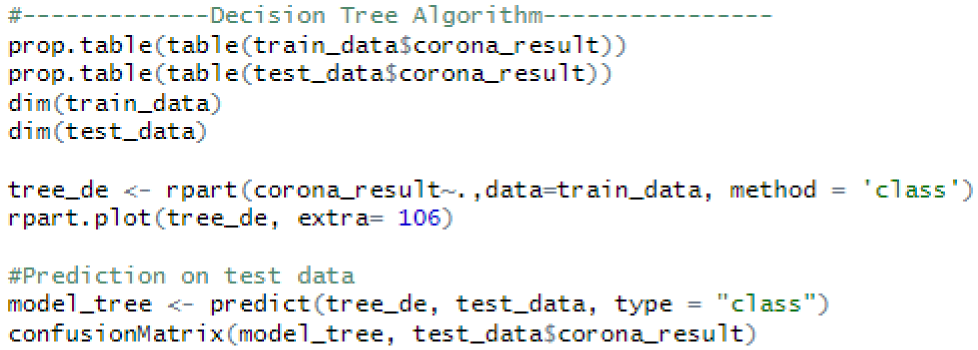


```
#use naive bayes to do training
#First splitting data, training set and testing set
indxTrain <- createDataPartition(y = clean_df$corona_result, p=0.75, list = FALSE)
train_data <- clean_df[indxTrain,]
test_data <- clean_df[-indxTrain,]

prop.table(table(clean_df$corona_result)) * 100

x <- train_data[,-11]
y <- train_data$corona_result

model = caret::train(x,y,'nb',trControl=trainControl(method='cv',number=10))
#save(model,file="E:/UM/Master/Sem 1/WQD7001 PRINCIPLES OF DATA SCIENCE/Assignment/Group/Final/Model/NB.rda")
#model evaluation (test on testing data)
Predict <- predict(model,newdata = test_data)
confusionMatrix(Predict, test_data$corona_result)
```

Figure 24: Naïve Bayes Function (With splitting data)

3.4.2 Decision Tree Algorithm

For decision tree algorithm, the data will also split into two part which are train data and test data. After that, the train data will be used to training with this decision tree algorithm. After the training process is done, this project will be testing the prediction accuracy using the test data.

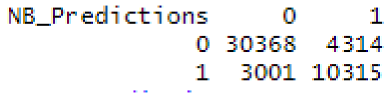


```
#-------------Decision Tree Algorithm----------------
prop.table(table(train_data$corona_result))
prop.table(table(test_data$corona_result))
dim(train_data)
dim(test_data)

tree_de <- rpart(corona_result~.,data=train_data, method = 'class')
rpart.plot(tree_de, extra= 106)

#Prediction on test data
model_tree <- predict(tree_de, test_data, type = "class")
confusionMatrix(model_tree, test_data$corona_result)
```

Figure 25: Decision tree algorithm (With splitting data)

3.5 Analysis Results Stage

After all the algorithms being trained and tested, this project comparing those three algorithms methods' results which are naïve bayes function without splitting data, naïve bayes function with splitting data and decision tree algorithm with splitting data. This project will use the confusion matrix to determine the accuracy for the algorithms.

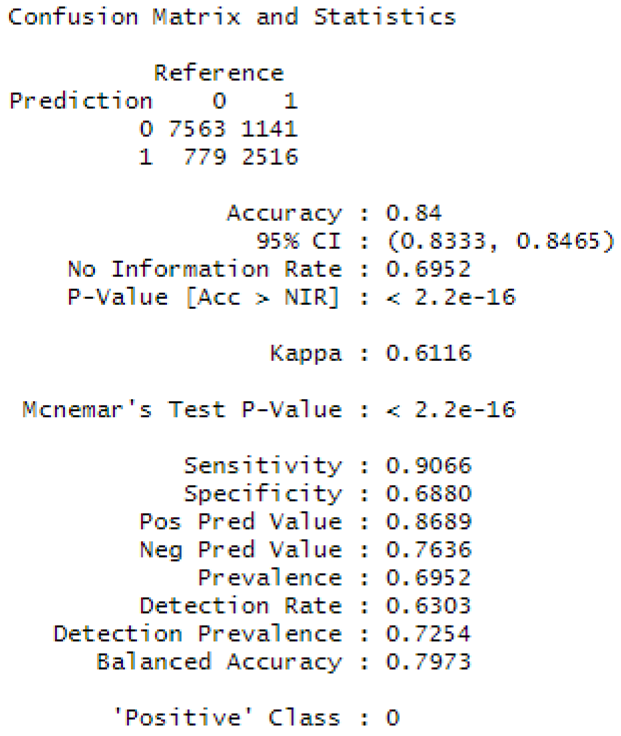


```
NB_Predictions     0     1
             0 30368  4314
             1  3001 10315
```

Figure 26: Confusion Matrix for naïve bayes function without splitting data

Based on figure 26, we can calculate the accuracy of this naïve bayes function without splitting data are 0.8475.

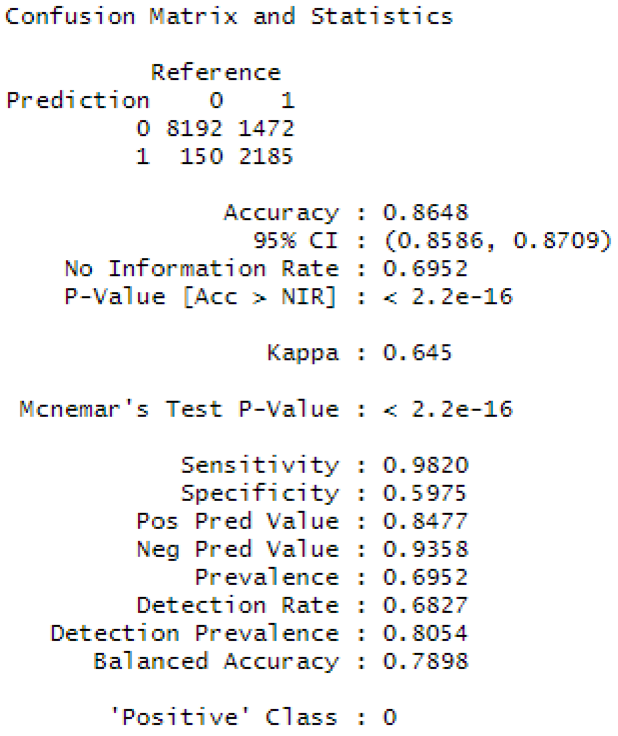
For the naïve bayes function with splitting data and decision tree algorithm with splitting data, we put 0.75 as the splitting ratio.



```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 7563 1141
         1  779 2516

               Accuracy : 0.84
                 95% CI : (0.8333, 0.8465)
    No Information Rate : 0.6952
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6116

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9066
            Specificity : 0.6880
         Pos Pred Value : 0.8689
         Neg Pred Value : 0.7636
             Prevalence : 0.6952
         Detection Rate : 0.6303
   Detection Prevalence : 0.7254
      Balanced Accuracy : 0.7973

       'Positive' Class : 0
```

Figure 27: Results of naïve bayes function with splitting data

As from figure 27, we can clearly see that the accuracy of naïve bayes function with splitting data is also around 0.84.



```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 8192 1472
         1  150 2185

               Accuracy : 0.8648
                 95% CI : (0.8586, 0.8709)
    No Information Rate : 0.6952
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.645

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9820
            Specificity : 0.5975
         Pos Pred Value : 0.8477
         Neg Pred Value : 0.9358
             Prevalence : 0.6952
         Detection Rate : 0.6827
   Detection Prevalence : 0.8054
      Balanced Accuracy : 0.7898

       'Positive' Class : 0
```

Figure 28: Results of decision tree algorithm with splitting data

As from figure 27, the prediction accuracy result of decision tree algorithm with splitting data is also around 0.8648 which are much higher than the naïve bayes algorithms method.

4.0 Discussion

After we compared all the results for each algorithm, we can clearly see that decision tree algorithm with splitting data has higher prediction accuracy compare to naïve bayes algorithm. Besides, for the naïve bayes algorithm with training data and without training data, both prediction values are almost similar. This indicate that the internal library function did consider about training process. Besides, based on the data exploration, we also noticed that this dataset contained unbiased variable that might affect the accuracy of the prediction results.

5.0 Limitation and conclusion

Even though the prediction system is able to predict with the accuracy up to 86.5%, there are still some of the limitation that need to be mentioned. First will be the unbiased variable data set. Based on figure 29, we can clearly see that this dataset has a lot of unbiased variable which might cause the results to become unbiased. Next is the size of the dataset. Because of those limitation of data sources, this dataset are very limit and the amount of data cannot represent for all the possibility of infecting this corona virus. Therefore, in future, we hope we can get more different dataset to enlarge the sample size.

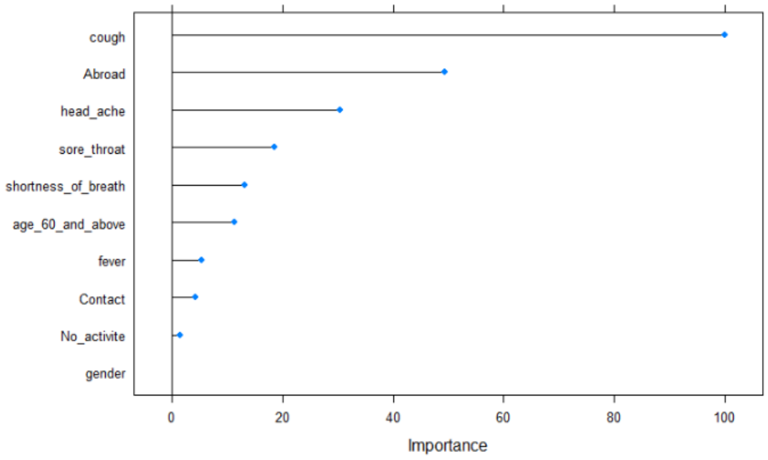


Figure 29: Variable performance for each categor

Reference

Alixdumont. (2021, January 4). Symptomes. Retrieved from https://www.kaggle.com/alixdumont/symptomes

Coronavirus symptoms fall into six different groupings, study finds. (2020, July 17). Retrieved from https://www.theguardian.com/world/2020/jul/17/covid-19-symptoms-falls-into-six-different-groupings-study-finds-coronavirus

COVID-19: Study reveals six clusters of symptoms that could be used as a clinical prediction tool. (2020, July 20). Retrieved from https://www.bmj.com/content/370/bmj.m2911

Remote monitoring of COVID-19 symptoms - Early prediction of aggressive COVID-19 progression and hospitalization. (2020, May 4). Retrieved from https://www.mayo.edu/research/remote-monitoring-covid19-symptoms/people-with-covid19

Smith, C. (2020, September 26). This simple test might predict how serious your coronavirus case will be. Retrieved from https://bgr.com/2020/09/27/coronavirus-symptoms-severe-covid-19-prediction-rdw-marker/