Cairo University
Faculty of Engineering
Credit Hours System

Big Data

CMPN-451 Project

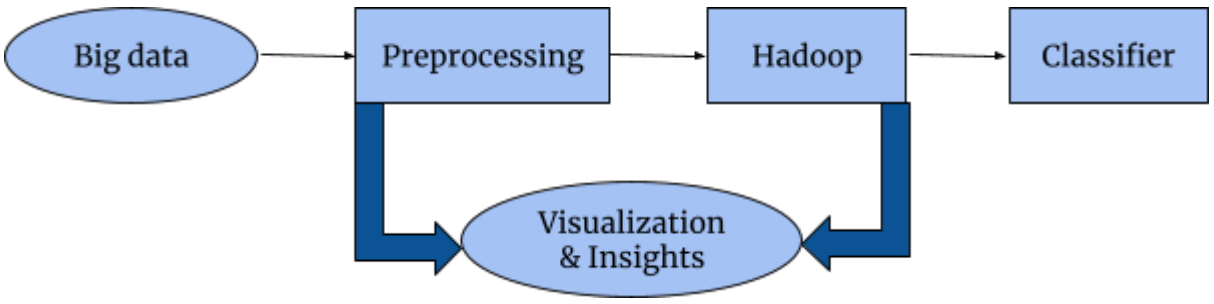
Submitted by:

- Rehab Ahmed - 1162297
- Salma Farghaly - 1164263
- Nancy Hassan - 1162114
- Omar Taher - 1165213

Problem description:

The past year the world has experienced the crash of COVID-19 virus which has spread all over the world. The virus apparently has many symptoms such as fever, tiredness, sore throat, etc. However, such symptoms vary from person to person, and not all people get the virus with the same severity.
Our dataset contains of cases of genders male, female and transgender, of different age groups across different countries, and the symptoms that appeared for each case and the degree of severity of the virus. We have 4 degrees of severity: None, Mild, Moderate, Severe. The symptoms in the dataset include: fever, tiredness, dry cough, difficulty in breathing, sore throat, pains, nasal congestion, runny nose, diarrhea.
Using the dataset, we want to target the severity of the virus, and try to predict the severity of a person's case given his symptoms in order to be able to take the appropriate medical protocol. Severe cases may need immediate shift to the nearest hospital, moderate cases may need to visit the doctor and have an immediate medicine prescription. Mild cases may need to have medicines prescribed to strengthen their immune system, but may not necessarily need to stay at the hospital. None severe cases may only stay at home to prevent being in contact with other people.

Project pipeline:



Preprocessing

The dataset contained 27 columns, comprised of 11 symptoms, 5 columns of different age groups (0-9, 10-19, 20-24, 25-59, 60-), 3 columns of gender (male, female, transgender), 4 columns of severity (none, mild, moderate, severe), 3 columns of whether the person was in contact with an infected person or not (yes, no, unknown) and 1 column for the country.
As we can see, this is totally inefficient so we decided to merge the columns of similar attributes. The preprocessing takes were as follows:

Task	Reason
1. Merging columns related together like 5 different age group columns into a single Age column containing 5 categories. This was repeated for the Gender, Severity and Contact columns.	This decreased the size of the dataset and made it more optimized. Instead of having 27 columns, now they are 16.
2. The data tuples contained 1 or 0 as labels of true/ false. Since the data is categorical, these 1s or 0s are replaced with the appropriate description for each class.	This was essential for hadoop in order to be able to get the relationships between the classes, they should be named categories for each class.

Hadoop

In order to be able to deal with the big data, we used hadoop file system and applied map reduce to get the relationships of each class (Severity: severe, moderate, mild, none) with all the other attributes. The map phase split the dataset into several splits. The mapper produced a list of key-value pairs with the same key. This is done for each split. The reduce phase combines the key-value pair lists from the mapper and counts the number of appearances of each. This generates the number of each class combination with the attributes for the entire dataset. An output file is generated, and is used to construct our classifier (Naive Bayes) as shown below.

Classifier Naive Bayes :

Trial 1

We split the task of naive bayes into :

a. Mapreduce using Hadoop to calculate the count of each attribute given each class
b. Implement the naive bayes from scratch in R

our model implementation as following:

1. Split the data into 60 % train 20 % validation 20% test

2. The training phase (**Learning Phase**) is mostly implemented using Hadoop as counting the number of occurance of each attribute as well as each class label counts.

3. The validation phase is tuning the hyper parameters which is laplace smoothing (α) to handle zero probability element as following:

$$P(x_i \mid C_k) = \frac{(number\ of\ points\ such\ that\ x_i occurs\ and\ class\ label = C_k) + \alpha}{number\ of\ points\ where\ class\ label = C_k + \alpha k}$$

Where α can be any real value > 0
k is the number of class labels.

We take a set of alpha values α**_set = [0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000].**

4. Then the test phase to evaluate our model on examples that it has never seen before.

Evaluation of our model:

Train	Validation	Test
Accuracy: 25.23 %	Accuracy: 25.483 % Laplace smoothing: 0.01	Accuracy: 24.444 %

Train

```
Confusion Matrix and Statistics


          Severe  Mild  None Moderate
  Severe   22378 22050 22200    22147
  Mild     25036 25341 25010    25074
  None       192   212   231      209
  Moderate     0     0     0        0

Overall Statistics

               Accuracy : 0.2523
                 95% CI : (0.2503, 0.2542)
    No Information Rate : 0.2505
    P-Value [Acc > NIR] : 0.03458

                  Kappa : 0.0024

 Mcnemar's Test P-Value : < 2e-16

Statistics by Class:

                     Class: Severe Class: Mild Class: None Class: Moderate
Sensitivity                 0.4701      0.5323    0.004869          0.0000
Specificity                 0.5340      0.4728    0.995702          1.0000
Pos Pred Value              0.2521      0.2522    0.273697             NaN
Neg Pred Value              0.7510      0.7516    0.750523          0.7505
Prevalence                  0.2505      0.2504    0.249584          0.2495
Detection Rate              0.1177      0.1333    0.001215          0.0000
Detection Prevalence        0.4670      0.5285    0.004440          0.0000
Balanced Accuracy           0.5020      0.5025    0.500286          0.5000
>
```

In medical tasks we should focus on precision and recall

Precision is a useful metric in cases where False Positive is a higher concern than False Negatives.

Recall tells us how many of the actual positive cases we were able to predict correctly with our model.

Recall ⇒ sensitivity
precision ⇒ Pos pred value
In class severe the recall is 47% which means that we predict class severe 47% times .
In class mild the recall is 53% which means that we predict class mild 53% times
On the other hand class none and moderate the recall is 0%
Which make us try to merge the classes that symptoms are close to each other

Test

```
Confusion Matrix and Statistics


          Severe Mild None Moderate
  Severe    7228 7492 7369     7354
  Mild      8477 8190 8517     8431
  None        84   82   70       66
  Moderate     0    0    0        0

Overall Statistics

               Accuracy : 0.2444
                 95% CI : (0.2411, 0.2478)
    No Information Rate : 0.2518
    P-Value [Acc > NIR] : 1

                  Kappa : -0.0061

 Mcnemar's Test P-Value : <2e-16

Statistics by Class:

                     Class: Severe Class: Mild Class: None Class: Moderate
Sensitivity                 0.4578      0.5195    0.004387          0.0000
Specificity                 0.5330      0.4658    0.995106          1.0000
Pos Pred Value              0.2455      0.2436    0.231788             NaN
Neg Pred Value              0.7476      0.7454    0.748073          0.7498
Prevalence                  0.2492      0.2488    0.251831          0.2502
Detection Rate              0.1141      0.1293    0.001105          0.0000
Detection Prevalence        0.4647      0.5305    0.004766          0.0000
Balanced Accuracy           0.4954      0.4927    0.499746          0.5000
> |
```

Other trials:

Trial 2

We change the predicted classes as following:

1. That “Moderate_sever” in most cases he/she should go to the doctor.
2. The “None_Mild” in most cases he/she is fine but can visit the doctor to make sure .
3.

We also try splitting the data into 70 % train 15 % validation 15% test to make sure that model learns on enough data.

Evaluation:

Train	Validation	Test
Accuracy: 50.2 %	Accuracy: 51 % Laplace smoothing: 1000	Accuracy:49 %

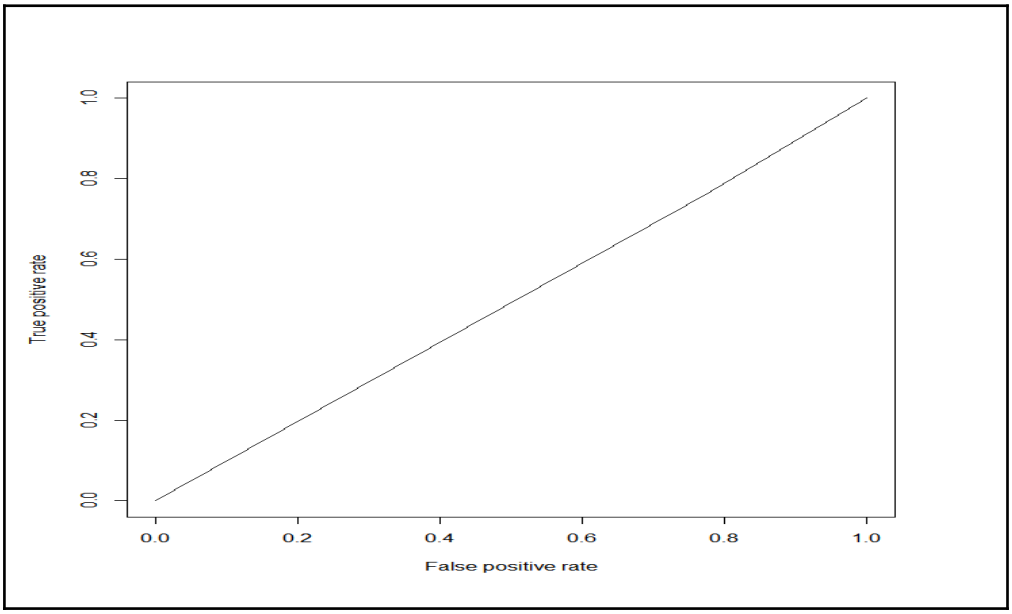
Test

```
Confusion Matrix and Statistics


                 Moderate_sever None_Mild
  Moderate_sever           5418      5661
  None_Mild               18374     18067

               Accuracy : 0.4942
                 95% CI : (0.4897, 0.4987)
    No Information Rate : 0.5007
    P-Value [Acc > NIR] : 0.9976

                  Kappa : -0.0108

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.2277
            Specificity : 0.7614
         Pos Pred Value : 0.4890
         Neg Pred Value : 0.4958
             Prevalence : 0.5007
         Detection Rate : 0.1140
   Detection Prevalence : 0.2331
      Balanced Accuracy : 0.4946

       'Positive' Class : Moderate_sever

>
```

Train

```
Confusion Matrix and Statistics


                 None_Mild Moderate_sever
  None_Mild          85197          84777
  Moderate_sever     25719          26067

               Accuracy : 0.5017
                 95% CI : (0.4996, 0.5038)
    No Information Rate : 0.5002
    P-Value [Acc > NIR] : 0.06999

                  Kappa : 0.0033

 Mcnemar's Test P-Value : < 2e-16

            Sensitivity : 0.7681
            Specificity : 0.2352
         Pos Pred Value : 0.5012
         Neg Pred Value : 0.5034
             Prevalence : 0.5002
         Detection Rate : 0.3842
   Detection Prevalence : 0.7665
      Balanced Accuracy : 0.5016

       'Positive' Class : None_Mild

>
```

TPR , FPR plot



AUC

```
> auc <- auc@y.values[[1]]
> auc
[1] 0.4945724
```

Failed Trial :

Here we try to change the behaviour of naive bayes by :

1. dropping the redundant features As None_ Experience , Gender , Country, Contact Don't Know, None_ symptoms
2. We want to detect the severe cases as a priority to be isolated in the hospital.
3. Merging none, mild , moderate as None_Severe
4. Splitting data into 70 % train 15 % validation 15% test to make sure that model learns on enough data

Evaluation:

Train	Validation	Test
Accuracy: 75%	Accuracy: 75.9% Laplace smoothing: 0.01	Accuracy: 75%

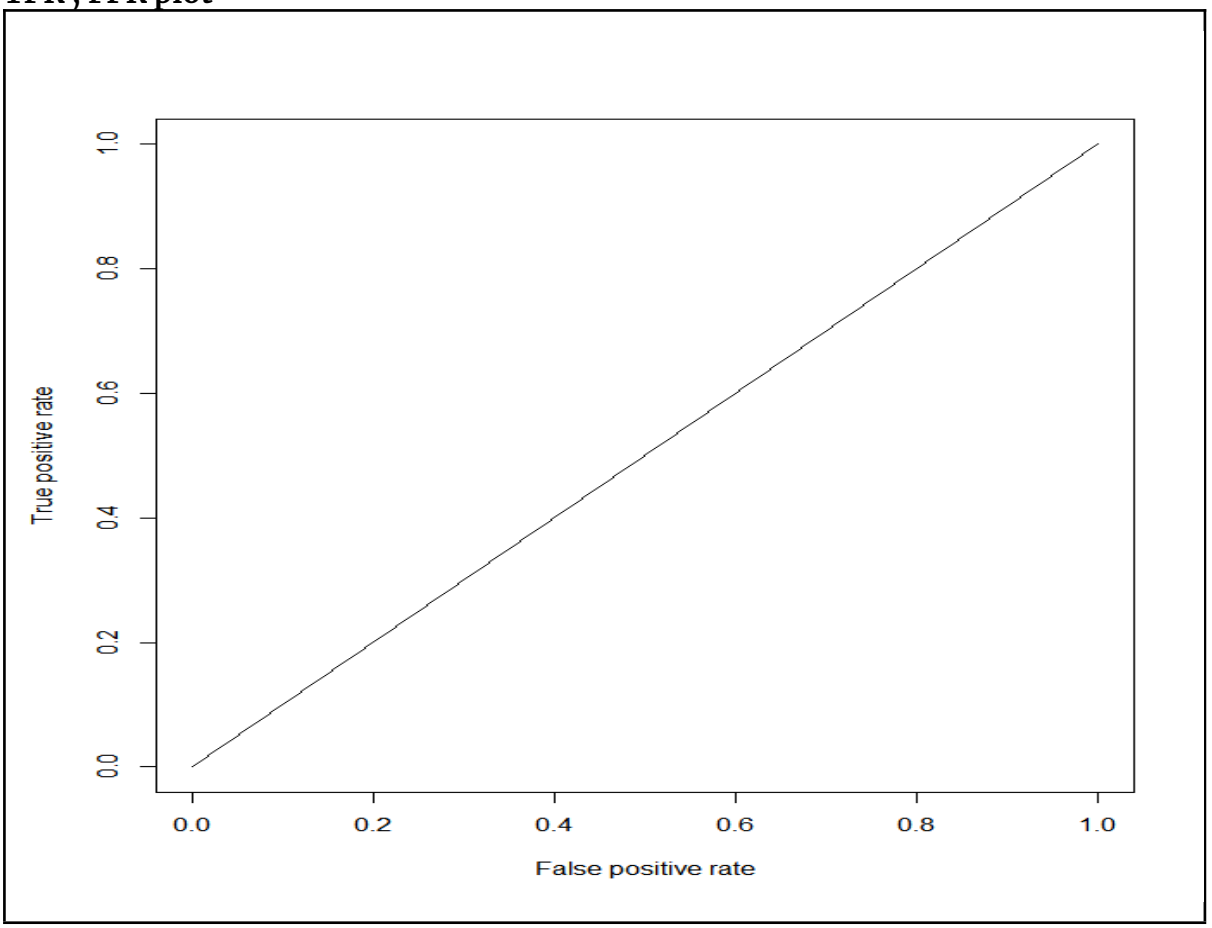
Train

```
Confusion Matrix and Statistics


          None  sever
  None  111009  36831
  sever      0      0

               Accuracy : 0.7509
                 95% CI : (0.7487, 0.7531)
    No Information Rate : 0.7509
    P-Value [Acc > NIR] : 0.5014

                  Kappa : 0

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.0000
            Specificity : 0.0000
         Pos Pred Value : 0.7509
         Neg Pred Value :    NaN
             Prevalence : 0.7509
         Detection Rate : 0.7509
   Detection Prevalence : 1.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : None

>
```

Test

```
Confusion Matrix and Statistics


          None sever
  None  23789  7891
  sever     0     0

               Accuracy : 0.7509
                 95% CI : (0.7461, 0.7557)
    No Information Rate : 0.7509
    P-Value [Acc > NIR] : 0.503

                  Kappa : 0

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.0000
            Specificity : 0.0000
         Pos Pred Value : 0.7509
         Neg Pred Value :    NaN
             Prevalence : 0.7509
         Detection Rate : 0.7509
   Detection Prevalence : 1.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : None
```

TPR , FPR plot



AUC

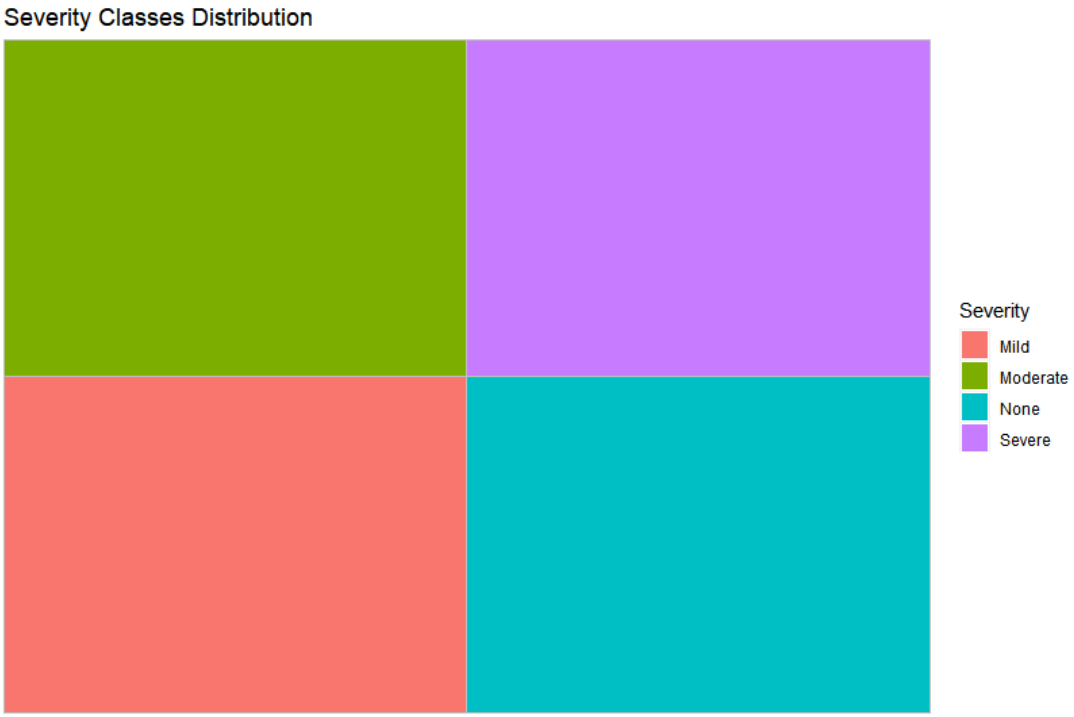
```
> auc
[1] 0.5
```

That 100% of the positives were successfully predicted by our model for class none only.
Cannot detect any severe class that was our priority unfortunately :(

Data Visualization

All the plots are created using library ggplot2.

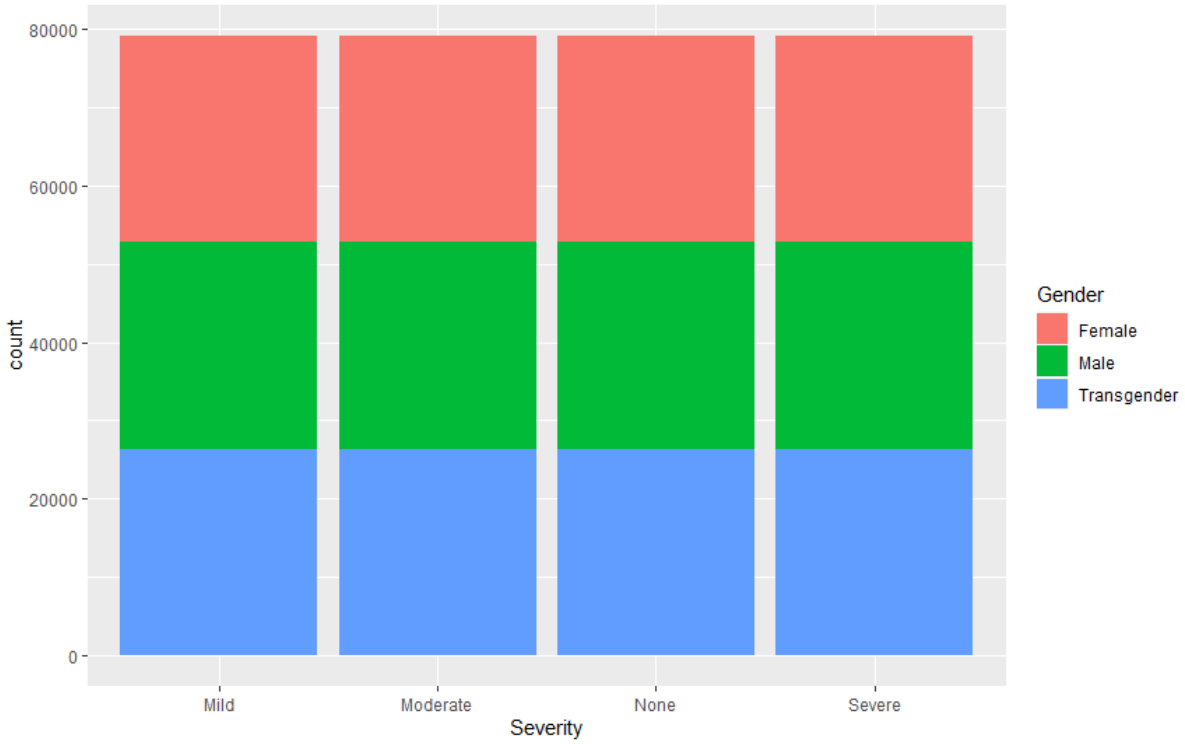
1. Classes Distribution

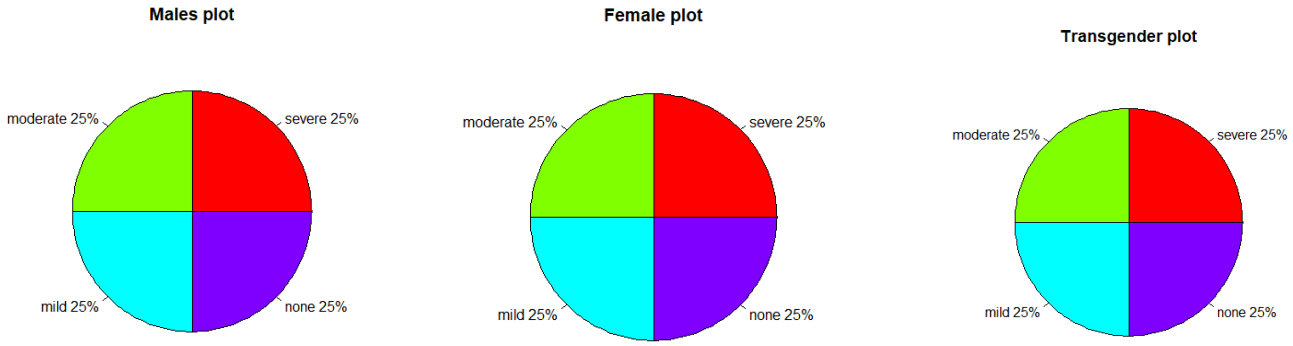


The plot shows that each class in the dataset is represented equally.

2. Gender Distributions

All types of gender belong to each class with the same probability.

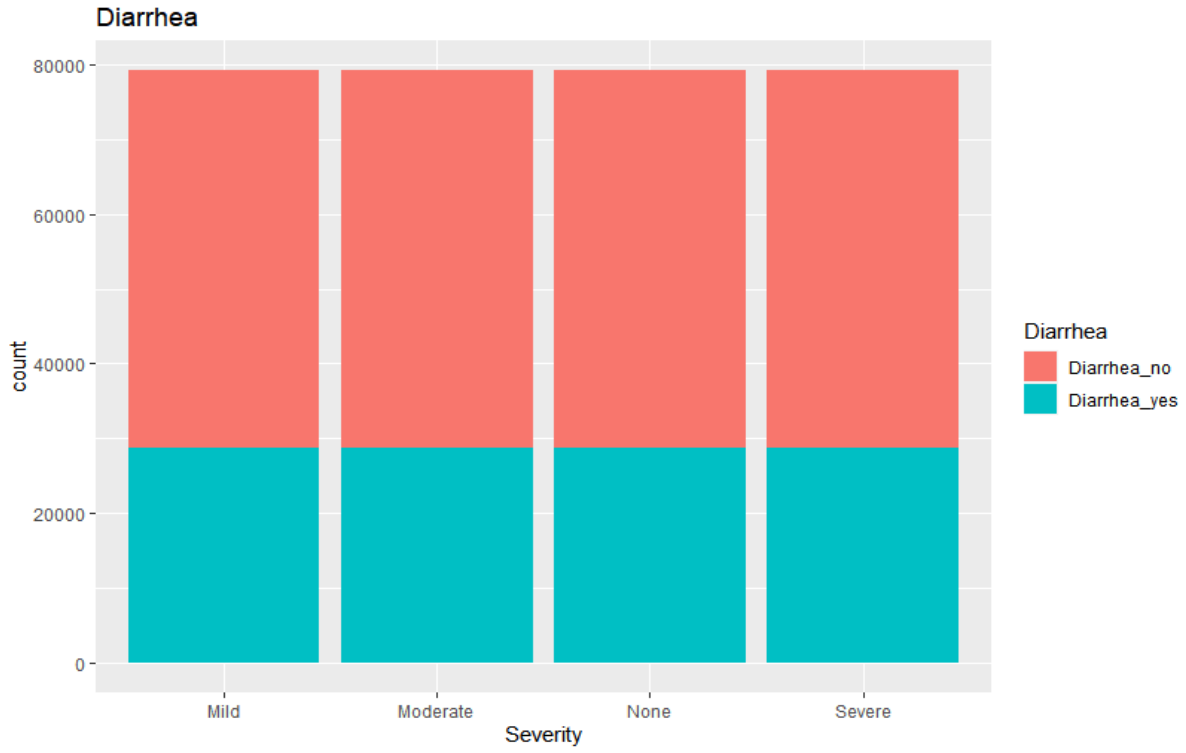


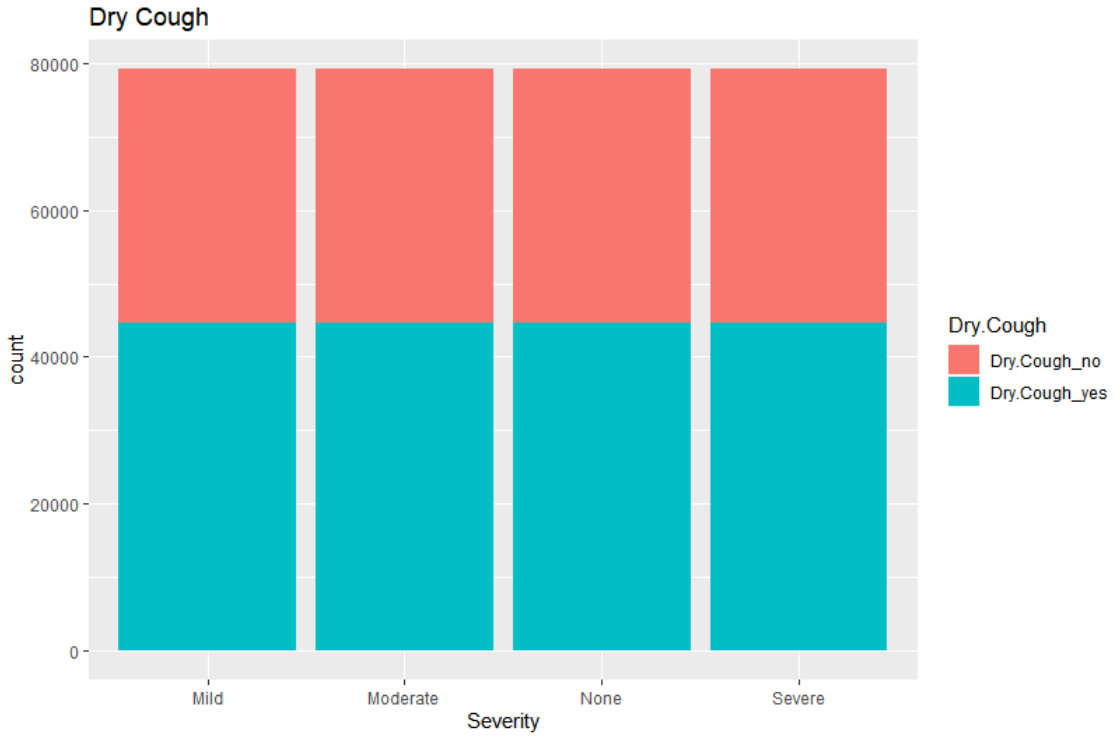


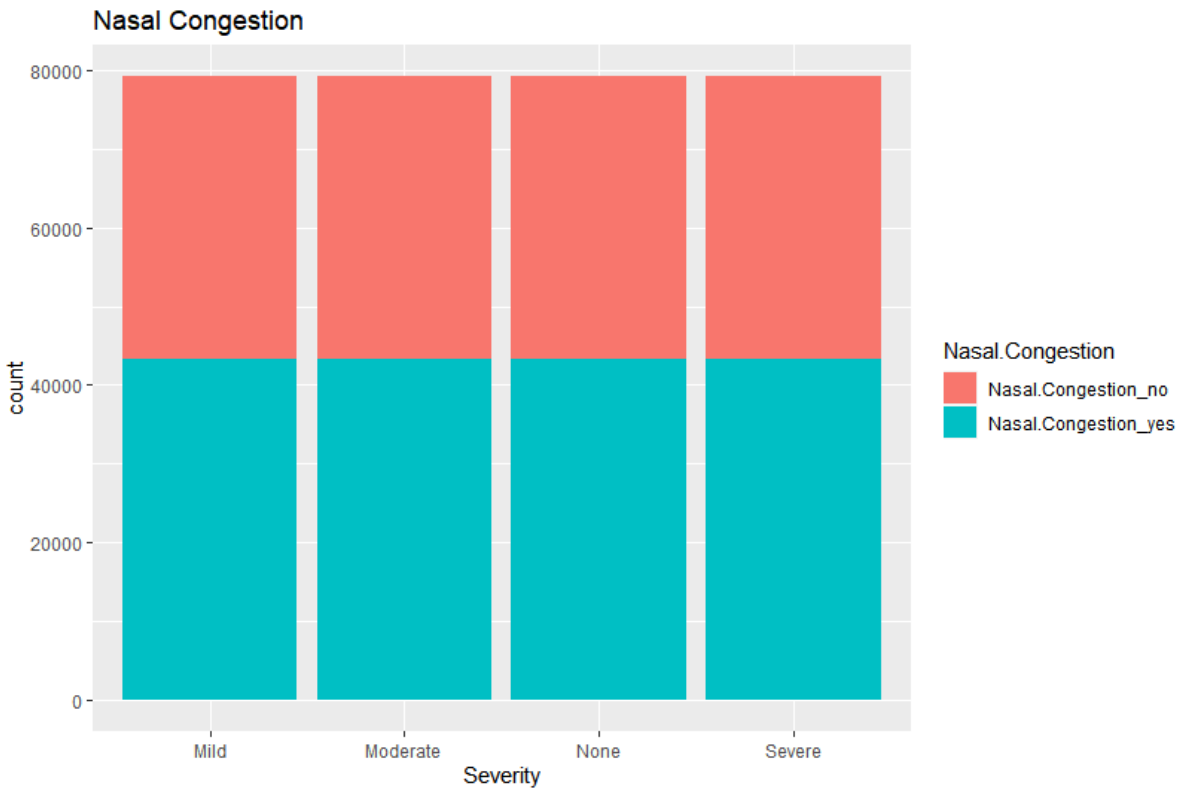
The plots show that each gender has the equal probability to belong to each class.

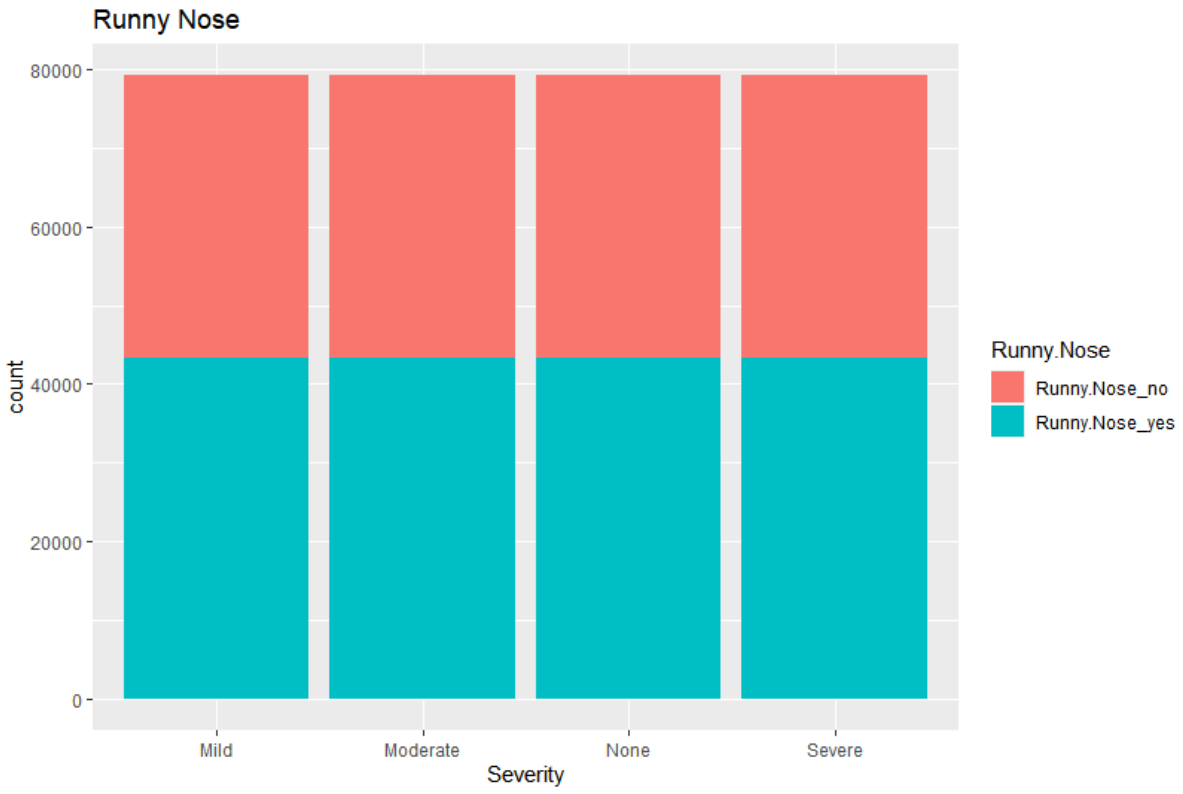
3. Symptoms Distribution

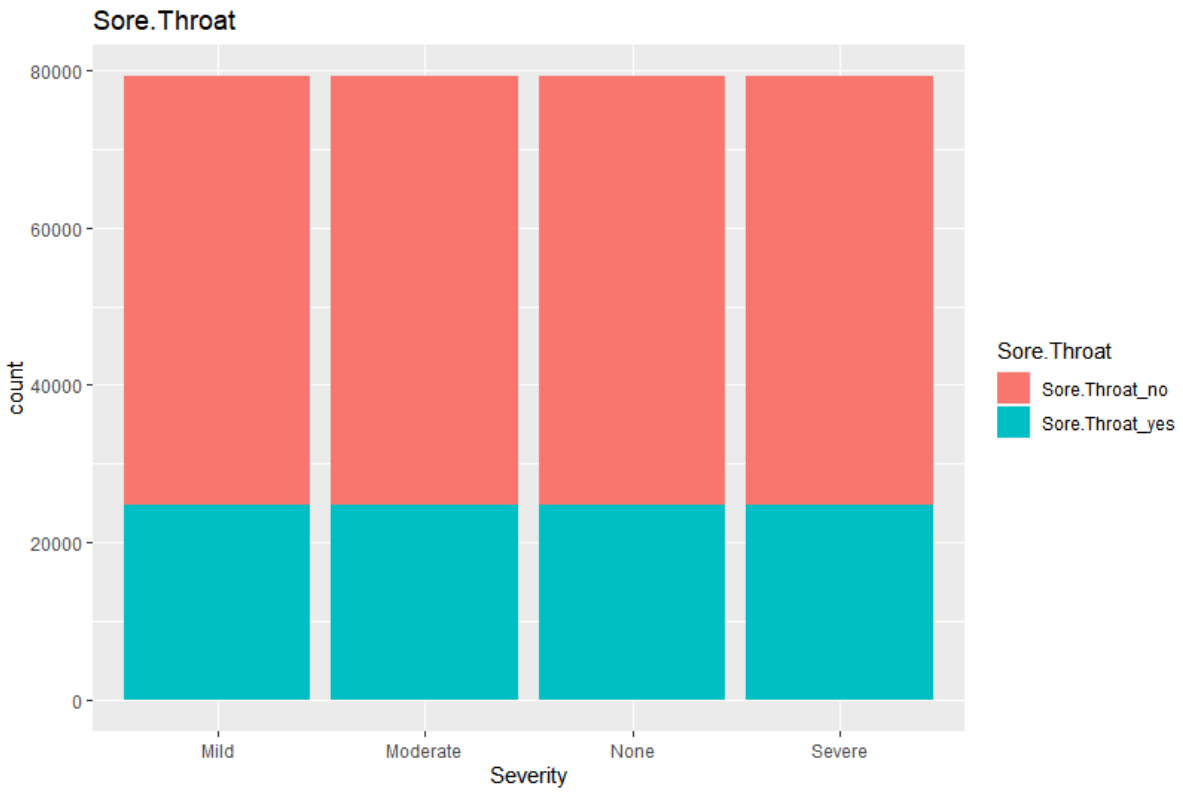
All Symptoms have the same distribution in all classes as shown in figures below :

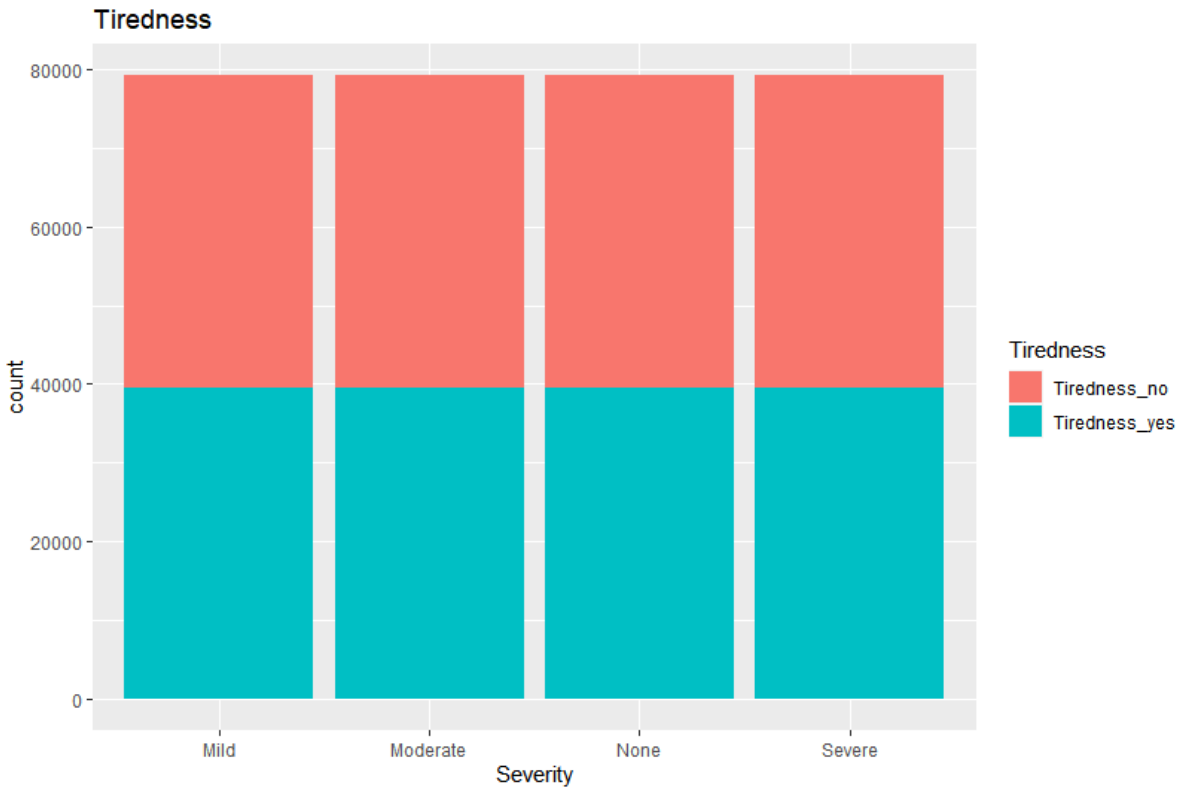





Nasal Congestion
80000
60000
40000
20000
0
count
Mild
Moderate
None
Severe
Severity
Nasal.Congestion
Nasal.Congestion_no
Nasal.Congestion_yes

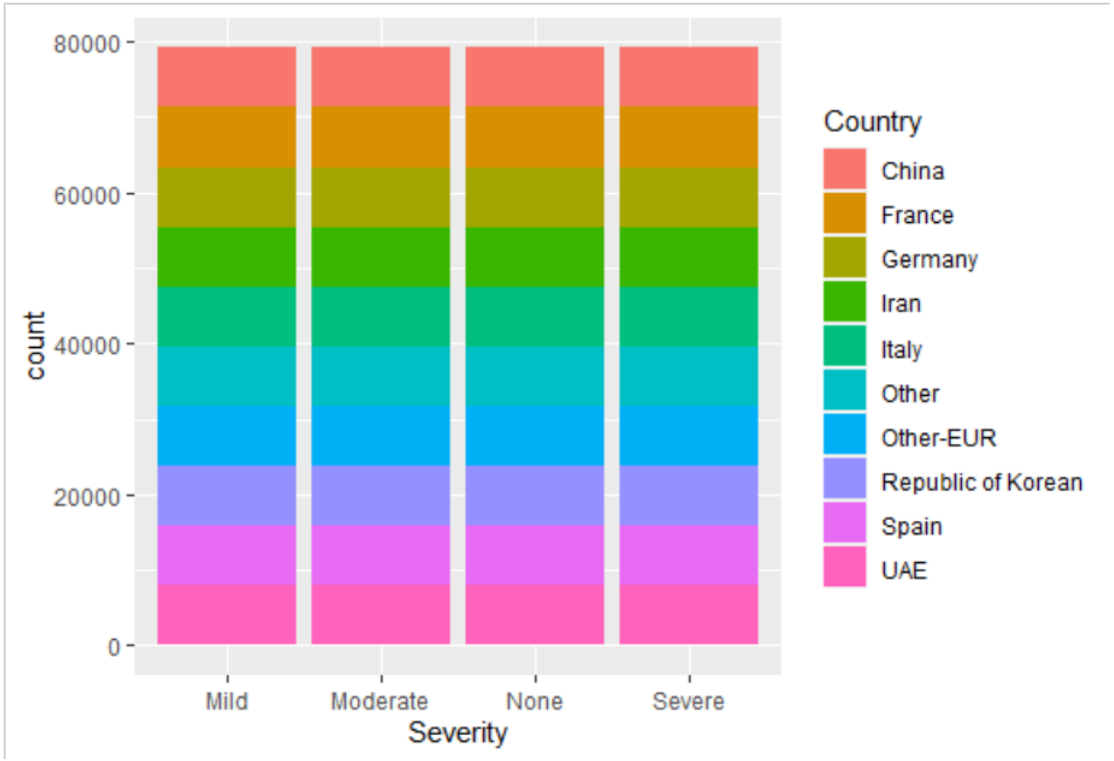

Runny Nose
80000
60000
40000
20000
0
count
Mild
Moderate
None
Severe
Severity
Runny.Nose
Runny.Nose_no
Runny.Nose_yes





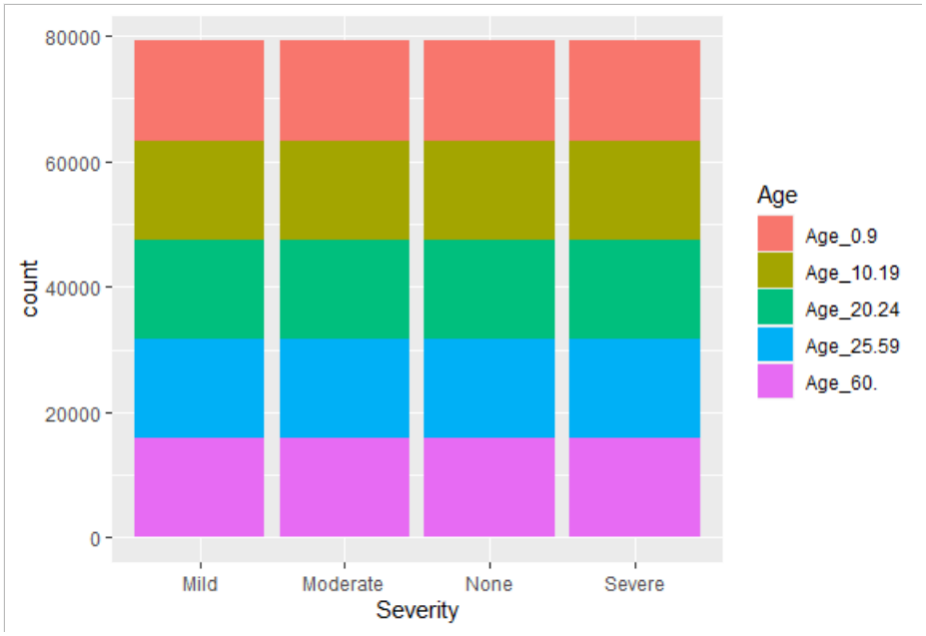
The plots show that no symptom has an impact on classifying the symptomatic.

We can also see the distribution of the countries in the dataset:



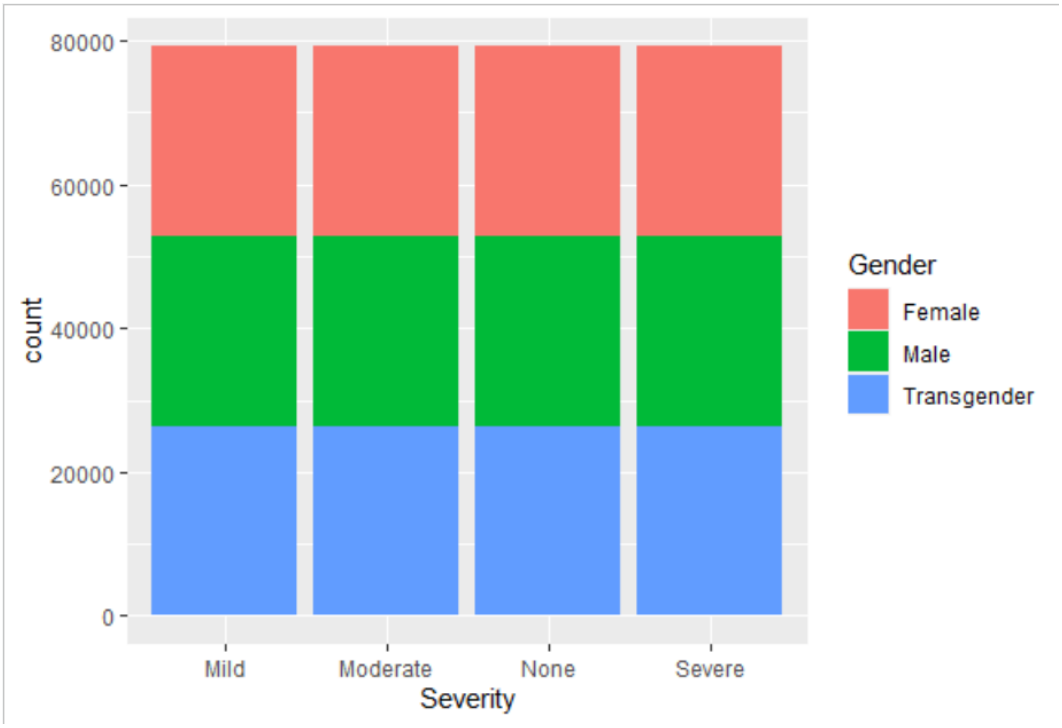
The plot shows that every country has the same number of the COVID-19 patients.

We can also see the distribution of the different age groups in the dataset:



The plot shows that age doesn't affect the severity of the case .

We can also see the distribution of the gender in the dataset:



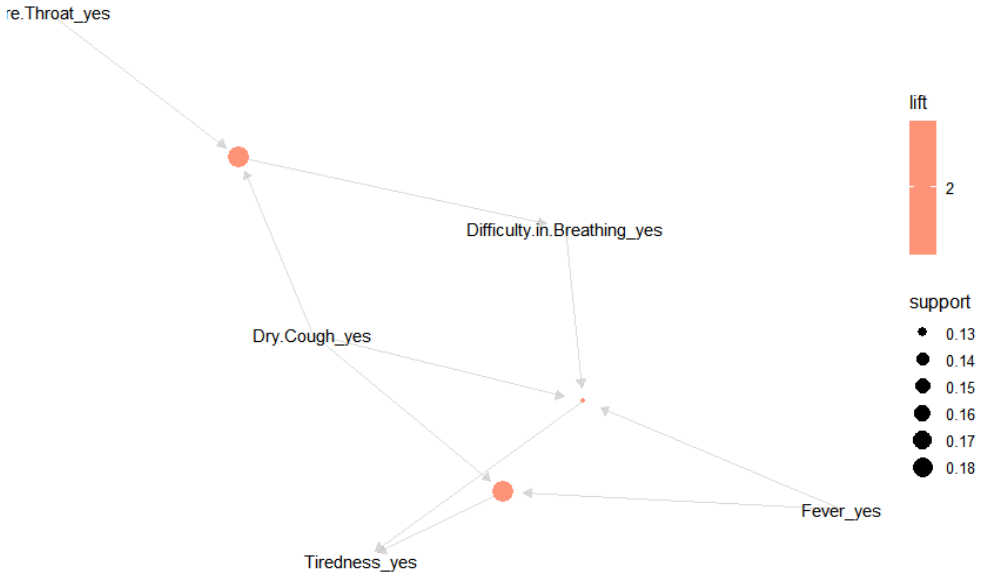
The plot shows that gender doesn’t affect the severity of the case .

Insights about dataset

Insights were generated using Association rules from the "arules" library in R. From the association rules, we came to the conclusions below:

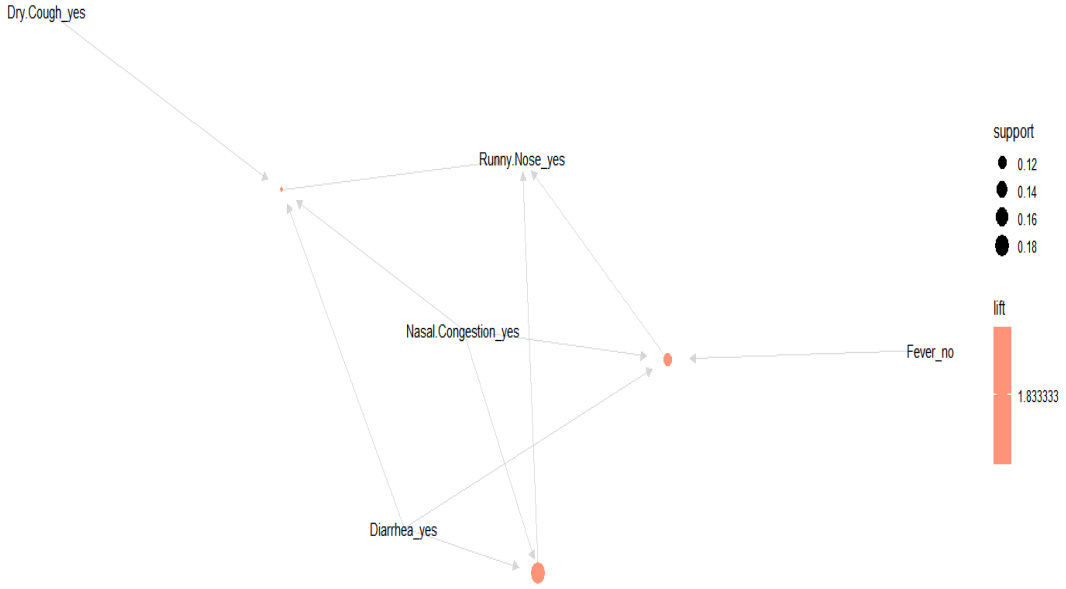
The following Symptoms are accompanied to each other

1) Difficulty in breathing is an accompanying symptom to sore throat and dry cough symptoms. It appears that these 3 symptoms mostly exist together. Whenever a person has these 3 symptoms together, there is a high probability that the person is infected with the virus and needs to take the appropriate action.



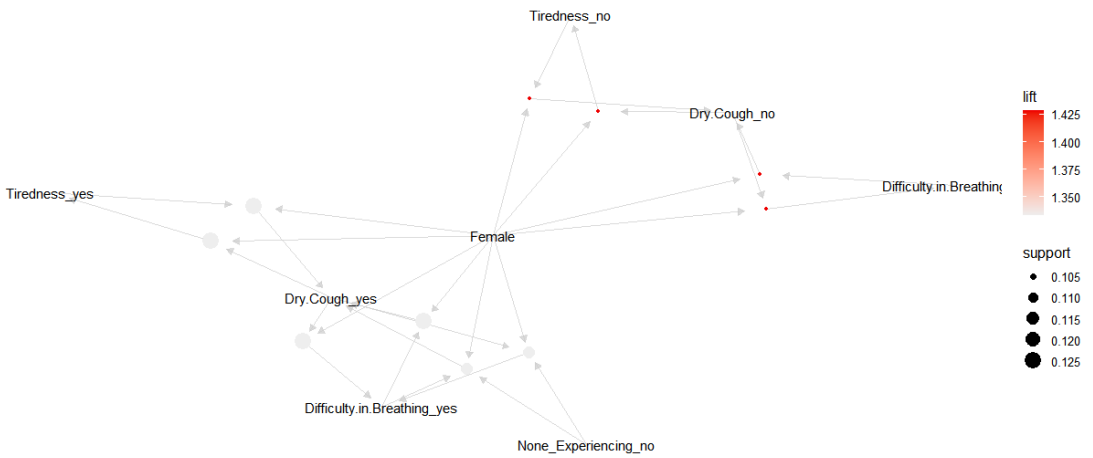
- This Graph shows the upper left circle that represents a rule that has sore throat and dry cough as in arrows (that means they are on the left side of rule) and difficulty in breathing as an out arrow(that means difficulty in breathing is on the right hand side of rule).

2) Runny nose is an accompanying symptom to nasal congestion and diarrhea symptoms as we can see in the below graphs.

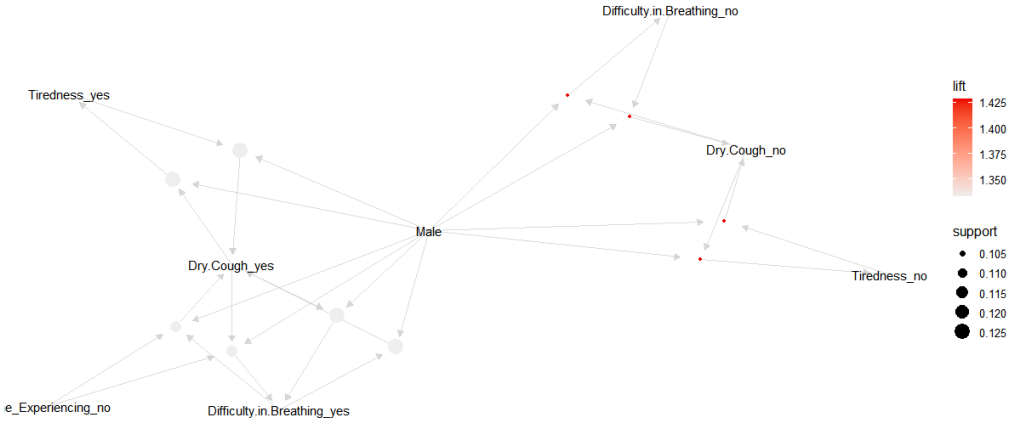


- The largest circle indicates the rule with greatest support and lift, It has Diarrhea and Nasal Congestion as input arrows to the circle (that means they are on the left side of the rule) and Runny nose as an output arrow from the circle (that means it is on right hand side of the rule).

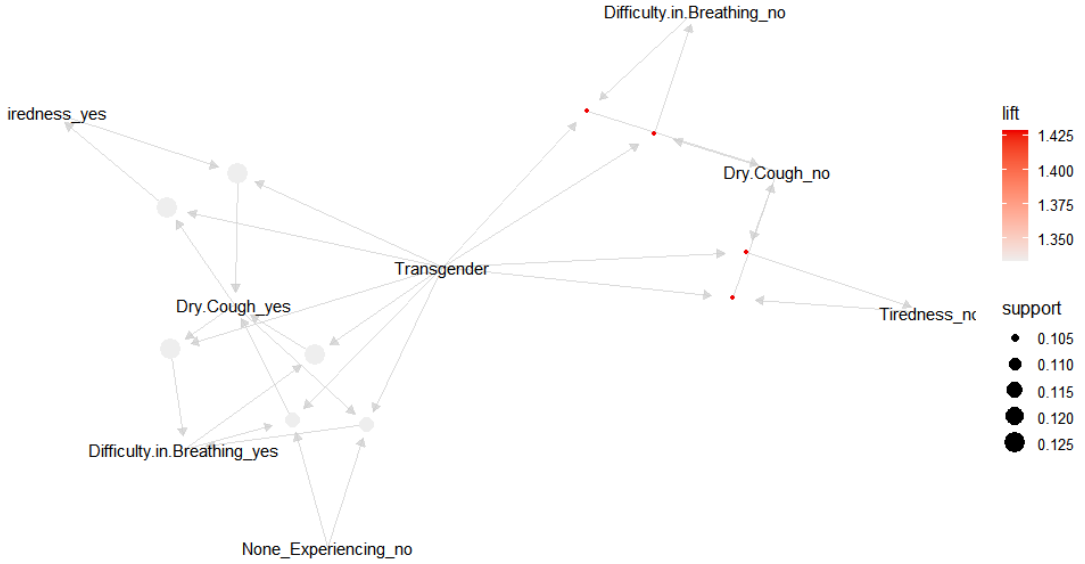
3) The gender of the patient doesn't make a difference in the symptoms that appear on him. We can see in the below 3 Graphs that Transgenders , males and females have exactly the same rules.
    - Generated Rules that has female on rhs



- Generated Rules that has Male on rhs

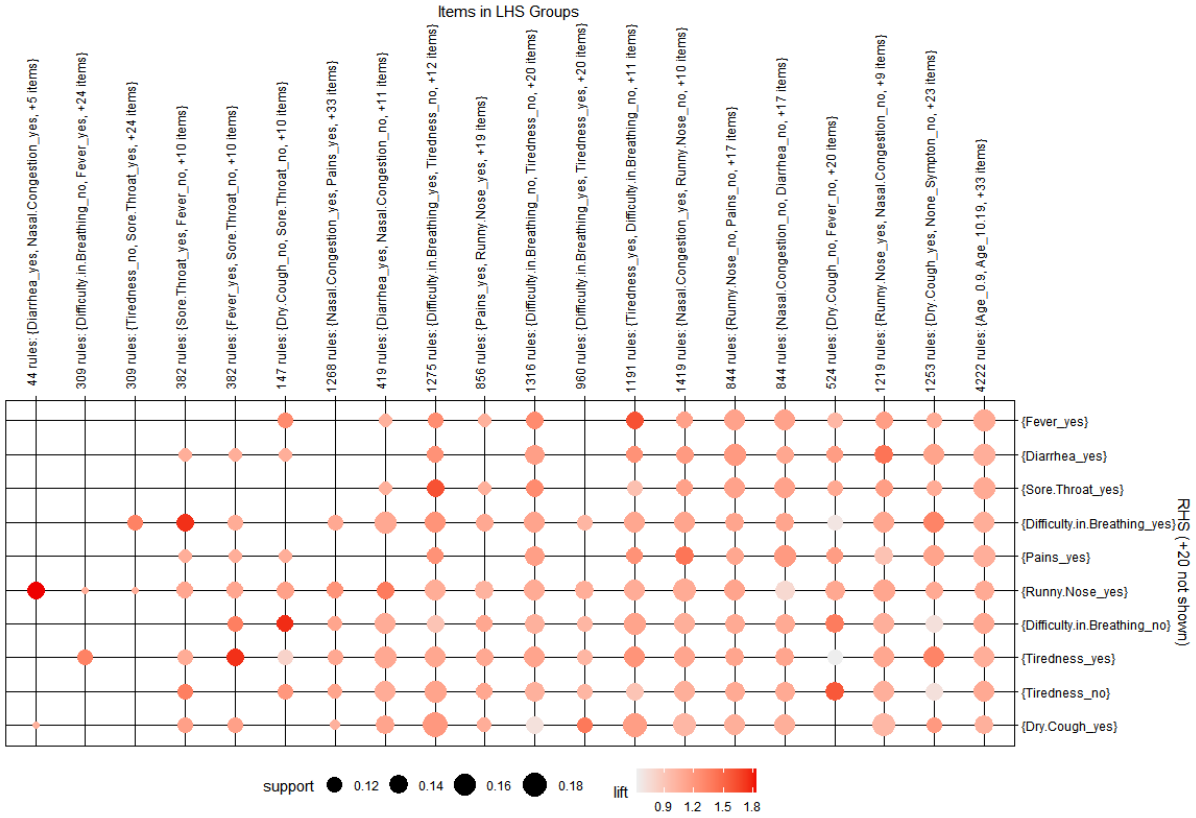


- Generated Rules that has Transgender on rhs



4)The dataset doesn't represent real cases it has equal distribution for any feature in different classes which means there is no discriminant features we can use differentiate between the classes

Rules Summarization



- The below graph summarizes the rules of our dataset. We can see from the graph that difficulty in breathing , Dry cough and Sore throat are accompanying each other because their rules have the highest lift, also Runny nose,nasal congestion and diarrhea are accompanying each other.

Future work / Enhancements

For future enhancements of the model, it is advised to enlarge the dataset with real life cases with their symptoms, so that the distribution of the classes is not constant for all categories, like in the dataset used. This will certainly increase the accuracy of the classifier and would help in a better prediction of the actions needed to be taken, based on how severe the case is.