capstone-project2

October 8, 2023

Capstone Project2:

Utilize machine learning approaches to predict the likelihood of a person contracting COVID-19 based on their individual information.

A speedy and accurate diagnosis of COVID-19 is made possible by effective SARS-CoV-2 screening, which can also lessen the burden on healthcare systems. There have been built prediction models that assess the likelihood of infection by combining a number of parameters. These are meant to help medical professionals all over the world treat patients, especially in light of the scarcity of healthcare resources. The current dataset has been downloaded from 'ABC' government website and contains around 2,78,848 individuals who have gone through the RT-PCR test. Data set contains 11 columns, including 8 features suspected to play an important role in the prediction of COVID19 outcome. Outcome variable is covid result test positive or negative. We have data from 11th March 2020 till 30th April 2020. Please consider 11th March till 15th April as a training and validation set. From 16th April till 30th April as a test set. Please further divide training and validation set at a ratio of 4:1.

The following list describes each of the dataset's features used by the model:

Features name:(corona_tested.csv)

1.ID(Individual ID)

2.Sex (male/female).

3.Age 60 above years (true/false)

4.Test date (date when tested for COVID)

B. Symptoms:

5. Cough (true/false).
6. Fever (true/false).
7. Sore throat (true/false).
8. Shortness of breath (true/false).
9. Headache (true/false).

C. Other information:

10. Known contact with an individual confirmed to have COVID-19 (true/false).

D. Covid report

11. Corona positive or negative

Instructions

Project Proposal for COVID19 prediction

Questions

Hypothesis

Approach

You will prepare a project proposal detailing the questions we are wanting to answer. The initial hypotheses about the data relationships and the approach you will take to get your answers.

Proposal is just a plan.

End goal is important

Section 1: Questions to Answer

What questions do you want to answer?

1Q)Why is your proposal important in today's world? How predicting a disease accurately can improve medical treatment?

Accurate prediction can enable early intervention and reduce the burden on healthcare systems.

Effective screening can enhance resource allocation and optimize medical treatment.

Predictive models contribute to proactive management of public health crises.

2Q)How is it going to impact the medical field when it comes to effective screening and reducing health care burden.

Early identification allows for timely treatment and isolation, reducing transmission.

Optimal resource allocation minimizes strain on healthcare resources.

Predictive models streamline screening processes for efficiency.

3Q)If any, what is the gap in the knowledge or how your proposed method can be helpful if required in the future for any other disease.

Identifying key features for COVID-19 prediction may provide insights for similar diseases.

Generalizable methods can be adapted for future disease outbreaks.

Section 2: Initial Hypothesis (or hypotheses)

2a)Assumptions based on Data Analysis (DA) Track:

Patterns in demographic data and pre-existing conditions may impact COVID-19 susceptibility.

Certain features, like travel history and exposure, are likely to influence transmission.

2b)Assumptions based on Machine Learning (ML) Track:

ML models will identify significant features impacting COVID-19 prediction.

The chosen model will outperform others based on relevant cost functions.

Section 3: Data Analysis Approach.

3a)Approach to Prove or Disprove Hypotheses:

Conduct exploratory data analysis (EDA) to identify correlations and patterns.

Validate assumptions through statistical analysis.

3b)Relevant Feature Engineering Techniques:

Creation of interaction features to capture combined effects.

Normalization and scaling for model interpretability.

3c)Justification of Data Analysis Approach:

EDA provides insights into potential relationships before model development.

Statistical analysis ensures robust hypothesis testing.

Section 4: Machine Learning Approach

4a)Method for Machine Learning Based Predictions of COVID-19:

Implement a combination of supervised learning models (e.g., logistic regression, decision trees, random forests, and SVM).

Utilize cross-validation to assess model performance.

4b)Justification of the Most Appropriate Model:

Model selection based on sensitivity, specificity, and area under the curve (AUC).

Consideration of interpretability and computational efficiency.

4c)Steps to Improve Accuracy:

Hyperparameter tuning using grid search or random search.

Feature selection based on model importance.

4d)Comparison of Models:

Evaluate performance metrics for each model.

Visualize results through ROC curves or precision-recall curves.

#Importing All Necessary Libraries:

[ ]:
```
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, confusion_matrix,␣
 ↪classification_report
```

#Importing corona_tested dataset:

```
[ ]: corona_tested = pd.read_csv('corona_tested.csv')
     corona_tested
```

```
<ipython-input-45-31ea5e6f3e0d>:1: DtypeWarning: Columns (2,3,4,5,6) have mixed
types. Specify dtype option on import or set low_memory=False.
  corona_tested = pd.read_csv('corona_tested.csv')
```

```
[ ]:         Ind_ID   Test_date Cough_symptoms  Fever Sore_throat  \
     0            1  11-03-2020           TRUE  FALSE        TRUE
     1            2  11-03-2020          FALSE   TRUE       FALSE
     2            3  11-03-2020          FALSE   TRUE       FALSE
     3            4  11-03-2020           TRUE  FALSE       FALSE
     4            5  11-03-2020           TRUE  FALSE       FALSE
     …          …          …             …     …           …
     278843  278844  30-04-2020          False  False       False
     278844  278845  30-04-2020          False  False       False
     278845  278846  30-04-2020          False  False       False
     278846  278847  30-04-2020          False  False       False
     278847  278848  30-04-2020          False  False       False

            Shortness_of_breath Headache    Corona Age_60_above     Sex  \
     0                    FALSE    FALSE  negative         None    None
     1                    FALSE    FALSE  positive         None    None
     2                    FALSE    FALSE  positive         None    None
     3                    FALSE    FALSE  negative         None    None
     4                    FALSE    FALSE  negative         None    None
     …                      …       …        …            …      …
     278843               False    False  positive         None    male
     278844               False    False  negative         None  female
     278845               False    False  negative         None    male
     278846               False    False  negative         None    male
     278847               False    False  negative         None  female

                      Known_contact
     0                       Abroad
     1                       Abroad
     2                       Abroad
     3                       Abroad
     4       Contact with confirmed
     …                          …
     278843                   Other
     278844                   Other
     278845                   Other
     278846                   Other
     278847                   Other
```

```
[278848 rows x 11 columns]
```

#Checking null values present in the corona_tested dataset columnwise:

```
[ ]: corona_tested.isnull().sum()
```

```
[ ]: Ind_ID                 0
     Test_date              0
     Cough_symptoms         0
     Fever                  0
     Sore_throat            0
     Shortness_of_breath    0
     Headache               0
     Corona                 0
     Age_60_above           0
     Sex                    0
     Known_contact          0
     dtype: int64
```

#Checking Non-Null Count and Datatype of each column present in the corona_tested dataset:

```
[ ]: corona_tested.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 278848 entries, 0 to 278847
Data columns (total 11 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   Ind_ID               278848 non-null  int64
 1   Test_date            278848 non-null  object
 2   Cough_symptoms       278848 non-null  object
 3   Fever                278848 non-null  object
 4   Sore_throat          278848 non-null  object
 5   Shortness_of_breath  278848 non-null  object
 6   Headache             278848 non-null  object
 7   Corona               278848 non-null  object
 8   Age_60_above         278848 non-null  object
 9   Sex                  278848 non-null  object
 10  Known_contact        278848 non-null  object
dtypes: int64(1), object(10)
memory usage: 23.4+ MB
```

#Checking Type of data present in each column:

```
[ ]: corona_tested.Ind_ID.unique()
```

```
[ ]: array([     1,      2,      3, …, 278846, 278847, 278848])
[ ]: corona_tested.Test_date.unique()
```

```
[ ]: array(['11-03-2020', '12-03-2020', '13-03-2020', '14-03-2020',
            '15-03-2020', '16-03-2020', '17-03-2020', '18-03-2020',
            '19-03-2020', '20-03-2020', '21-03-2020', '22-03-2020',
            '23-03-2020', '24-03-2020', '25-03-2020', '26-03-2020',
            '27-03-2020', '28-03-2020', '29-03-2020', '30-03-2020',
            '31-03-2020', '01-04-2020', '02-04-2020', '03-04-2020',
            '04-04-2020', '05-04-2020', '06-04-2020', '07-04-2020',
            '08-04-2020', '09-04-2020', '10-04-2020', '11-04-2020',
            '12-04-2020', '13-04-2020', '14-04-2020', '15-04-2020',
            '16-04-2020', '17-04-2020', '18-04-2020', '19-04-2020',
            '20-04-2020', '21-04-2020', '22-04-2020', '23-04-2020',
            '24-04-2020', '25-04-2020', '26-04-2020', '27-04-2020',
            '28-04-2020', '29-04-2020', '30-04-2020'], dtype=object)
```

```
[ ]: corona_tested.Cough_symptoms.unique()
```

```
[ ]: array(['TRUE', 'FALSE', 'None', False, True], dtype=object)
```

```
[ ]: corona_tested.Fever.unique()
```

```
[ ]: array(['FALSE', 'TRUE', 'None', False, True], dtype=object)
```

```
[ ]: corona_tested.Sore_throat.unique()
```

```
[ ]: array(['TRUE', 'FALSE', 'None', False, True], dtype=object)
```

```
[ ]: corona_tested.Shortness_of_breath.unique()
```

```
[ ]: array(['FALSE', 'TRUE', 'None', False, True], dtype=object)
```

```
[ ]: corona_tested.Headache.unique()
```

```
[ ]: array(['FALSE', 'TRUE', 'None', False, True], dtype=object)
```

```
[ ]: corona_tested.Corona.unique()
```

```
[ ]: array(['negative', 'positive', 'other'], dtype=object)
```

```
[ ]: corona_tested.Age_60_above.unique()
```

```
[ ]: array(['None', 'No', 'Yes'], dtype=object)
```

```
[ ]: corona_tested.Sex.unique()
```

```
[ ]: array(['None', 'male', 'female'], dtype=object)
[ ]: corona_tested.Known_contact.unique()
```

```
[ ]: array(['Abroad', 'Contact with confirmed', 'Other'], dtype=object)
```

-There are ‘None’ type of data present in these columns. Cough_symptoms,Fever,Sore_throat,Shortness_of_breath,Headache,Age_60_above,Sex.’None’ is used to represent the absence of a value or a null value in Python.so first you can replace ‘none’ with ‘NA’.In Pandas, pd.NA represents a missing value or NA (Not Available). It is similar to None or np.nan but provides more consistency and functionality, especially in the context of Pandas DataFrames.

-There are ‘other’ type of data present in the columns Corona,Known_contact. so first you can replace ‘other’ with ‘NA’.which is a common practice when dealing with missing or undefined values in numerical data. This makes it easier to handle missing values using functions and methods provided by libraries like NumPy and Pandas.

#Replace ‘None’ and ‘other’ with ‘NA’

```
[ ]: corona_tested.replace('None', pd.NA, inplace=True)
```

```
[ ]: corona_tested['Corona'] = corona_tested['Corona'].replace('other', np.nan)
```

```
[ ]: corona_tested['Known_contact'] = corona_tested['Known_contact'].
      ↪replace('Other', np.nan)
```

#checking, is ‘NA’ is replaced or not?

```
[ ]: corona_tested.Cough_symptoms.unique()
```

```
[ ]: array(['TRUE', 'FALSE', <NA>, False, True], dtype=object)
```

```
[ ]: corona_tested.Fever.unique()
```

```
[ ]: array(['FALSE', 'TRUE', <NA>, False, True], dtype=object)
```

```
[ ]: corona_tested.Sore_throat.unique()
```

```
[ ]: array(['TRUE', 'FALSE', <NA>, False, True], dtype=object)
```

```
[ ]: corona_tested.Shortness_of_breath.unique()
```

```
[ ]: array(['FALSE', 'TRUE', <NA>, False, True], dtype=object)
```

```
[ ]: corona_tested.Headache.unique()
```

```
[ ]: array(['FALSE', 'TRUE', <NA>, False, True], dtype=object)
```

```
[ ]: corona_tested.Corona.unique()
[ ]: array(['negative', 'positive', nan], dtype=object)
```

```
[ ]: corona_tested.Age_60_above.unique()
```

```
[ ]: array([<NA>, 'No', 'Yes'], dtype=object)
```

```
[ ]: corona_tested.Sex.unique()
```

```
[ ]: array([<NA>, 'male', 'female'], dtype=object)
```

```
[ ]: corona_tested.Known_contact.unique()
```

```
[ ]: array(['Abroad', 'Contact with confirmed', nan], dtype=object)
```

#filling Null values with Mode value:

```
[ ]: corona_tested['Cough_symptoms'].fillna(corona_tested['Cough_symptoms'].
      ↪mode()[0], inplace=True)
```

```
[ ]: corona_tested['Fever'].fillna(corona_tested['Fever'].mode()[0], inplace=True)
```

```
[ ]: corona_tested['Sore_throat'].fillna(corona_tested['Sore_throat'].mode()[0],␣
      ↪inplace=True)
```

```
[ ]: corona_tested['Shortness_of_breath'].
      ↪fillna(corona_tested['Shortness_of_breath'].mode()[0], inplace=True)
```

```
[ ]: corona_tested['Headache'].fillna(corona_tested['Headache'].mode()[0],␣
      ↪inplace=True)
```

```
[ ]: corona_tested['Corona'].fillna(corona_tested['Corona'].mode()[0], inplace=True)
```

```
[ ]: corona_tested['Age_60_above'].fillna(corona_tested['Age_60_above'].mode()[0],␣
      ↪inplace=True)
```

```
[ ]: corona_tested['Sex'].fillna(corona_tested['Sex'].mode()[0], inplace=True)
```

```
[ ]: corona_tested['Known_contact'].fillna(corona_tested['Known_contact'].mode()[0],␣
      ↪inplace=True)
```

#Again checking to see,is Null values are replaced with mode value:

```
[ ]: corona_tested.Cough_symptoms.unique()
```

```
[ ]: array(['TRUE', 'FALSE', False, True], dtype=object)
```

```
[ ]: corona_tested.Fever.unique()
[ ]: array(['FALSE', 'TRUE', False, True], dtype=object)
```

```
[ ]: corona_tested.Sore_throat.unique()
```

```
[ ]: array(['TRUE', 'FALSE', False, True], dtype=object)
```

```
[ ]: corona_tested.Shortness_of_breath.unique()
```

```
[ ]: array(['FALSE', 'TRUE', False, True], dtype=object)
```

```
[ ]: corona_tested.Headache.unique()
```

```
[ ]: array(['FALSE', 'TRUE', False, True], dtype=object)
```

```
[ ]: corona_tested.Corona.unique()
```

```
[ ]: array(['negative', 'positive'], dtype=object)
```

```
[ ]: corona_tested.Age_60_above.unique()
```

```
[ ]: array(['No', 'Yes'], dtype=object)
```

```
[ ]: corona_tested.Sex.unique()
```

```
[ ]: array(['female', 'male'], dtype=object)
```

```
[ ]: corona_tested.Known_contact.unique()
```

```
[ ]: array(['Abroad', 'Contact with confirmed'], dtype=object)
```

#After performing fillna() operation again checking is there any null values present in columns:

```
[ ]: corona_tested.isnull().sum()
```

```
[ ]: Ind_ID                 0
     Test_date              0
     Cough_symptoms         0
     Fever                  0
     Sore_throat            0
     Shortness_of_breath    0
     Headache               0
     Corona                 0
     Age_60_above           0
     Sex                    0
     Known_contact          0
     dtype: int64
```

#Selecting Indepenent variables:

```
[ ]: X =␣
      ↪corona_tested[['Test_date','Cough_symptoms','Fever','Sore_throat','Shortness_of_breath','He
     X
```

```
[ ]:         Test_date Cough_symptoms  Fever Sore_throat Shortness_of_breath  \
     0       11-03-2020           TRUE  FALSE        TRUE               FALSE
     1       11-03-2020          FALSE   TRUE       FALSE               FALSE
     2       11-03-2020          FALSE   TRUE       FALSE               FALSE
     3       11-03-2020           TRUE  FALSE       FALSE               FALSE
     4       11-03-2020           TRUE  FALSE       FALSE               FALSE
     …            …              …     …         …                 …
     278843  30-04-2020          False  False       False               False
     278844  30-04-2020          False  False       False               False
     278845  30-04-2020          False  False       False               False
     278846  30-04-2020          False  False       False               False
     278847  30-04-2020          False  False       False               False

            Headache Age_60_above     Sex           Known_contact
     0         FALSE           No  female                  Abroad
     1         FALSE           No  female                  Abroad
     2         FALSE           No  female                  Abroad
     3         FALSE           No  female                  Abroad
     4         FALSE           No  female  Contact with confirmed
     …         …            …     …                  …
     278843    False           No    male                  Abroad
     278844    False           No  female                  Abroad
     278845    False           No    male                  Abroad
     278846    False           No    male                  Abroad
     278847    False           No  female                  Abroad

     [278848 rows x 9 columns]
```

#Selecting Target Variable:

```
[ ]: y = corona_tested[['Corona']]
     y
```

```
[ ]:           Corona
     0       negative
     1       positive
     2       positive
     3       negative
     4       negative
     …          …
     278843  positive
     278844  negative
     278845  negative
```

```
278846  negative
278847  negative

[278848 rows x 1 columns]
```

#Performing Dummy Encoding for Categorical data columns:

```
[ ]: X = pd.get_dummies(X,columns =␣
     ↪['Test_date','Cough_symptoms','Fever','Sore_throat','Shortness_of_breath','Headache','Age_6
     X
```

```
[ ]:        Test_date_01-04-2020  Test_date_02-04-2020  Test_date_03-04-2020  \
     0                         0                     0                     0
     1                         0                     0                     0
     2                         0                     0                     0
     3                         0                     0                     0
     4                         0                     0                     0
     …                       …                   …                   …
     278843                    0                     0                     0
     278844                    0                     0                     0
     278845                    0                     0                     0
     278846                    0                     0                     0
     278847                    0                     0                     0

            Test_date_04-04-2020  Test_date_05-04-2020  Test_date_06-04-2020  \
     0                         0                     0                     0
     1                         0                     0                     0
     2                         0                     0                     0
     3                         0                     0                     0
     4                         0                     0                     0
     …                       …                   …                   …
     278843                    0                     0                     0
     278844                    0                     0                     0
     278845                    0                     0                     0
     278846                    0                     0                     0
     278847                    0                     0                     0

            Test_date_07-04-2020  Test_date_08-04-2020  Test_date_09-04-2020  \
     0                         0                     0                     0
     1                         0                     0                     0
     2                         0                     0                     0
     3                         0                     0                     0
     4                         0                     0                     0
     …                       …                   …                   …
     278843                    0                     0                     0
     278844                    0                     0                     0
     278845                    0                     0                     0
```

```
278846                    0                    0                    0
278847                    0                    0                    0

        Test_date_10-04-2020  …  Headache_False  Headache_True  \
0                          0  …               0              0
1                          0  …               0              0
2                          0  …               0              0
3                          0  …               0              0
4                          0  …               0              0
…                        …  …             …            …
278843                     0  …               1              0
278844                     0  …               1              0
278845                     0  …               1              0
278846                     0  …               1              0
278847                     0  …               1              0

        Headache_FALSE  Headache_TRUE  Age_60_above_No  Age_60_above_Yes  \
0                    1              0                1                 0
1                    1              0                1                 0
2                    1              0                1                 0
3                    1              0                1                 0
4                    1              0                1                 0
…                  …            …              …               …
278843               0              0                1                 0
278844               0              0                1                 0
278845               0              0                1                 0
278846               0              0                1                 0
278847               0              0                1                 0

        Sex_female  Sex_male  Known_contact_Abroad  \
0                1         0                     1
1                1         0                     1
2                1         0                     1
3                1         0                     1
4                1         0                     0
…              …       …                   …
278843           0         1                     1
278844           1         0                     1
278845           0         1                     1
278846           0         1                     1
278847           1         0                     1

        Known_contact_Contact with confirmed
0                                          0
1                                          0
2                                          0
3                                          0
```

```
4                                              1
…                                         …
278843                                         0
278844                                         0
278845                                         0
278846                                         0
278847                                         0

[278848 rows x 77 columns]
```

#Split the dataset into X_train, X_test, y_train, y_test:

```
[ ]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25,␣
      ↪random_state=0)
```

```
[ ]: X_train
```

```
[ ]:         Test_date_01-04-2020  Test_date_02-04-2020  Test_date_03-04-2020  \
     101132                     0                     0                     0
     248238                     0                     0                     0
     227259                     0                     0                     0
     153806                     0                     0                     0
     188607                     0                     0                     0
     …                       …                     …                     …
     211543                     0                     0                     0
     86293                      0                     0                     1
     122579                     0                     0                     0
     152315                     0                     0                     0
     117952                     0                     0                     0

             Test_date_04-04-2020  Test_date_05-04-2020  Test_date_06-04-2020  \
     101132                     0                     1                     0
     248238                     0                     0                     0
     227259                     0                     0                     0
     153806                     0                     0                     0
     188607                     0                     0                     0
     …                       …                     …                     …
     211543                     0                     0                     0
     86293                      0                     0                     0
     122579                     0                     0                     0
     152315                     0                     0                     0
     117952                     0                     0                     0

             Test_date_07-04-2020  Test_date_08-04-2020  Test_date_09-04-2020  \
     101132                     0                     0                     0
     248238                     0                     0                     0
     227259                     0                     0                     0
```

```
153806                        0                       0                       0
188607                        0                       0                       0
…                      …                    …                    …
211543                        0                       0                       0
86293                         0                       0                       0
122579                        0                       0                       1
152315                        0                       0                       0
117952                        0                       1                       0

        Test_date_10-04-2020  …  Headache_False  Headache_True  \
101132                     0  …               1              0
248238                     0  …               1              0
227259                     0  …               1              0
153806                     0  …               1              0
188607                     0  …               1              0
…                        … …             …            …
211543                     0  …               1              0
86293                      0  …               1              0
122579                     0  …               1              0
152315                     0  …               1              0
117952                     0  …               1              0

        Headache_FALSE  Headache_TRUE  Age_60_above_No  Age_60_above_Yes  \
101132               0              0                1                 0
248238               0              0                1                 0
227259               0              0                1                 0
153806               0              0                1                 0
188607               0              0                1                 0
…                  …            …              …               …
211543               0              0                1                 0
86293                0              0                1                 0
122579               0              0                1                 0
152315               0              0                1                 0
117952               0              0                1                 0

        Sex_female  Sex_male  Known_contact_Abroad  \
101132           0         1                     1
248238           0         1                     1
227259           0         1                     1
153806           1         0                     1
188607           0         1                     1
…              …       …                   …
211543           1         0                     1
86293            0         1                     1
122579           0         1                     1
152315           1         0                     1
117952           0         1                     1
```

```
        Known_contact_Contact with confirmed
101132                                     0
248238                                     0
227259                                     0
153806                                     0
188607                                     0
…                                        …
211543                                     0
86293                                      0
122579                                     0
152315                                     0
117952                                     0

[209136 rows x 77 columns]
```

[ ]:
```
X_test
```

[ ]:
```
        Test_date_01-04-2020  Test_date_02-04-2020  Test_date_03-04-2020  \
229115                     0                     0                     0
181871                     0                     0                     0
219603                     0                     0                     0
138213                     0                     0                     0
207510                     0                     0                     0
…                        …                     …                     …
63073                      0                     0                     0
207324                     0                     0                     0
150511                     0                     0                     0
146770                     0                     0                     0
80249                      0                     1                     0

        Test_date_04-04-2020  Test_date_05-04-2020  Test_date_06-04-2020  \
229115                     0                     0                     0
181871                     0                     0                     0
219603                     0                     0                     0
138213                     0                     0                     0
207510                     0                     0                     0
…                        …                     …                     …
63073                      0                     0                     0
207324                     0                     0                     0
150511                     0                     0                     0
146770                     0                     0                     0
80249                      0                     0                     0

        Test_date_07-04-2020  Test_date_08-04-2020  Test_date_09-04-2020  \
229115                     0                     0                     0
181871                     0                     0                     0
```

```
219603                    0                    0                    0
138213                    0                    0                    0
207510                    0                    0                    0
…                       …                    …                    …
63073                     0                    0                    0
207324                    0                    0                    0
150511                    0                    0                    0
146770                    0                    0                    0
80249                     0                    0                    0

        Test_date_10-04-2020  …  Headache_False  Headache_True  \
229115                     0  …               1              0
181871                     0  …               1              0
219603                     0  …               1              0
138213                     0  …               1              0
207510                     0  …               1              0
…                        …  …             …            …
63073                      0  …               0              0
207324                     0  …               1              0
150511                     0  …               1              0
146770                     0  …               1              0
80249                      0  …               1              0

        Headache_FALSE  Headache_TRUE  Age_60_above_No  Age_60_above_Yes  \
229115               0              0                1                 0
181871               0              0                1                 0
219603               0              0                1                 0
138213               0              0                1                 0
207510               0              0                1                 0
…                  …            …              …               …
63073                1              0                1                 0
207324               0              0                1                 0
150511               0              0                1                 0
146770               0              0                1                 0
80249                0              0                0                 1

        Sex_female  Sex_male  Known_contact_Abroad  \
229115           0         1                     1
181871           0         1                     1
219603           0         1                     1
138213           0         1                     1
207510           1         0                     1
…              …       …                   …
63073            0         1                     0
207324           1         0                     1
150511           0         1                     1
146770           1         0                     1
```

```
80249            1         0                      1

        Known_contact_Contact with confirmed
229115                                     0
181871                                     0
219603                                     0
138213                                     0
207510                                     0
…                                      …
63073                                      1
207324                                     0
150511                                     0
146770                                     0
80249                                      0

[69712 rows x 77 columns]
```

```
[ ]: y_train
```

```
[ ]:           Corona
     101132  negative
     248238  negative
     227259  negative
     153806  negative
     188607  negative
     …          …
     211543  negative
     86293   negative
     122579  positive
     152315  negative
     117952  negative

     [209136 rows x 1 columns]
```

```
[ ]: y_test
```

```
[ ]:           Corona
     229115  negative
     181871  negative
     219603  negative
     138213  negative
     207510  negative
     …          …
     63073   positive
     207324  negative
     150511  negative
     146770  negative
```

```
80249   negative

[69712 rows x 1 columns]
```

#1)Model1:LogisticRegression:

```
[ ]: # Create a logistic regression model instance
     model = LogisticRegression()

     #Train the model using the training sets
     model.fit(X_train, y_train)

     #Predict the response for test dataset
     y_pred = model.predict(X_test)

     #accuracy_score,confusion_matrix,classification_report
     accuracy = accuracy_score(y_test, y_pred)
     conf_matrix = confusion_matrix(y_test, y_pred)
     class_report = classification_report(y_test, y_pred)

     print("Accuracy:", accuracy)
     print("Confusion Matrix:\n", conf_matrix)
     print("Classification Report:\n", class_report)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logistic.py:458:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  n_iter_i = _check_optimize_result(
```

```
Accuracy: 0.9642816157906816
Confusion Matrix:
 [[65399   619]
 [ 1871  1823]]
Classification Report:
               precision    recall  f1-score   support

    negative       0.97      0.99      0.98     66018
    positive       0.75      0.49      0.59      3694

    accuracy                           0.96     69712
   macro avg       0.86      0.74      0.79     69712
weighted avg       0.96      0.96      0.96     69712
```

#2)Model2:Decision Tree model:

[ ]:
```
# Initialize the Decision Tree model
model = DecisionTreeClassifier(random_state=42)

# Fit the model to the training data
model.fit(X_train, y_train)

# Make predictions on the testing data
y_pred = model.predict(X_test)

#accuracy_score,confusion_matrix,classification_report
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
class_report = classification_report(y_test, y_pred)

print("Accuracy:", accuracy)
print("Confusion Matrix:\n", conf_matrix)
print("Classification Report:\n", class_report)
```

```
Accuracy: 0.9687858618315355
Confusion Matrix:
 [[65488   530]
 [ 1646  2048]]
Classification Report:
               precision    recall  f1-score   support

    negative       0.98      0.99      0.98     66018
    positive       0.79      0.55      0.65      3694

    accuracy                           0.97     69712
   macro avg       0.88      0.77      0.82     69712
weighted avg       0.97      0.97      0.97     69712
```

#3)MODEL3:RandomForest

[ ]:
```
#Fitting Decision Tree classifier to the training set
from sklearn.ensemble import RandomForestClassifier
classifier= RandomForestClassifier(n_estimators= 9, criterion="entropy")

#Train the model using the training sets
```

```
classifier.fit(X_train, y_train)

#Predicting the test set result
y_pred= classifier.predict(X_test)

#accuracy_score,confusion_matrix,classification_report
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
class_report = classification_report(y_test, y_pred)

print("Accuracy:", accuracy)
print("Confusion Matrix:\n", conf_matrix)
print("Classification Report:\n", class_report)
```

```
<ipython-input-106-26c6efd8267f>:6: DataConversionWarning: A column-vector y was
passed when a 1d array was expected. Please change the shape of y to
(n_samples,), for example using ravel().
  classifier.fit(X_train, y_train)
```

```
Accuracy: 0.9691301354142758
Confusion Matrix:
 [[65455   563]
 [ 1589  2105]]
Classification Report:
               precision    recall  f1-score   support

    negative       0.98      0.99      0.98     66018
    positive       0.79      0.57      0.66      3694

    accuracy                           0.97     69712
   macro avg       0.88      0.78      0.82     69712
weighted avg       0.97      0.97      0.97     69712
```

#4)MODEL4:GRADIANT BOOSTING

```
[ ]: from sklearn.ensemble import GradientBoostingClassifier
     from sklearn.model_selection import train_test_split
     from sklearn.metrics import accuracy_score

     # Create a Gradient Boosting Classifier instance with 100 trees and a fixed␣
      ↪random seed
     model = GradientBoostingClassifier(n_estimators=100, random_state=42)

     #Train the model using the training sets
     model.fit(X_train, y_train)

     #Predicting the test set result
```

```
y_pred = model.predict(X_test)

#accuracy_score,confusion_matrix,classification_report
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
class_report = classification_report(y_test, y_pred)

print("Accuracy:", accuracy)
print("Confusion Matrix:\n", conf_matrix)
print("Classification Report:\n", class_report)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_gb.py:437:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

```
Accuracy: 0.9686567592380078
Confusion Matrix:
 [[65405   613]
 [ 1572  2122]]
Classification Report:
               precision    recall  f1-score   support

    negative       0.98      0.99      0.98     66018
    positive       0.78      0.57      0.66      3694

    accuracy                           0.97     69712
   macro avg       0.88      0.78      0.82     69712
weighted avg       0.97      0.97      0.97     69712
```

```
[ ]: print('LogisticRegression:accuracy = 96\nDecision Tree model:accuracy =␣
      ↪97\nRandomForestClassifier:accuracy = 97\nGradientBoostingClassifier:
      ↪accuracy = 97')
```

```
LogisticRegression:accuracy = 96
Decision Tree model:accuracy = 97
RandomForestClassifier:accuracy = 97
GradientBoostingClassifier:accuracy = 97
```

#download the cleaned 'corona_tested' file for performing SQL operation:

```
[ ]: corona_tested.to_csv('corona_tested.csv', index=False)
     # importing file from a local folder
     from google.colab import files
     files.download('corona_tested.csv')
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.Javascript object>
```