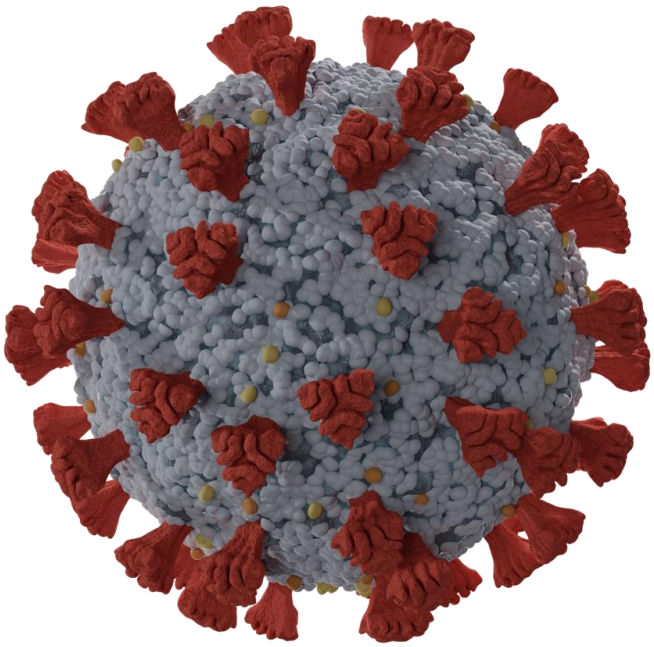
CMPE 255

RAKSHA

(Using supervised machine-learning algorithms to guess the covid-19 test result by analyzing symptoms)



Team Members

KOMMULA RAJASHEKAR REDDY - 016006211
SHRAVANI PARSI - 016003559
VISHWESH SHAH - 015971527
PRATYUSHA MANCHALLA - 015231866

I. Abstract

In December 2019, a highly contagious and deadly disease was detected – Coronavirus (COVID-19). This has affected 212 countries worldwide.

With the world's deadliest coronavirus (COVID-19) epidemic presently underway, one thing is certain: everyone can assist our country contain the COVID-19 epidemic. Wearing a mask, washing your hands, maintaining physical distance, and avoiding large indoor gatherings are all proven public health practices that not only reduce our own risk of infection by SARS-CoV-2 (the virus that causes coronavirus disease, or COVID-19), but also prevent COVID-19 from spreading to our coworkers, friends, and loved ones. Testing as many persons as possible will also be beneficial.

COVID-19 testing is so critical that the NIH started the Quick Acceleration of Diagnostics (RADx) Initiative in April 2020 to produce rapid, easy-to-use, accurate testing and make it available nationally. The RADx Underserved People (RADx-UP) initiative is part of this endeavor to identify ways to stem the spread of COVID-19, especially among racial and ethnic minorities and other vulnerable populations who have been disproportionately affected by this epidemic.

With the help of the concepts of data mining, we are predicting the covid 19 result against the lab result using Supervised Machine Learning Algorithms.

II. Introduction

Data mining helps to excerpt any concealed, potential data from the dataset. A traditional data mining process involves – “clear business problems, data integration, data extraction, data conversion, data cleaning, data analysis, result evaluation, data application [3]”.

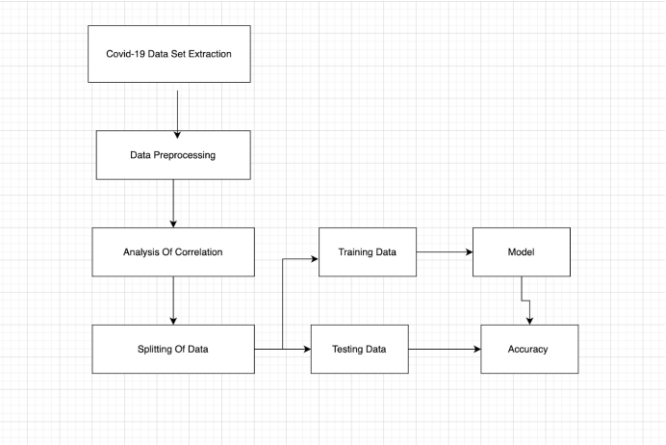
In this research, we intend to develop a model that uses supervised machine learning methods to predict the presence of the COVID-19 virus in a person. Based on the symptoms, we are forecasting the outcome of the covid-19 lab result. We propose to employ the covid antigen test lab findings as our targeted variable utilizing symptoms attributes. "In the medical field, machine learning may be used to identify illnesses with great accuracy [2]."

We used the following supervised learning algorithms:

1. Decision Tree
2. Random Forest
3. Logistic Regression
4. K-nearest neighbor (KNN)
5. SVM (Support Vector Machine)
6. Naive Bayes

III. Data exploration & Data processing

Data exploration, collection are the first steps of data mining. It is important to collect relevant data. But usually, the collected will be loose i.e., loosely controlled and this gives different values and combinations of data. Hence, it is very important to remove irrelevant and redundant information in the data before proceeding with analysis.



Thus comes the significance of data preprocessing, which “includes data preparation, compounded by integration, cleaning, normalization and

transformation of data; and data reduction tasks; such as feature selection, instance selection, discretization, etc.".

We explored multiple sources to collect dataset that are useful to predict the presence of Covid19 in a person's body based on the symptoms found in lab result. "https://www.gob.mx/salud/documentos/datos-abiertos-152127" data set is finalized and studied. The columns in the dataset are processed.

The following steps are followed as a part of initial analysis and data preprocessing in this project: The initial dataset consisted of 35 columns related to various information like symptoms date, patient type, incubation, pneumonia, ICU, results and other columns that indicate the medical conditions of the patient.

- Certain columns are renamed according to their meaning.
- Columns that are redundant for the analysis are dropped.
- Similarly, the values that are not needed are dropped from the dataset.
- Values in certain columns are replaced with numbers wherever applicable.
- After correlation analysis, few columns that are not required for analysis are dropped/removed.

IV. Supervised Machine Learning Algorithms Implementation

The increase in data increases the importance of data analysis as mentioned in [1]. In such scenarios, supervised machine learning algorithms help us to produce fast and efficient solutions. Humans learn from the past experiences. But a computer does not have something like "experiences" to learn from. Hence, it utilizes data to learn. Algorithms will be applied on the data that the computer/system has and some results will be predicted based on it. The data can contain discrete class attributes, for examples, high or low risk, positive or negative, loan approved or not approved etc.

In this project, we are analyzing Covid 19 dataset. In the dataset we have various attributes related to patient's medical history like diabetes, asthma, hypertension, chronic kidney, smoking etc. For a person detected with covid, this data will be useful to draw an analysis on the impact the virus has on the patient based on the patient's medical history.

Supervised machine learning consists of two important categories – data and goal. A dataset contains huge set of records of data.

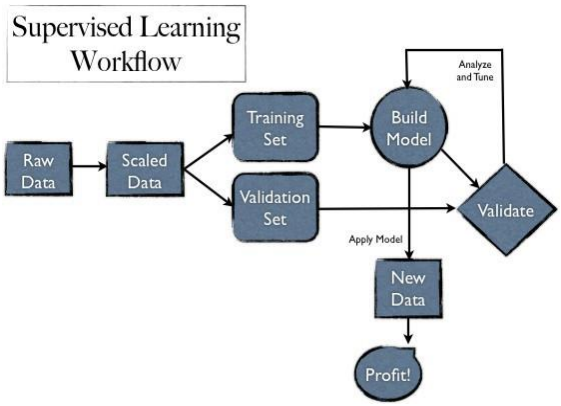
Steps:

The following steps are followed in the project for predicting the output.

- The preprocessed data is divided into training and testing data. Certain percentage of entire dataset will be considered as training dataset and the rest as testing dataset. In this project, 40% of the dataset is considered as test dataset and 60% as training dataset.
- The training data is sent to a learning algorithm
- A model is created based on the learning algorithm and the model predicts the outcomes.
- Test data is applied on the model
- Accuracy is calculated

Terminology used in the project:

GridSearchCV: This is a library function and



is a part of sklearn.model_selection package. It helps us to go through the hyperparameters that are predefined and

fits the model on the training dataset. It provides best hyper parameters that can be used for the selected model. To achieve this, grid parameters are set corresponding to the model. It helps us to train the model with parameters that fit best to the data.

Confusion Matrix: It is in a tabular form. The matrix is also known as error matrix. It is used to measure and visualize the performance of a classification algorithm.

ROC Curve: It is "Receiver Operating Characteristic" curve. It is a graphical plot that shows the performance of a classification model. It is a famous method used to measure accuracy. If the curve falls under top left of the line, then it is indicated as a good performance or else as a bad performance.

We are using the following supervised learning algorithms in this project:

- Decision Tree
- Random Forest
- Logistic Regression
- K-nearest neighbor (KNN)
- SVM (Support Vector Machine)
- Naive Bayes

Logistic Regression

A hyperplane is a plane whose number of dimensions is one less than its ambient space. For example, a 2D plane is a hyperplane for a 3D space, while a 1D plane (a line) is a hyperplane for a 2D space. Logistic regression is one of the basic, popular and frequently used linear statistical algorithm. It is a supervised machine learning algorithm "where each data has a label [2]". It is different to linear regression based on the type of problem that is addressed. Linear regression addresses regression problems while logistic solves classification problems. Classification means predicting a class or a label. It is a machine learning model that

predicts a dependent variable for a given set of independent variables. Here the dependent variable is categorical. It is a machine learning model that uses a hyperplane in a dimensional space. It separates data points with number of features into their classes.

In this project, we are patenting basic details, symptom and various medical conditions of patients as independent variables and Covid result acts as the dependent variable. The following steps are followed to apply this algorithm for the covid dataset:

- Logistic Regression model is created

```
model_lg = Pipeline([
    ('pca', PCA()),
    ('logreg', LogisticRegression())
])
```

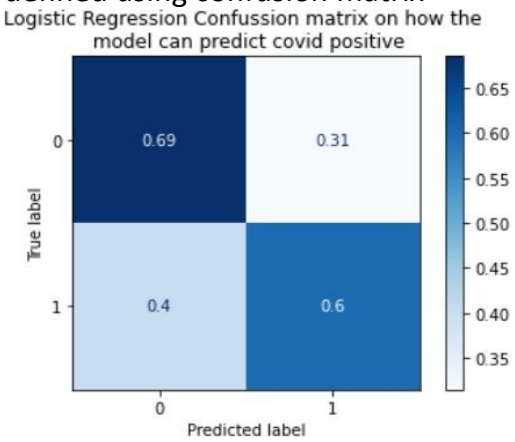
- GridSearchCV is used to get the best hyper parameters. It resulted in the following values shown in the figure.
- Here, C is the inversion regularization parameter. It has to be a positive float. Usually, small values indicate strong regularization.
- Solver is used to solve optimization problems in hyper planes. Different solver available are 'liblinear', 'newton-cg', , 'sag' 'lbfgs'. For small

datasets, liblinear is a better choice and sag is used for large ones as it is fast. The other two can handle multiclass problems.

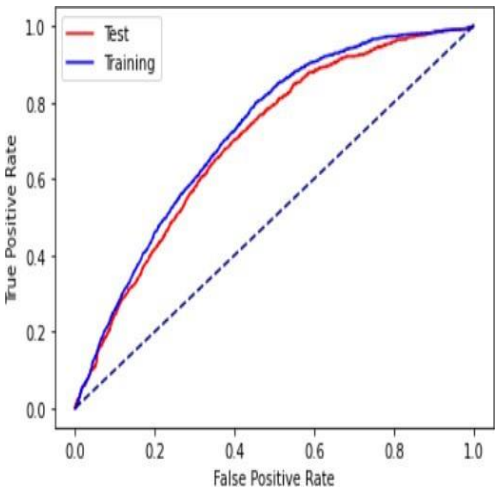
- Validation and test score are computed.

```
model_lg = Pipeline([
    ('pca', PCA()),
    ('logreg', LogisticRegression())
])
```

- Prediction is also done using test dataset.
- The performance of the model is defined using confusion matrix



- ROC curve is drawn to display the performance.



Decision Tree

It is a binary tree and a supervised machine learning algorithm. It's structure is like a flow chart. It recursively splits the dataset until we are left with pure leaf nodes. Pure leaf node imply data with same type of class. In the decision tree, we have 2 types of nodes –decision nodes and terminal or leaf nodes. Decision node contain the condition that is

useful to split the data. The leaf nodes contain a class. The data is split based on a certain feature/parameter. Best fit is found by maximizing the entropy gain.

Decision trees are used for both regression and classification problems. The following steps are followed in the project:

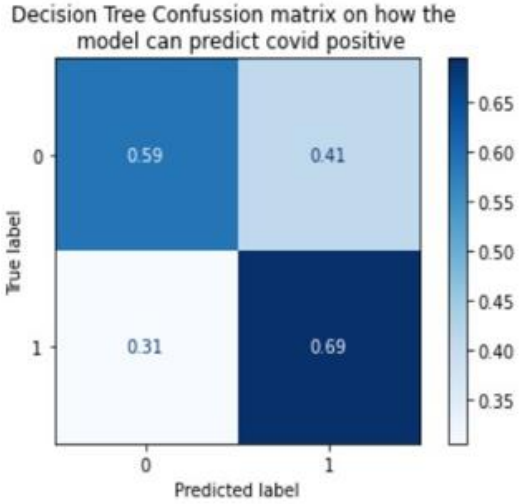
- A model is created

```
model_dt = Pipeline([('dt', DecisionTreeClassifier())]
)
```

- Some predefined hyper parameters are set and passed to GridSearchCV
- Best parameter values are obtained from GridSearchCV

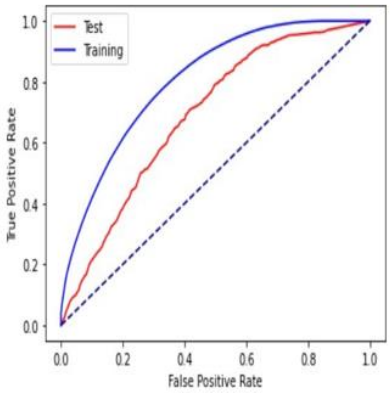
```
Pipeline(steps=[('dt',
                 DecisionTreeClassifier(class_weight='balanced', max_depth=20,
                                        min_samples_leaf=2,
                                        min_samples_split=50))])
```

- Validation and test score are computed.
- Prediction is also done using test dataset.
- The performance of the model is defined using confusion matrix



- ROC curve is drawn to display the

performance.



Random Forest

It is a collection of multiple random decision trees. Decision trees have high variance as they are highly sensitive to the training data. Random forest is less sensitive to the training data. It fits decision trees on sub-samples of the dataset. It averages to resolve the problem of overfitting and to improve the accuracy.

The following steps are followed in the project for this model:

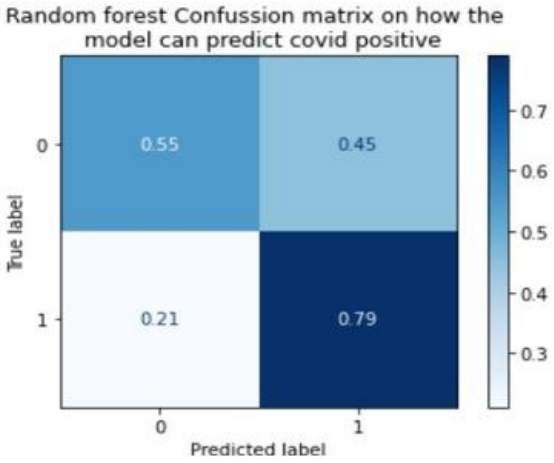
- A model is created.

```
rf_pipeline = Pipeline([
    ('rf', RandomForestClassifier())]
)
```

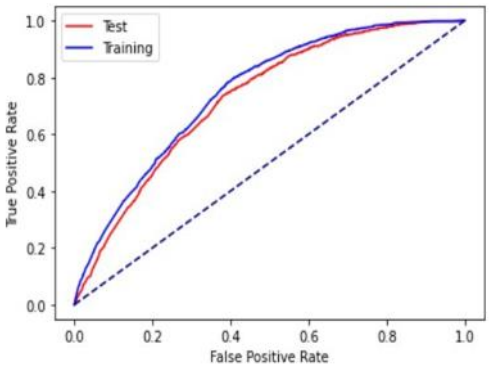
- Some predefined hyper parameters are set and passed to GridSearchCV
- Best parameter values are obtained from GridSearchCV.
- Here, max_depth is the maximum depth of the tree. If it is not specified, then the nodes get split and tree expands till it is left with pure leaves or until they have samples less than min_samples_split
- n_estimators is indication of number of trees
- class_weight: There are 2 modes – balanced and balanced_subsample. In balanced, weights are composed automatically based on output column and in the balanced_subsample, they are calculated based on bootstrap sample of each tree.

```
Pipeline(steps=[('rf',
                 RandomForestClassifier(class_weight='balanced_subsample',
                                        max_depth=5, max_samples=2000,
                                        n_estimators=10))])
```

- Validation and test score are computed.
- Prediction is also done using test dataset.
- The performance of the model is defined using confusion matrix



- ROC curve is drawn to display the performance.



SVM (Support Vector Machine)

It is one of most elegant methods for classification. It represents each object as a point in an N-dimensional space and its coordination are termed as features. This algorithm classifies the objects into 2 categories. It classifies by drawing a

hyperplane which could be a line in 2D or 3D. All points of a certain category fall on same side of the plane. There could be many planes but SVM tries to fit the one that suits the best. Distance between 2 points of either category is called 'Margin' and supporting vectors are the points that are exactly on the margin.

SVM is used to solve convex optimization problem. It maximizes the margin to ensure that points related to same category lie on one of the plane. SVMs are easy to understand and implement.

The following steps are followed in the project for this model:

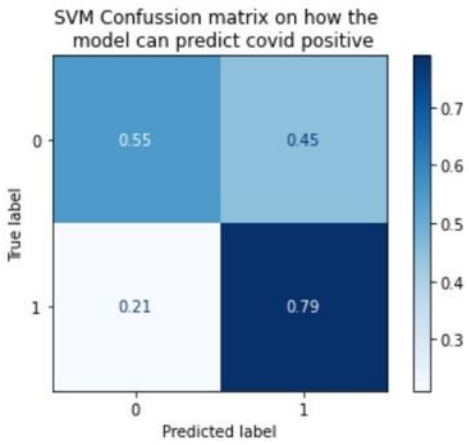
- A model is created.

```
from sklearn.svm import SVC
svm_m = modeling_pipeline = Pipeline([('svc', SVC(probability=True))])
```

- Some predefined hyper parameters are set and passed to GridSearchCV
- Best parameter values are obtained from GridSearchCV.
- Here, C is the penalty parameter.
- Kernel trick helps when some points are not possible to show in space. It splits them using hyper plane in a high dimensional space and projects them onto normal space. It has different types - linear', 'precomputed', 'rbf', 'poly', 'sigmoid', or a callable where rbf is the default.

```
Pipeline(steps=[('svc', SVC(C=10, kernel='linear', probability=True))])
```

- Validation and test score are computed.
- Prediction is also done using test dataset.
- The performance of the model is defined using confusion matrix



Naive Bayes

This machine learning algorithm is used for classification problems. Primarily, it is used for text classification that needs high dimensional training dataset. It is known for simplicity and effectiveness as it helps to build models fast. It tells the probability of an object based on features. The term 'naïve' is due to its assumption that occurrence of a particular feature is independent of others. This method is based on Bayes theorem.

The following steps are followed in the project for this model:

- A model is created. MultinominalNB is the Naïve Bayes classifier. It classifies discrete features and considers feature counts for this purpose.

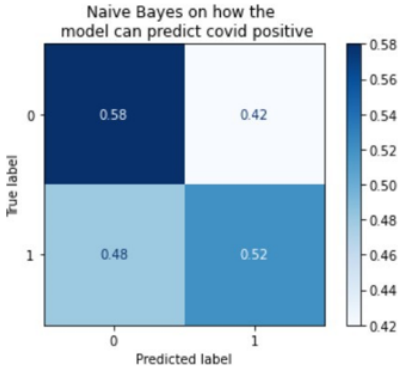
```
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import ConfusionMatrixDisplay

nb_m = modeling_pipeline = Pipeline([('model', MultinomialNB())])
```

- Some predefined hyper parameters are set and passed to GridSearchCV
- Best parameter values are obtained from GridSearchCV.
- Alpha is a smoothing parameter

```
Pipeline(steps=[('model', MultinomialNB(alpha=1))])
```

- Validation and test score are computed.
- Prediction is also done using test dataset.
- The performance of the model is defined using confusion matrix



K-nearest neighbor (KNN)

If there are N training vectors, then KNN algorithm identifies k nearest neighbors of ‘c’ regardless of labels. ‘c’ is another feature that we need to estimate the class.

The following steps are followed in the project for this model:

- A model is created.

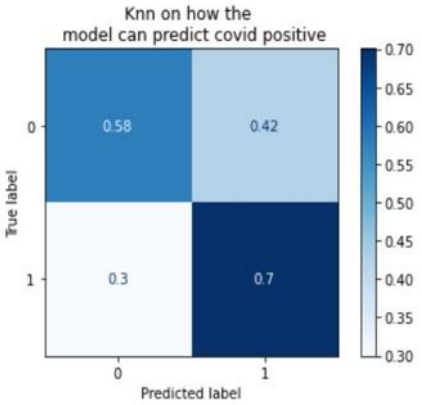
```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import ConfusionMatrixDisplay

param_knn = [
  {'model__n_neighbors': [ 10, 15,20,25], 'model__weights': ['uniform','distance']}
 ]
```

- Some predefined hyper parameters are set and passed to GridSearchCV
- Best parameter values are obtained from GridSearchCV.
- Indicating the number of neighbors is important in this model
- n_neighbors indicate the number of neighbors
- weight function is used for prediction. It can have – uniform or distance or callable. In uniform, all points present in the neighborhood are equally weighted. This is the default one.
- In distance, points are weighted by their distance inverse callable is user defined. It takes an array of distances and outputs an array of weights.

```
Pipeline(steps=[('model', KNeighborsClassifier(n_neighbors=15))])
```

- Validation and test score are computed.
- Prediction is also done using test dataset.
- The performance of the model is defined using confusion matrix



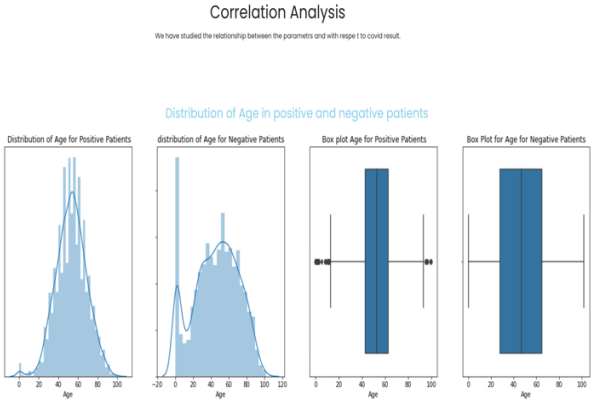
V. Project Evaluation

COVID-19 endangers both the healthcare and economic sectors. It is clear that non-clinical approaches such as machine learning, data mining, expert systems, and other artificial intelligence techniques will be critical in detecting and controlling the COVID-19 outbreak. The motivation for this initiative is to contribute my domain and technical skills to the battle against the illness.

In this project, supervised machine learning techniques are used to develop predictive models for COVID-19 infection, using an epidemiology labeled dataset for positive and negative COVID-19 cases in Mexico, with supervised learning algorithms such as decision tree, logistic regression, Random

Forest and naive Bayes, support vector machine, and k-nearest neighbors.

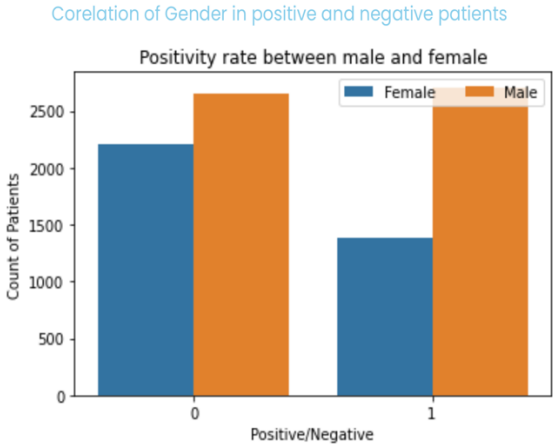
Correlation Analysis



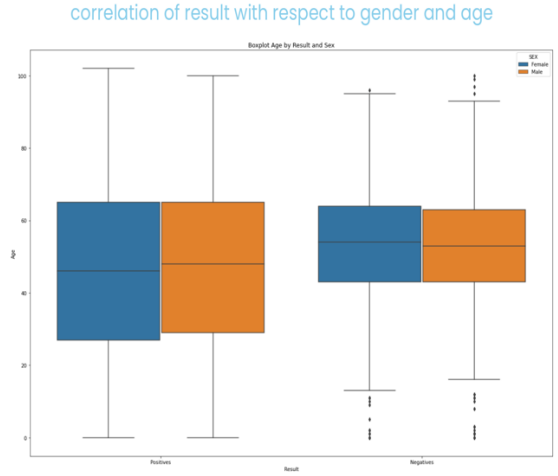
The graph for positive patience has a Gaussian curve. Hence, it can be assumed during any measurement values will follow a normal distribution with an equal number of measurements above and below the mean value. It can be determined from the graph that people of middle-aged group (30-60) has higher chance of getting positive.

The graph for negative patience has a Polynomial curve. Polynomial functions of degree 2 or more have graphs that do not have sharp corners; recall that these types of graphs

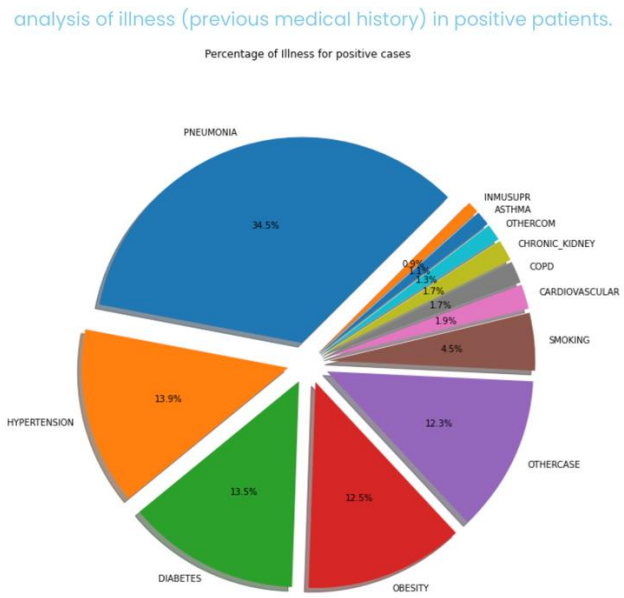
are called smooth curves. We can determine by the graph that it is scattered for negative patients and we cannot interpret the exact relationship between the age group and negative report of that age group.



From the above graph, it can be depicted that more men are affected by Covid virus compared to women and irrespective of whether positive or negative, men seem to show more symptoms than women and hence might have been tested to detect the presence of the virus.

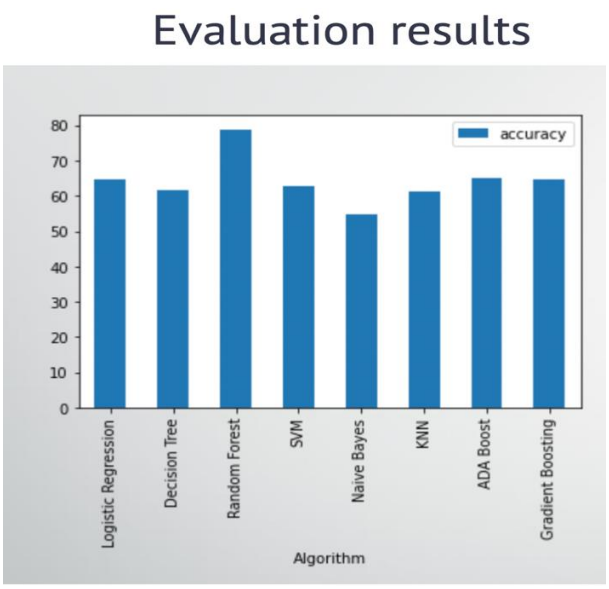


The above plot is to show the result with respect to age and gender.



The pie chart shows how the people are affected based on their medical history.

Evaluation of Results



As we can see the Random Forest model gives us highest accuracy of 78.80%.

VI. Conclusion

Early COVID-19 prediction can help us reducing unnecessary burden on healthcare systems by aiding in the diagnosis of COVID-19 patients.

Prior to developing the models in this project, the correlation coefficient analysis between multiple dependent and independent

characteristics was done to determine the strength of the relationship between each dependent and independent feature of the dataset.

The models were trained using 60% of the training dataset, while the remaining 40% was used to test them. We have used 'Accuracy' parameter to evaluate performance for all this models. And as you can see, The Random Forest model has the Highest accuracy of 78.80 percent, followed by logistic regression, ADA boost, and Gradient Boosting, all of which are around 65 percent.

VII. Future Work

There is more scope to this project. It can be extended to create a sophisticated algorithm that can predict any virus/disease based on patient history, medical conditions and potential risk factors.

VIII. External Links

Colab Link:
https://colab.research.google.com/drive/1NxhllLp1CJINNDGG0Cc2xLsQmhZ_w2MT#scrollTo=jwDLdQX-fnoS

Dataset Link:
https://www.gob.mx/salud/documentos/datos-abiertos-152127

Webpage Link:
A webpage has been developed to show all the algorithms, correlation analysis and evaluation results at one place.
https://raksha-modal.netlify.app

IX. References

[1] S.Bekmezci and T.Yiğit, "Comparing

estimation achievements by determining ideal training iteration numbers in supervised machine learning algorithms" *2017 International Conference on Computer Science and Engineering (UBMK)*, Antalya, pp. 649-654, Oct 2017.

[2] V. V. P. Wibowo, Z. Rustam, A. R. Laeli and A. A. Sa'id, " Logistic Regression and Logistic Regression-Genetic Algorithm for Classification of Liver Cancer Data " *2021 International Conference on Decision Aid Sciences and Application (DASA)*, Sakheer, pp. 244-248, Dec 2021.

[3] X. Oiao, and J. Luo, "Design Method of Banquet Intelligent Side Dishes System Based on Data Mining and Correlation Analysis" *2020 IEEE 20th International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, Macau, pp. 434-437, Dec 2020.

X. Appendix

Machine Learning Algorithms

The entire 21st century is changed by the evolution of AI (Artificial Intelligence). AI has got the spotlight in terms of technology. The advancements of AI are faster than we predicted. Due to this exponential growth in the AI century, Machine Learning (ML) has become the most important training field in this century. Hence, it is very important to redefine the way we live and understand the importance of these advancements in our life.

Machine learning is a subset of AI. It is the science that feeds the computers with data and makes them learn on their own without being programmed explicitly. That means, machine learning is an application of AI that provides systems with an ability to learn automatically on their own and also improve their performance without having to program explicitly. In machine learning, there will be continuous feeding of data so that the machine can interpret the data, understand the insights, detect patterns and also identify key features which can be used to solve problems and make better decisions in the future. This process is similar to how our brain works. It mainly focuses on developing computer programs which can access the data, avail it by learning themselves. An algorithm will be applied on the collected data and predict the outcome. This outcome will be helpful to make better decisions.

Machine learning can be used in many sectors like in e-commerce sectors to predict the shoppers' behavior.

Types of Machine Learning

Basically, there are three types of machine learning algorithms. They are:

1. Supervised Learning
2. Unsupervised Learning
3. Reinforcement Learning

Supervised Learning: 'Supervised' means to direct an activity and ensure that it is done and works correctly. In this type of learning, the machine learns under the guidance i.e., machines learn by feeding them data. Input and output are already known and informed to the machine. Supervised machine learning can apply what was learnt from the past events to a new data using labeled examples and in turn, this is useful to predict the future events. The algorithm produces an inferred function which will be beneficial to do the predictions for the output values. Once training is completed, the system provides targets to any new data input. This learning algorithm is useful in comparing its output against the correct and proposed output. Upon comparing, any errors are notified so that the model can be changed accordingly.

Unsupervised Learning: 'Unsupervised' means to act without any supervision. In this learning, the data is not labeled and it is left to the

machine to figure out the output by finding hidden patterns in the data.

Reinforcement Learning: 'Reinforcement' means to establish a behavior pattern. In this learning, hit and trail concept is followed. An agent, stuck in an environment, interacts with it by producing actions and discovers the errors or rewards. Once the agent gets trained, then it can predict the new data.

Comparison of machine learnings

Supervised, unsupervised and reinforcement machine learnings are compared against each other in various categories and presented in the below tabular format.

Category	**Supervised**	**Unsupervised**	**Reinforcement**
Definition	Machine is taught using label data.	Data provided to the machine is not labelled and the machine has to learn without any supervision.	An agent is put in an unknown environment and the agent has to explore the environment by taking actions
Type of problems solved	2 main categories of problems: regression and classification problems.	This type of learning can be used to solve association problems and clustering problems, anomaly detection	Input depends on the actions we take and for each action it takes it can get a reward or punishment.
Types of data	Labelled data	Unlabeled data	No predefined data
Training	External Supervision	No supervision	The entire process is training and testing as there is no predefined data.
Aim/ End goal	Forecast an outcome- the machine can directly give you a predicted outcome as it has a very well-defined training phase	It is all about discovering patterns and finding useful insights.	Agent is lot like a human child
Approach followed	Map the known input to known output	Finds patterns and trends in data and continues till it reaches the output	Trial and error method is followed
Output feedback	Direct feedback mechanism	No feedback mechanism as the machine	Agent gets rewards/punishments from the environmen

	as the machine is trained with labelled input and output	gets unlabeled input and is unaware of the output	t based on the actions
Popular algorithms	Linear regression, Logistic regression Support Vector machines (SVM), decision trees, logistic regression, random forest, K nearest neighbour	K-Means Apriori C-Means	Q-learning SARSA
Applications	Mainly used in the business sector for risk analysis, sales, profit	The recommendations we get while shopping online, credit card fraud detection anamoly	Self-driving cars, games etc.

	etc.	detection etc.	