

DEVELOPMENT OF ARTIFICIAL INTELLIGENCE BASED PLATFORM USING BIG DATA ANALYTICS FOR INTEGRATING MULTIPLE DISEASE RELATED DATA POINTS TO DEVELOP FUTURE DISEASE PREDICTION AND SURVEILLANCE SYSTEM

Submitted in partial fulfillment of the
requirements for the award of
Bachelor Of Technology Degree in Information Technology

By

P.VASANTH (REG NO:39120132)



**DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHISALAI,
CHENNAI - 600119**

MAY- 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with—A⁺ grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in

DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **VASANTH P (Reg No.39120132)** who carried out the Project Phase-2 entitled “**DEVELOPMENT OF ARTIFICIAL INTELLIGENCE BASED PLATFORM USING BIG DATA ANALYTICS FOR INTEGRATING MULTIPLE DISEASE RELATED DATA POINTS TO DEVELOP FUTURE DISEASE PREDICTION AND SURVEILLANCE SYSTEM**” under my supervision from January 2023 to May 2023.

Internal Guide

V. Nirmal Rani
11/05/2023

Dr. V.NIRMAL RANI M.Tech., Ph.D.,

R. Subhashini
10/5/23

Head of the Department

Dr. R. SUBHASHINI M.E., Ph.D.,

Submitted for Viva voce Examination held on 11.05.2023

[Signature]

Internal Examiner

[Signature]

External Examiner

DECLARATION

I, **Vasanth P** (Reg.No: 39120132), hereby declare that the Project Phase-2 Report entitled **Development Of Artificial Intelligence Based Platform Using Big Data Analytics For Integrating Multiple Disease Related Data Points To Develop Future Disease Prediction and Surveillance System** done by me under the guidance of **Dr.V. Nirmal Rani, M.TECH.,Ph.D** is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in **Information Technology**.

DATE:11.05.2023

PLACE:Chennai

A handwritten signature in black ink, appearing to read 'P. Vasanth', with a stylized flourish at the end.

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **the Board of Management of SATHYABAMA** for their kind encouragement in doing this project and completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph. D, Dean**, School of Computing, **Dr.R.Subhashini, M.E., Ph.D. Head of the Department of Information Technology** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.V.Nirmal Rani M.TECH., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

ABSTRACT

Contagious illnesses continue to be a primary cause of mortality among juveniles and young adults, representing 50% of all fatalities in underdeveloped nations. Despite efforts to control known infectious diseases, the battle against these deadly illnesses is far from over, and new threats continue to emerge. The emergence of COVID-19 is currently posing a challenge to researchers. COVID-19 has emerged as a global public health crisis, and accurate risk prediction models are critical for effective clinical management of patients. The evaluates the performance of different AI algorithms in predicting mortality risk among COVID-19 patients using a large dataset of over 1 million patients. A prediction System is necessary to ensure that the government can implement the addressing mechanism and public health policies, particularly in settings with limited health resources. This paper focuses on the development of an artificial intelligence-based prediction system that incorporates multiple disease-related data points using big data analytics.

LIST OF CONTENTS

Chapter No	TITLE		Page No.
	ABSTRACT		v
	LIST OF FIGURES		vii
1	INTRODUCTION		1
2	LITERATURE SURVEY		2
	2.1	Bibliographic research	2
	2.2	Inference from Literature Survey	6
	3REQUIREMENTS ANALYSIS		8
	3.1	Package Requirements	8
	3.2	Software Requirements Specification Document	8
	3.3	Design And Implementation Constraints	9
4	DESCRIPTION OF THE PROPOSED SYSTEM		10
	4.1	Selected Methodology or process model	10
	4.2	Analysis	10
	4.3	Description of Software for Implementation and Testing Plan of the Proposed Model/System	21
	4.4	Machine Learning Models	22
	4.5	Feature scaling	39
	4.6	Description of Surveillance System	44
	4.7	Methodology	44
	4.8	Surveillance System Dashboard	45
5	Results and Discussion		51
	5.1	Results	51
	5.2	Discussion	53
6	CONCLUSION AND FUTURE WORK		56
	6.1	Conclusion And Future Work	56
	REFERENCES		59
	APPENDIX		61
	Research Paper		61

LIST OF FIGURES

FIG NO	FIG NAME	PAGE NO
4.1	Data Variables	10
4.2	Case Positivity	16
4.3	Gender Plot	16
4.4	COPD	16
4.5	Asthma	16
4.6	Immuno Suppresion	16
4.7	Hypertension	16
4.8	Other Comm Disease	17
4.9	Cardiovascular	17
4.10	Obesity	17
4.11	Renal chronic	17
4.12	Diabetes	18
4.13	Smoking	18
4.14	ICU Admission	18
4.15	Random Forest Plot	24
4.16	SVM Model	25
4.17	SVM Graph	26
4.18	Decision Tree	27
4.19	Gradient Boost	30
4.20	Bernoulli Naive	33
4.21	Ada Boost	35
4.22	Logistic Model	36
4.23	Logistic Regresion	37
4.24	KNN Algorithm	39
4.25	Feature Scaling	41
4.26	Correlation Matrix Before FS	42
4.27	Correlation Matrix after FS	43
4.28	Streptococcus Pneumonia Serotypes	47
4.29	Laboratory Confirmed Cases	48
4.30	Distribution of Pathogens	49
4.31	Surveillance Network	50
5.1	Accuracy of Algorithms	52

CHAPTER I

INTRODUCTION

This is ICMR-NATIONAL INSTITUTE OF EPIDEMIOLOGY Project. the aim of the project is to Development of Artificial Intelligence Based Platform Using Big Data Analytics for Integrating Multiple Disease Related Data Points To Develop Future Disease Prediction System

In December 2019, a coronavirus disease 2019 (COVID-19) outbreak emerged in Wuhan City, Hubei province in China. COVID-19 is a disease of the respiratory tract characterized by a severe acute respiratory syndrome; the causative agent is a beta coronavirus named severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). Despite a rigorous global containment and quarantine efforts, the incidence of COVID-19 continued to expand and rise, becoming a global pandemic, that currently is in all continents. The pandemic caused by the novel coronavirus, SARSCoV-2, has become one of the biggest health challenges worldwide. There are nearly 44 million confirmed cases and more than 5.4 lakh deaths as of sept 9th, 2022. This makes the COVID-19 pandemic one of the public health problems with a meaningful impact in the history of humanity. COVID-19 is commonly asymptomatic. But a significant proportion of COVID-19 cases develop pneumonia and acute severe respiratory failure. These cases commonly require hospitalization, intensive care unit admission, and intubation. Older age, the presence of comorbidities, particularly hypertension, diabetes, obesity, and smoking are factors that increase the risk of severe disease. The characteristics of COVID-19 may vary depending on the demographic and epidemiological areas. Assessing the instantaneous rate of death at any time during follow-up for the specific risk factors is crucial to determine the appropriateness of the mitigation strategies and to set up priorities to control the COVID-19 epidemic. This is especially so for countries that are in the present moment focusing their efforts on fighting the pandemic. The SARS-CoV-2 outbreak poses a challenge to health care systems due to its high complication rates in patients. Here, we identify risk factors and propose a clinical score to predict COVID-19 lethality.

CHAPTER 2

2. LITERATURE SURVEY

2.1 BIBLIOGRAPHIC RESEARCH

2.1.1 Liang W, Liang H, Ou L, et al. The development and validation of a clinical risk score to predict the occurrence of critical illness in hospitalized patients with COVID-19 is a process that involves creating a tool to estimate the likelihood of severe disease progression in patients with COVID-19 who are admitted to the hospital. This score is designed to help healthcare providers identify patients who may require more intensive treatment and monitoring in order to prevent adverse outcomes. The development of such a risk score typically involves analyzing data from a large group of patients with COVID-19 who have been admitted to hospitals. Researchers analyze a range of clinical and demographic factors that may be associated with severe disease progression in these patients. These factors may include age, underlying medical conditions, laboratory test results, and clinical symptoms. Once potential risk factors have been identified, researchers can use statistical methods to develop a scoring system that assigns a numerical value to each risk factor based on its strength of association with severe disease progression. The scores for each risk factor can then be combined to generate an overall risk score for each patient. This score provides an estimation of the probability of critical disease progression for a hospitalized patient with COVID-19. Researchers typically compare the predicted risk scores with actual outcomes to assess the accuracy of the tool.

2.1.2 Gong J, Ou J, Qiu X, et al. The tool for early prediction of severe COVID-19 is a risk nomogram developed through a multicenter study in Wuhan and Guangdong. This nomogram is designed to help healthcare providers identify patients with a higher risk of severe disease progression and potentially reduce mortality rates. The study analyzed data from a large group of patients with COVID-19 and identified several factors that were associated with a higher risk of severe disease, such as age, underlying medical conditions, and laboratory test results. These factors were used to

develop a risk nomogram, which assigns a numerical score to each risk factor and provides an overall risk score for each patient. Using the risk nomogram, healthcare providers can quickly identify patients with a higher risk of severe disease and provide more intensive treatment and monitoring.

2.1.3 Qi X, Jiang Z, Yu Q, et al. The machine learning-based CT radiomics model is a tool developed through a multicenter study to predict the hospital stay of patients with pneumonia associated with SARS-CoV-2 infection. The model uses artificial intelligence and radiomics analysis to analyze CT scans of patients and predict their hospital stay. By predicting the hospital stay of patients with pneumonia associated with SARS-CoV-2 infection, the model can assist healthcare providers in better allocating resources and preparing for the needs of patients. This can help hospitals and healthcare systems be better prepared to manage patient care and optimize resource utilization. The machine learning-based CT radiomics model is an important development in the field of COVID-19 management, as it provides a tool to help healthcare providers make more informed decisions about patient care and resource allocation. By optimizing the use of resources and predicting hospital stays more accurately, the model can help improve patient outcomes and reduce healthcare costs.

2.1.4 Dutta S, Bandyopadhyay SK. The machine learning approach for confirmation of COVID-19 cases is a tool developed to help confirm positive and negative cases of COVID-19, as well as predict death or release outcomes for patients. This approach uses artificial intelligence and machine learning algorithms to analyze data from multiple sources, such as clinical symptoms, laboratory test results, and radiographic findings, to make accurate predictions. The AI-based prediction tool has shown significant accuracy in confirming COVID-19 cases and predicting outcomes. This tool can be used to help healthcare providers make more informed decisions about patient care, resource allocation, and policymaking. By accurately predicting outcomes, the tool can also help hospitals and healthcare systems better manage their resources and plan for the needs of patients.

2.2.5 Siwiak M, Szczesny P, Siwiak M. From the index case to global spread: the global mobility-based modelling of the COVID-19

pandemic implies higher infection rate and lower detection Predictions will help check supply and medical assistance and help policymakers prepare for outbreak

2.1.5 Van den Broeck W, Gioannini C, et al. The Glean viz computational tool is a publicly available software that allows researchers to explore realistic epidemic spreading scenarios on a global scale. The tool uses advanced computational models and simulations to predict the spread of infectious diseases like COVID-19 and to explore potential outcomes under different scenarios. One application of the Glean viz tool is in predicting the hospital stay for patients with pneumonia associated with COVID-19 infection. The tool has demonstrated feasibility and accuracy in making these predictions, which can help healthcare providers better prepare for patient care and optimize resource utilization. By using the Glean viz computational tool, researchers and healthcare providers can explore a range of scenarios and assess the potential impact of interventions such as social distancing measures, vaccination campaigns, and travel restrictions. This can help inform policymaking and public health decision-making, and aid in the development of effective strategies to mitigate the spread of COVID-19.

2.1.6 Regmi K, Lwin CM. The impact of social distancing measures for preventing COVID-19 is a systematic review and meta-analysis protocol designed to assess the effectiveness of social distancing measures in reducing the spread of COVID-19. The study uses advanced statistical methods such as regression algorithms to analyze data from multiple sources, including published studies and government reports. The study aims to provide a comprehensive assessment of the impact of social distancing measures on reducing the spread of COVID-19. This can help inform policymakers and public health decision-makers about the most effective strategies for mitigating the spread of the disease. By analyzing the data using regression algorithms, the study can also identify potential confounding factors and adjust for these in the analysis. This can help ensure that the results are accurate and unbiased

2.1.7 Albahri AS, Al-Obaidi JR, Zaidan AA, et al. The multi-biological laboratory examination framework is a tool developed for the prioritization of patients with

COVID-19. The tool uses a combination of integrated Analytic Hierarchy Process (AHP) and group VIKOR methods to assess and rank patients based on their laboratory examination results. By using this framework, healthcare providers can prioritize patients based on their risk levels and severity of the disease, which can help optimize resource utilization and patient care. The AHP method is used to evaluate the importance of each laboratory examination result, while the VIKOR method is used to rank patients based on their laboratory examination results. The framework is based on multiple biological laboratory examinations, including blood tests, chest imaging, and other diagnostic tests. By integrating these results, the framework can provide a comprehensive assessment of each patient's condition and prioritize patient care accordingly.

2.1.8 Alazab M, Awajan A, Mesleh A, et al. COVID-19 prediction and detection using deep learning is a machine learning approach to predict and detect the occurrence of COVID-19. The model uses deep learning algorithms to analyze data from multiple sources, including patient health records, laboratory results, and epidemiological data. The model is highly accurate for short-term predictions, typically 1-3 days ahead, which can help healthcare providers anticipate and prepare for potential outbreaks. By using advanced machine learning techniques, the model can identify patterns and trends in the data that may be indicative of the spread of COVID-19.

2.1.9 Li Y, Horowitz MA, Liu J, et al. Individual-level fatality prediction of COVID-19 patients using AI methods is a machine-learning approach to predicting the risk of fatality among COVID-19 patients. The method uses logistic regression, a type of multivariable regression model, to analyze patient data and identify factors that contribute to the risk of fatality. By using this approach, healthcare providers can identify patients who are at a higher risk of fatality and allocate resources accordingly. This can help ensure that patients receive the appropriate level of care and support, and that healthcare resources are used more efficiently. The AI-based predictions generated by this method can assist the flow of patients in the healthcare system.

2.2 INFERENCE FROM LITERATURE SURVEY

ARIMA models

ARIMA models can be built in an array of software tools, including Python. These models are used in statistics and econometrics to measure events that happen over a span of time. ARIMA models predict future data in a series using past data. An ARIMA model can be constructed for any number series that display patterns and is not a random event series. For example, sales data from a footwear store would be an example of time series data because the data are collected over a period of time. One of the key characteristics is that the data are collected at constant, regular intervals

SEIR models

SEIR models are commonly used for assessing infection data during the different phases of an infectious outbreak. SEIR models are among the most widely adopted mathematical frameworks to describe disease dynamics and forecast potential contagion scenarios. After an infectious disease outbreak, a SEIR model can be helpful in determining the efficacy of different interventions, such as lock-downs. These models are based on a series of complex ordinary differential equations that take into account the number of people who are sick, the pattern of people who recover over time after sickness, and the people who die.

LSTM models

LSTM models are a type of recurrent neural network (RNN) used to predict new infection numbers over time by processing and forecasting several issues related to time series. With repeating modules like an RNN, an LSTM model has a chain-like structure, except that instead of a single neural network layer as in RNNs, an LSTM model has 4 layers that communicate in a slightly different manner, each of which performs its own special network role. In an LSTM cell, each repeating module has a cell state. Through using various gates in the cell, the LSTM cell has the power to add or subtract information to

the cell state. There are 3 gates for the standard LSTM cell that control the sum of data input or output to/from the cell state and protect the cell state.

Regression models

Regression analysis is a method of quantitative research that is used in studies modeling and analyzing several variables, where a dependent variable and 1 or more independent variables are included in the relationship. In basic terms, regression analysis is a mathematical approach used to evaluate the existence of the relationship between a dependent variable and 1 or more independent variables. The 2 most widely used regression analyses are: (a) Logistic regression: in logistic regression, an independent variable is used to estimate the dependent variable. (b) Support vector regression (SVR): SVR provides the flexibility to determine how much error is suitable in a model and to find an appropriate line (or hyperplane in higher dimensions) to match the results.

GLEM models

Global epidemic and mobility (GLEM) models are being used in a number of COVID-19 related studies and analyses. These models involve a stochastic computational framework that combines high-resolution demographic and mobility data across the globe to predict the epidemic distribution across the globe. The goal of the GLEM model is to optimize versatility in specifying the disease compartment model and configuring the simulation scenario. It allows the user to set a number of criteria, including compartment-specific features, transition values, and environmental effects

CHAPTER 3

3. REQUIREMENT ANALYSIS

3.1. PACKAGES REQUIREMENTS-

1. Random Forest Algorithm: Scikit-Learn (Python), Random Forest(R), H2o (R/Python).
2. Svm Algorithm: Scikit-Learn (Python), E1071 (R), Libsvm (Java).
3. Decision Tree: Scikit-Learn (Python), Rpart (R), Party (R).
4. Gradient Boosting Algorithm: Xgboost (Python/R), Lightgbm(Python/R), Catboost.
- 5.Naïve Bayes Algorithm: Scikit-Learn (Python), E1071 (R).
- 6.Bernoulli Nb Algorithm: Scikit-Learn (Python), E1071 (R).
7. Ada Boost Algorithm: Scikit-Learn (Python), Gbm (R), H2o (R/Python)
- .8.Logistic Regression Algorithm: Scikit-Learn (Python), Glm (R) ,H2o (R/Python)
9. Knn Algorithm: Scikit-Learn (Python), Class (R) ,Kknn (R).

3.2. HARDWARE AND SOFTWARE SPECIFICATION

✓ Hardware Requirement

PC, Pentium 4 processor, 1 GB RAM, CPU 3.06 GHz

✓ Software Requirement

MATLAB R2018a

✓ Technology Used

Python

3.3 DESIGN AND IMPLEMENTATION CONSTRAINTS

3.3.1 Constraints in Analysis

- Constraints as Informal Text
- Constraints as Operational Restrictions
- Constraints Integrated in Existing Model Concepts
- Constraints as a Separate Concept
- Constraints Implied by the Model Structure

3.3.2 Constraints in Design

- Determination of the Involved Classes
- Determination of the Involved Objects
- Determination of the Involved Actions
- Determination of the Require Clauses
- Global actions and Constraint Realize

3.3.3 Constraints in Implementation

A hierarchical structuring of relations may result in more classes and a more complicated structure to implement. Therefore, it is advisable to transform the hierarchical relation structure to a simpler structure such as a classical flat one. It is rather straightforward to transform the developed hierarchical model into a bipartite, flat model, consisting of classes on the one hand and flat relations on the other. Flat relations are preferred at the design level for reasons of simplicity and implementation ease. There is no identity or functionality associated with a flat relation. A flat relation corresponds with the relation concept of entity-relationship modeling and many object-oriented methods.

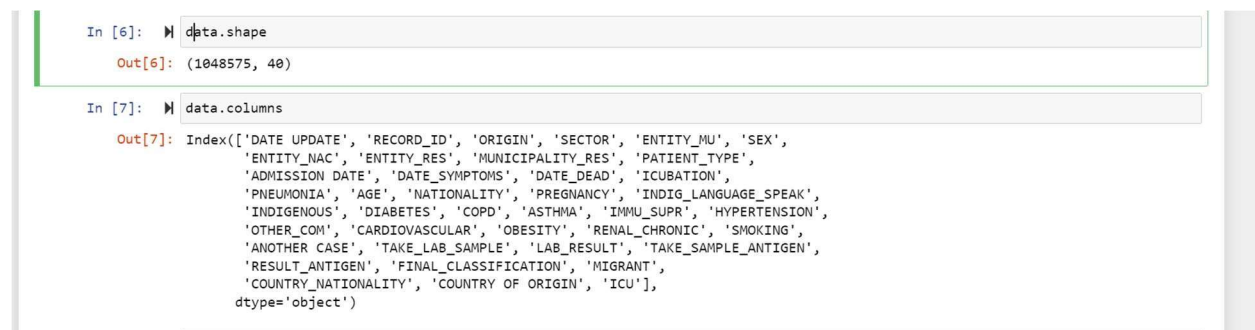
CHAPTER IV

4. DESCRIPTION OF THE PROPOSED SYSTEM

4.1. PROCESS MODEL

Data Source:

A database including COVID-19 monitoring cases, since January 1, 2020, was downloaded from the open data source of Epidemiologic Surveillance Source of Respiratory Viral Diseases that include information from monitoring units all along the country from the public and private health sectors. Positive cases were extracted and edited. Data from 10,48,575 patients diagnosed positively to COVID-19, from the first positive case registered were analyzed. All positive cases to COVID-19 were diagnosed using real-time PCR and were officialized by the National Network for Epidemiologic Surveillance recognized by the Institute of Epidemiological Diagnosis.



```
In [6]: data.shape
Out[6]: (1048575, 40)

In [7]: data.columns
Out[7]: Index(['DATE_UPDATE', 'RECORD_ID', 'ORIGIN', 'SECTOR', 'ENTITY_MU', 'SEX',
              'ENTITY_NAC', 'ENTITY_RES', 'MUNICIPALITY_RES', 'PATIENT_TYPE',
              'ADMISSION_DATE', 'DATE_SYMPTOMS', 'DATE_DEAD', 'ICUBATION',
              'PNEUMONIA', 'AGE', 'NATIONALITY', 'PREGNANCY', 'INDIG_LANGUAGE_SPEAK',
              'INDIGENOUS', 'DIABETES', 'COPD', 'ASTHMA', 'IMMU_SUPR', 'HYPERTENSION',
              'OTHER_COM', 'CARDIOVASCULAR', 'OBESITY', 'RENAL_CHRONIC', 'SMOKING',
              'ANOTHER_CASE', 'TAKE_LAB_SAMPLE', 'LAB_RESULT', 'TAKE_SAMPLE_ANTIGEN',
              'RESULT_ANTIGEN', 'FINAL_CLASSIFICATION', 'MIGRANT',
              'COUNTRY_NATIONALITY', 'COUNTRY OF ORIGIN', 'ICU'],
             dtype='object')
```

Fig:4.1 Data Variables

4.2 ANALYSIS-

The data set includes information of age, sex, smoking condition, exposure history, comorbidity traits, and clinical care management. Continuous age variable was described using the median and the interquartile range. Sex and discrete yes or no variables included survival information, smoking condition, patient hospitalization, COVID-19 contact, hypertension, obesity, cardiopathy, Pregnancy, pneumonia, chronic obstructive pulmonary disease, asthma, immunosuppression, chronic kidney disease, and other complications. Clinical care management was described by intensive care unit (ICU) access and endotracheal intubation. All discrete variables were described using the number of cases and percentages. Given the small number of observations for some

variables such as origin, sector, municipality, area, residential place, update date, COVID-19 contact, ICU, and intubation were excluded which is not essential for the study

DATA Is very important for every aspect of the research. The ICMR- national institute of epidemiology does test, surveys all over India collects the data of the disease which is predominantly spread across the country.

- Provides a systematic approach for dissecting a health problem into its component parts.
- Ensures that you are fully versed in the basic dimensions of a health problem.
- Identifies populations at increased risk for the health problem under investigation.
- Provides timely information for decision-makers, the media, the public, others about ongoing investigations.
- Supports decisions for initiating or modifying control and prevention measures.
- Measures the progress of control and prevention Program.
- Enables generation of testable hypotheses regarding the etiology, exposure mode, control measure effectiveness, and other aspects of the health problem.
- Helps validate the eventual incrimination of causes or Risk factors.

Many real-world networks exhibit community structure, and different communities may often possess heterogeneity. In this paper, considering the heterogeneity among communities, we construct a new community network model in which the communities show significant differences in average degree. Based on this heterogeneous community network, we propose a novel mathematical epidemic model for each community and study the epidemic dynamics in this network model. We find that the location of the initial infection node only affects the spreading velocity and barely influences the epidemic prevalence. And the epidemic threshold of entire network decreases with the increase of heterogeneity among communities. Moreover, the epidemic prevalence increases with the increase of heterogeneity around the epidemic threshold, while the converse situation holds when the infection rate is much greater than the epidemic threshold.

The mathematical models have become important tools in analyzing the spread and control of infectious diseases. The development of such models is aimed at both understand observed epidemiological patterns and predicting the consequences of the introduction of public health interventions to control the spread of diseases. Some

diseases not confer immunity against the disease but other diseases confer immunity so recovered individuals gain immunity against disease. These types of disease can be modifications by SI and SIS where S susceptible and I infective respectively. Both epidemic models (SI and SIS) are one of the most basic and most important models in describing of many diseases. Therefore, it attracted many authors attention and several papers have been published. For example, Gao and Heathcote considered an SIS model with a standard disease incidence and density dependent demographics. Li and Ma studied an SIS model with vaccination and temporary immunity. Karnack and McKendrick proposed a simple SIS model with infective immigrants. In recent years, many papers found treatment function for example, Li et al proposed the SIS model with a limited resource for treatment. Shurowq k. Shafeeq studied the effect of treatment, immigrants and vaccinated on the dynamic of SIS epidemic model. In this paper we proposed and studied a mathematical model consisting of SII S* epidemic model with treatment, in which it is assumed that the disease transmitted by contact as well as external sources in the environment. The local as well as global stability analysis.

Modelling of disease transmission via compartmental models is well established and generally highly effective. However, the differential equations of these models depend on good estimates of underlying rate parameters and will then provide a continuous solution under the assumption that the population is well-mixed and homogeneous (i.e., all individuals have equal contact with all others). Under these assumptions disease propagation is driven by the parameter R_0 —the ratio of the rate of new infections to the rate of removal of infectious individuals from the transmission pool. Typically, and particularly for contemporary and evolving transmission, these parameters can be somewhat difficult to estimate the emergence of an earlier coronavirus, associated with the severe acute respiratory syndrome (SARS) in 2003, provided an opportunity to apply the structures and concepts of complex systems to the modelling of infectious diseases. introduced a complex network-based model of propagation and showed good agreement between simulations of that model and available case data. They found that epidemic parameters widely quoted in the literature were only consistent with observed case data when including significant nosocomial transmission. Finally, and most importantly for the current discussion, the scale-free topology of the model explained super-spreader events through contact rather than requiring pathologically highly infectious individuals. Both the

network-based models used to model SARS in 2003. and the model we describe here are network models of contact between individuals. Unlike what we will propose in this current communication, the model of SARS in 2003 was topologically stationary]. The model assumed a lattice with long-range (i.e., small-world) connections following a power-law degree distribution. In those papers time varying control strategies were reflected only in changes of the rate parameters. The current coronavirus outbreak (that is, COVID-19) poses a different and unique challenge. Since February 2020 (and up to the time of writing) global transport networks and daily movement of individuals have been disrupted on a global scale. Entire cities and countries have engaged in various levels of “lockdown”. We argue that it is neither appropriate nor sufficient to model this simply by modifying the rate of transmission or rate of removal. In the current work we propose a network switching model through which the topology of the network changes to reflect various changes in government and community mitigation and control strategies.

What is data cleaning?

Data cleaning is the process of fixing or removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset. When combining multiple data sources, there are many opportunities for data to be duplicated or mislabeled. If data is incorrect, outcomes and algorithms are unreliable, even though they may look correct. There is no one absolute way to prescribe the exact steps in the data cleaning process because the processes will vary from dataset to dataset. But it is crucial to establish a template for your data cleaning process so you know you are doing it the right way every time.

Step 1: Remove duplicate or irrelevant observations

Remove unwanted observations from your dataset, including duplicate observations or irrelevant observations. Duplicate observations will happen most often during data collection.

When you combine data sets from multiple places, scrape data, or receive data from clients or multiple departments, there are opportunities to create duplicate data. De-duplication is one of the largest areas to be considered in this process. Irrelevant observations are when you notice observations that do not fit into the specific problem you are trying to analyze. For example, if you want to analyze data regarding millennial

customers, but your dataset includes older generations, you might remove those irrelevant observations. This can make analysis more efficient and minimize distraction from your primary target—as well as creating a more manageable and more performant dataset.

Step 2: Fix structural errors

Structural errors are when you measure or transfer data and notice strange naming conventions, typos, or incorrect capitalization. These inconsistencies can cause mislabeled categories or classes. For example, you may find “N/A” and “Not Applicable” both appear, but they should be analyzed as the same category.

Step 3: Filter unwanted outliers

Often, there will be one-off observations where, at a glance, they do not appear to fit within the data you are analyzing. If you have a legitimate reason to remove an outlier, like improper data- entry, doing so will help the performance of the data you are working with. However, sometimes it is the appearance of an outlier that will prove a theory you are working on. Remember: just because an outlier exists, does not mean it is incorrect. This step is needed to determine the validity of that number. If an outlier proves to be irrelevant for analysis or is a mistake, consider removing it.

Step 4: Handle missing data

You cannot ignore missing data because many algorithms will not accept missing values. There are a couple of ways to deal with missing data. Neither is optimal, but both can be considered.

1. As a first option, you can drop observations that have missing values, but doing this will drop or lose information, so be mindful of this before you remove it.

2. As a second option, you can input missing values based on other observations; again, there is an opportunity to lose integrity of the data because you may be operating from assumptions and not actual observations.

3. As a third option, you might alter the way the data is used to effectively navigate null values.

Step 5: Validate and QA

At the end of the data cleaning process, you should be able to validate the data. Determining the quality of data requires an examination of its characteristics, then weighing those characteristics according to what is most important to your organization and the application(s) for which they will be used.

5 characteristics of quality data:

- Validity. The degree to which your data conforms to defined business rules or constraints.
- Accuracy. Ensure your data is close to the true values.
- Completeness. The degree to which all required data is known.
- Consistency. Ensure your data is consistent within the same dataset and/or across multiple data sets.
- Uniformity. The degree to which the data is specified using the same unit of measure. Benefits of data cleaning Having clean data will ultimately increase overall productivity and allow for the highest quality information in your decision-making.

Benefits include:

- Removal of errors when multiple sources of data are at play. Fewer errors make for happier clients and less-frustrated employees.
- Ability to map the different functions and what your data is intended to do.
- Monitoring errors and better reporting to see where errors are coming from, making it easier to fix incorrect or corrupt data and future, applications.
- Using tools for data cleaning will make for more efficient business practices and quicker decision-making.
- After studying the structure of the data sets and doing the required modifications. The data is analyzed. the modified data-

Then all the disease data points are taken, they are then compared with the target variable i.e., Final classification. This comparative analysis is plotted. Results of the the analysis are given below

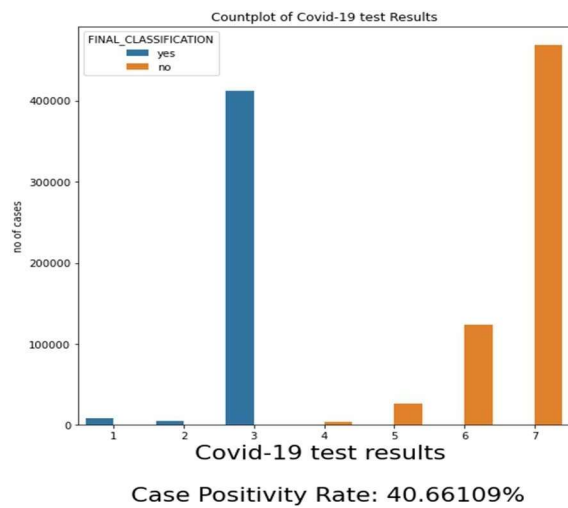


fig 4.2 Case positivity plot

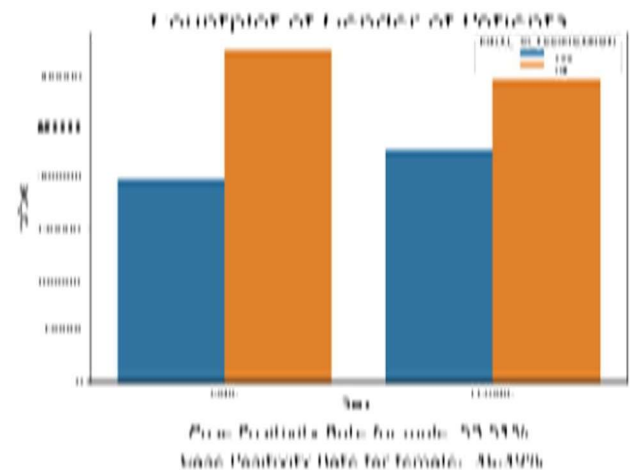


fig 4.3 Gender plot

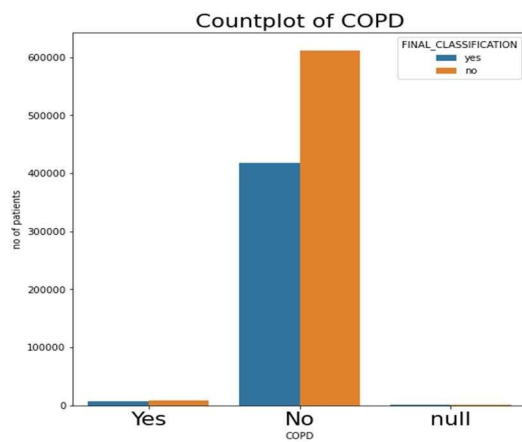


fig 4.4 COPD

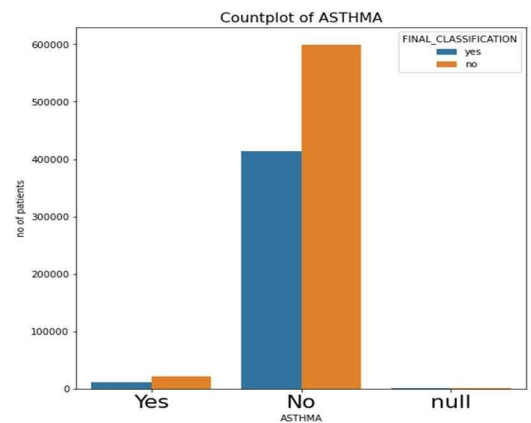


fig 4.5 Asthma

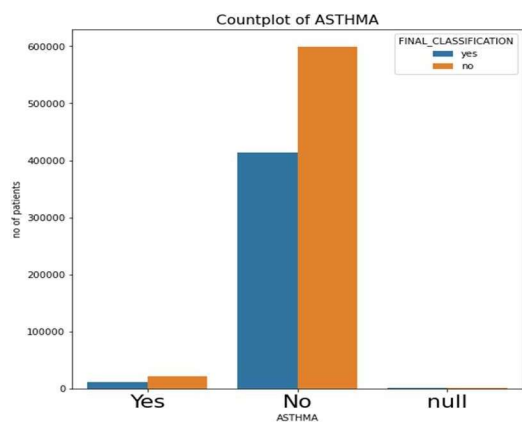


fig 4.6 Immuno suppression

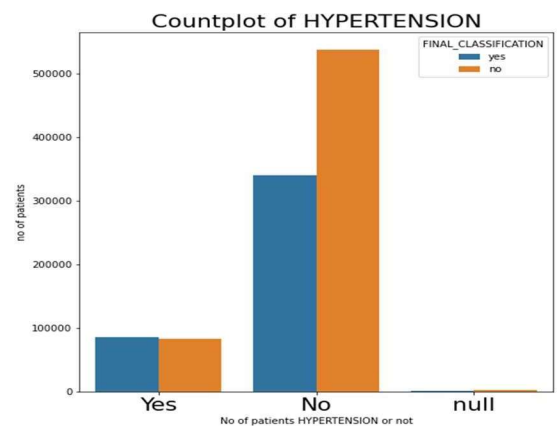


fig 4.7 Hypertension

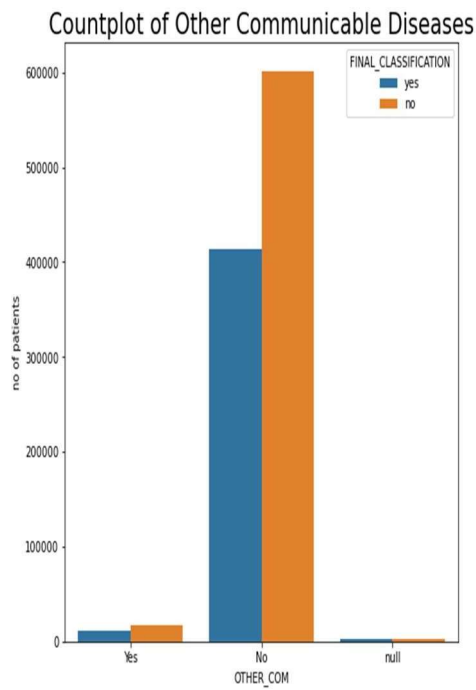


fig 4.8 Other comm. disease

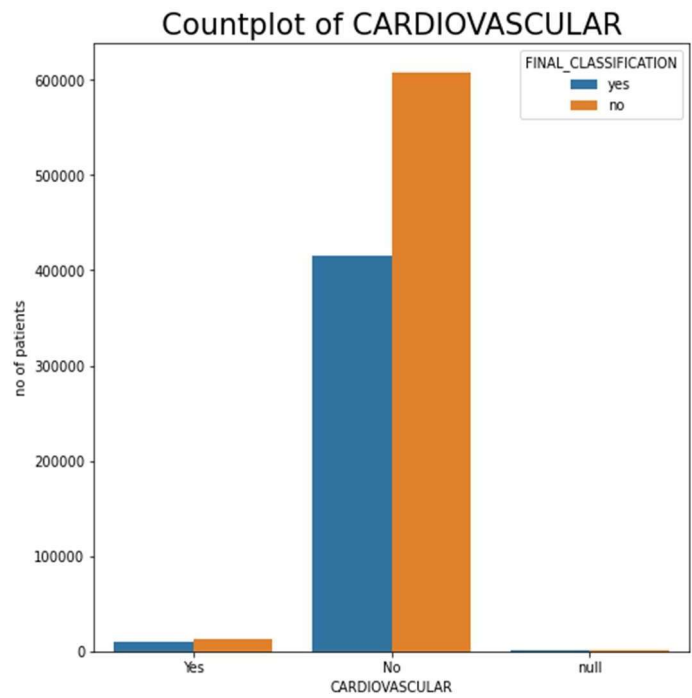


fig 4.9 cardiovascular

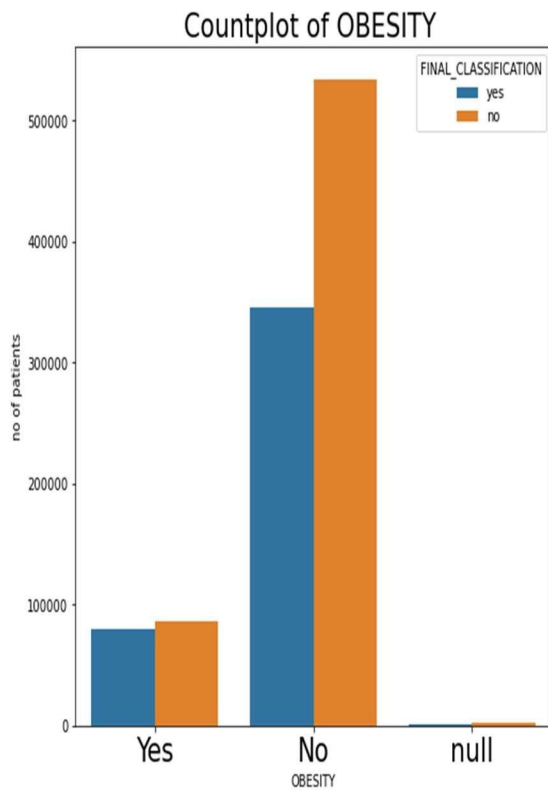


fig 4.10 Obesity

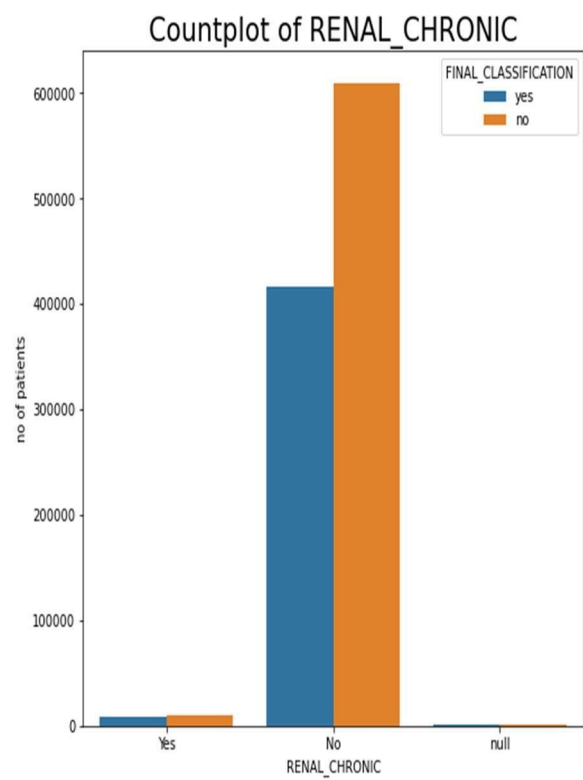


fig 4.11 Renal chronic

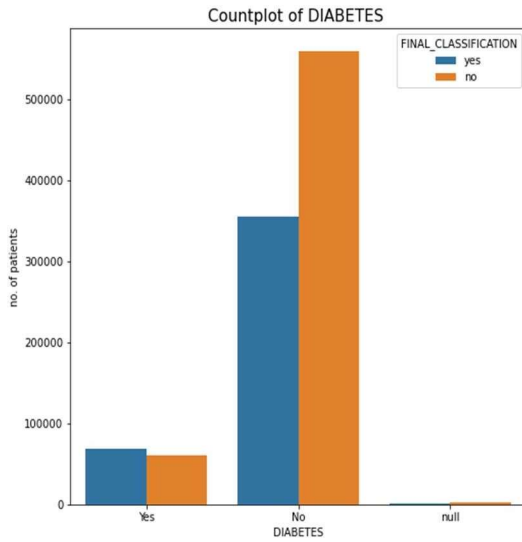


fig 4.12 diabetes

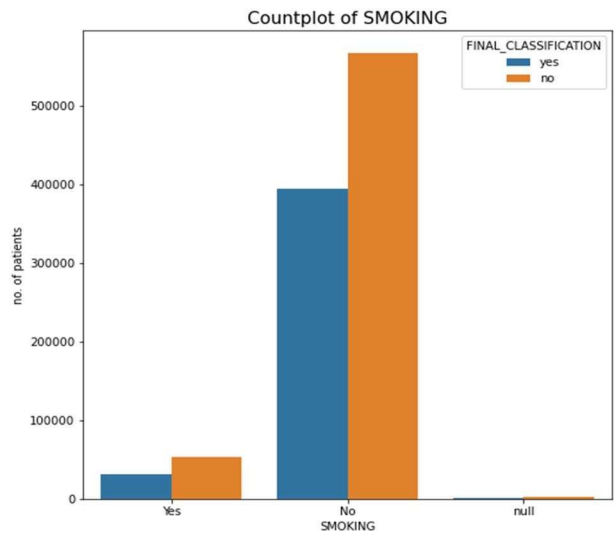


fig 4.13 smoking

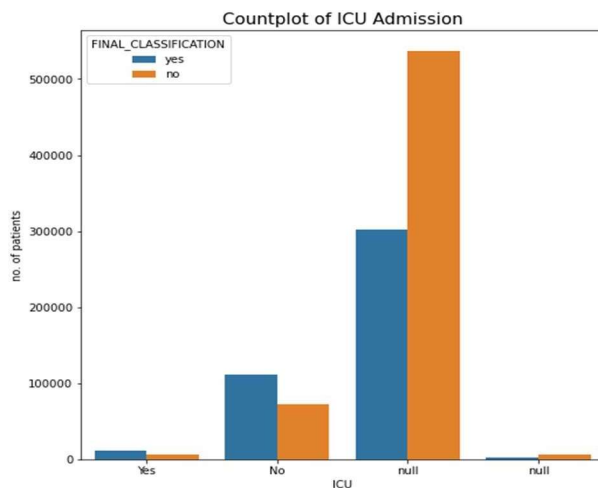


fig 4.14 ICU Admission

After the analysis of individual disease data points, the whole data is taken and label encoded to 0,1 for prediction by machine learning algorithm.

The data frame is taken and splitted into x_set & y_set where y_set is target variable. the data sets are divided for training and testing in ratio 80:20.

X_train, X_test, y_train, y_test= (838860, 13), (838860,), (209715, 13), (209715,)

Now data frame is ready to be used for the machine learning algorithm.

In this Study we use totally 9 Algorithms model to get the accurate prediction of the severity of covid patients ORGANISING EPIDEMIC DATA-n

Organizing descriptive data into tables, graphs, diagrams, maps, or charts provides a rapid, objective, and coherent grasp of the data. Whether the tables or graphs help the investigator understand the data or explain the data in a report or to an audience, their organization should quickly reveal the principal patterns and the exceptions to those patterns. Tables, graphs, maps, and charts all have four elements in common: a title, data, footnotes, and text (Box 6.2). In this chapter, additional guidelines for preparing these data displays will appear where the specific data display type is first applied.

Tables are commonly used for characterizing disease cases or other health events and are ideal for displaying numeric values. In addition to the previously mentioned elements in common to all data displays.

Tables have column and row headings that identify the data type and any units of measurement that apply to all data in that column or row. A well-structured analytical table that is organized to focus on comparisons will help you understand the data and explain the data to others. In arranging analytical tables, you should begin with the arrangement of the data space by following a simple set of guidelines.

Cases are customarily organized in a table called a line-listing. This arrangement facilitates sorting to reorganize cases by relevant characteristics. The line-listing in has been sorted by days between vaccination and onset to reveal the pattern of this important time– event association. Commonly in descriptive epidemiology, you organize cases by frequency of clinical findings. If the disease cause is unknown, this arrangement can assist the epidemiologist in developing hypotheses regarding possible exposures. For example, initial respiratory symptoms might indicate exposure through the upper airways

Components of Statistical Data Displays

- A statistical data display should include, at a minimum,
- A title that includes the what, where, and when that identifies the data, it introduces.
- A data space where the data are organized and displayed to indicate patterns.
- Footnotes that explain any abbreviations used, the data sources, units of measurement, and other necessary details or data.

Text that highlights the main patterns of the data (this text might appear within the table or graphic or in the body of the report).

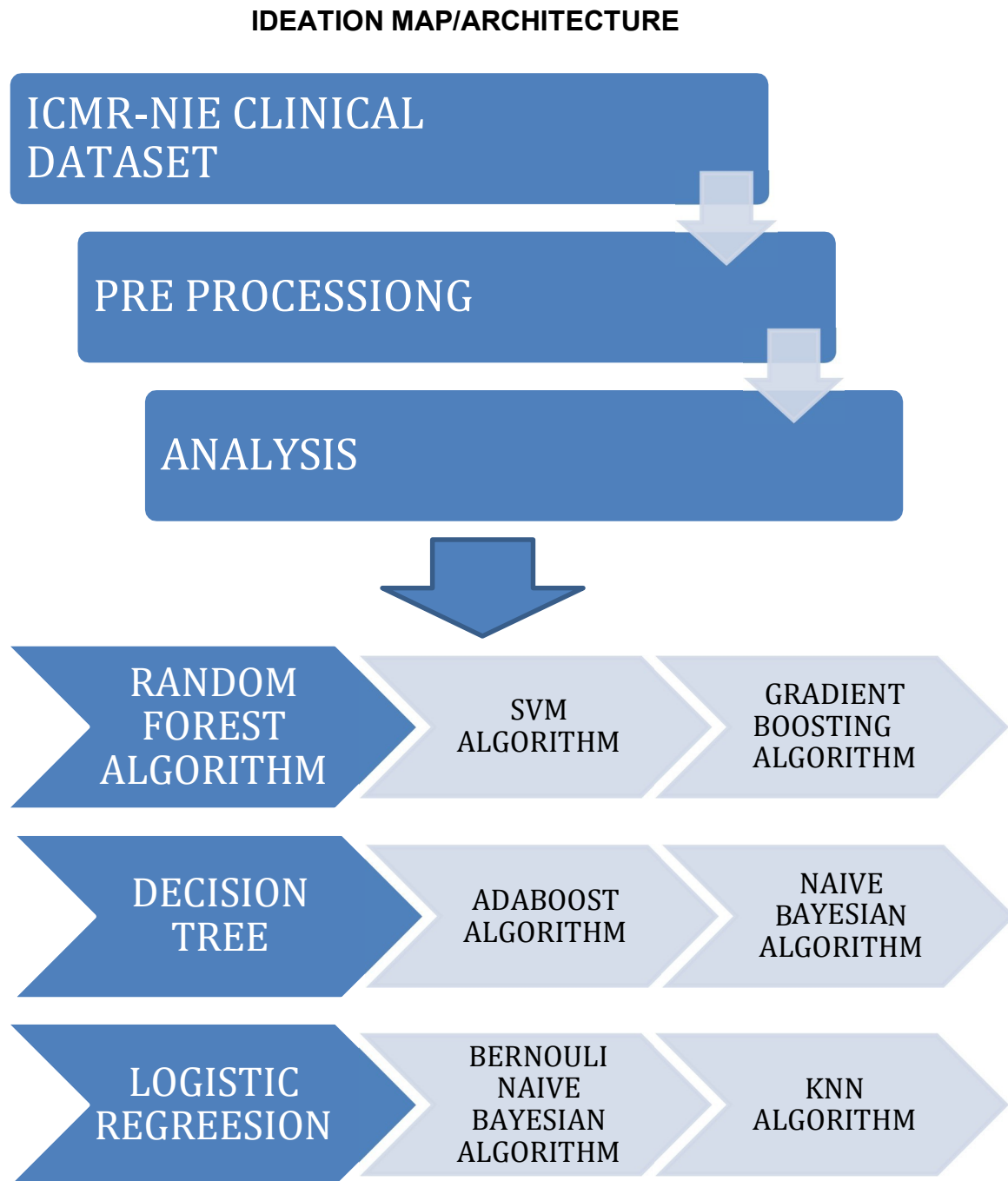
Guidelines for Graphical Data Presentation

- When using transformed data (e.g., logarithmic, normalized, or ranked), represent equal units of the transformed data with equal distances on the axis.
- Represent dependent variables on the vertical scale and independent variables on the horizontal scale.
- Use alternatives to joining data points with a line. Consider instead.
- No line at all (use data markers only).
- A trend line of best fit underlying the data markers.
- A moving average line underlying the data markers.

Aspect ratios (data space width to height) of approximately 2:1 work well. Extreme aspect ratios distort data.

- Scale the graph to fill the data space and to improve resolution. If this means that you must exclude the zero level, exclude it, but note for the reader that this has been done.
- Do not insist on a zero level unless it is an integral feature of the data (e.g., an endpoint).
- Use graphic designs that reveal the data from the broad overview to the fine detail.
- To compare two lines, plot their difference directly.
- Use visually prominent symbols to plot and emphasize the data.
- Make sure overlapping plotting symbols are distinguishable.
- When two or more data sets are plotted in the same data space
- Design point markers and lines for visual discrimination; and
- Differentiate them with labels, legends, or keys.
- To avoid clutter and maintain undistorted comparisons, consider using two or more separate panels for different strata on the same graph.
- When comparing two graphs of the same dependent variable, use scaling that improves comparison and resolution.
- Clearly indicate scale divisions and scaling units.

4.3. Testing Plan of the Proposed Model/System



Contact Diagrams

Contact diagrams are versatile tools for revealing relationships between individual cases in time. In contact diagrams (Figure 6.2, panel A) (5), which are commonly used for visualizing person-to-person transmission, different markers are used to indicate the different groups exposed or at risk.

Epidemic Curves

Epidemic curves (Box 6.5) are histograms of frequency distributions of incident cases of disease or other health events displayed by time intervals. Epidemic curves often have patterns that reveal likely transmission modes. The following sections describe certain kinds of epidemic situations that can be diagnosed by plotting cases on epidemic

4.4. MACHINE LEARNING MODELS FOR PREDICTION SYSTEM

1)RANDOM FOREST ALGORITHM-

"Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.

Random Forest is a machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. Random Forest combines multiple classifiers to solve a complex problem and to improve the performance of the model. use of Random Forest algorithm in this study:

- It takes less training time as compared to other algorithms.
- It predicts output with high accuracy, even for the large dataset it runs efficiently.
- It can also maintain accuracy when a large proportion of data is missing.

Process-

Step 1-

The training and testing data sets X_{train} , X_{test} , y_{train} , y_{test} is taken

Step 2-

Fitting the Random Forest algorithm to the training set. To fit it, we will import the Random Forest Classifier class from the **sklearn.ensemble** library.

Step 3-

Since our model is fitted to the training set, so now we can predict the test result. For prediction, we will create a new prediction vector `y_pred`.

Step 4-

find the accuracy, precision, recall scores and confusion matrix for the random forest algorithm

Step 5-

final step is to visualize the predicted result. To visualize the training set result, we will plot a graph for the Random Forest classifier. The classifier will predict high risk or low risk for the users bases on their disease history

Result-

Accuracy is 0.8355962115251651

precision is 0.8092823171057553

Recall is 0.2673490204354402

Out [13]:

```
array([[109799, 14623],  
       [ 62490, 22803]], dtype=int64)
```

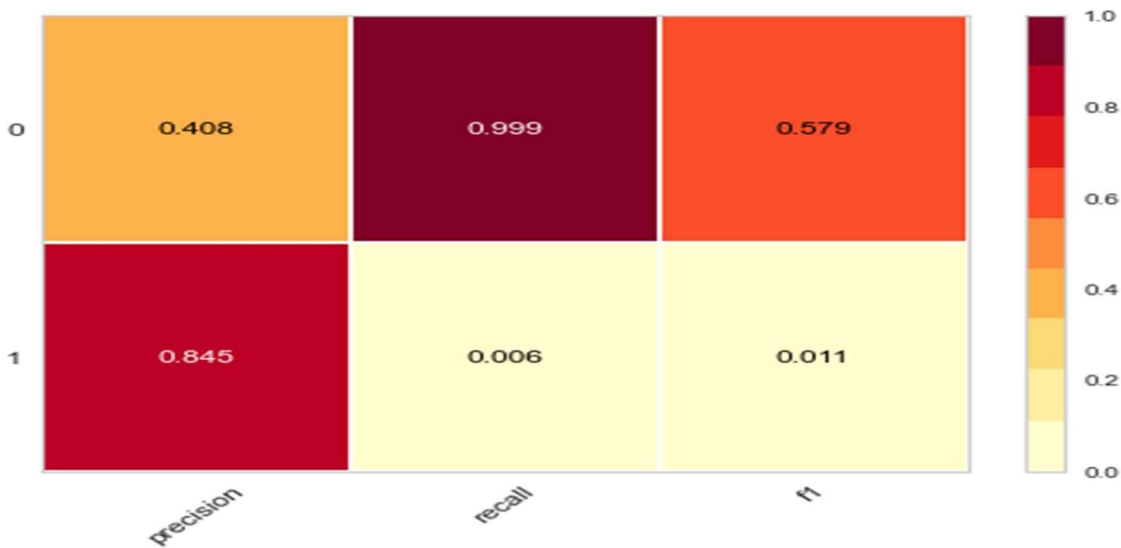


Fig 4.15 random forest

II) SUPPORT VECTOR MACHINE

- The objective of SVM algorithm is to find a hyperplane in an N-dimensional space that distinctly classifies the data points. This algorithm function that takes low dimensional input space and transforms it into higher-dimensional space, ie it converts not separable problem to separable problem, Effective in high dimensional cases
- Its memory efficient as it uses a subset of training points in the decision function called support vectors
- Different kernel functions can be specified for the decision functions and its possible to specify custom kernels
- The math equation for the hyperplane is a linear equation.
- $a_0 + a_1x_1 + a_2x_2 + \dots + a_nx_n$

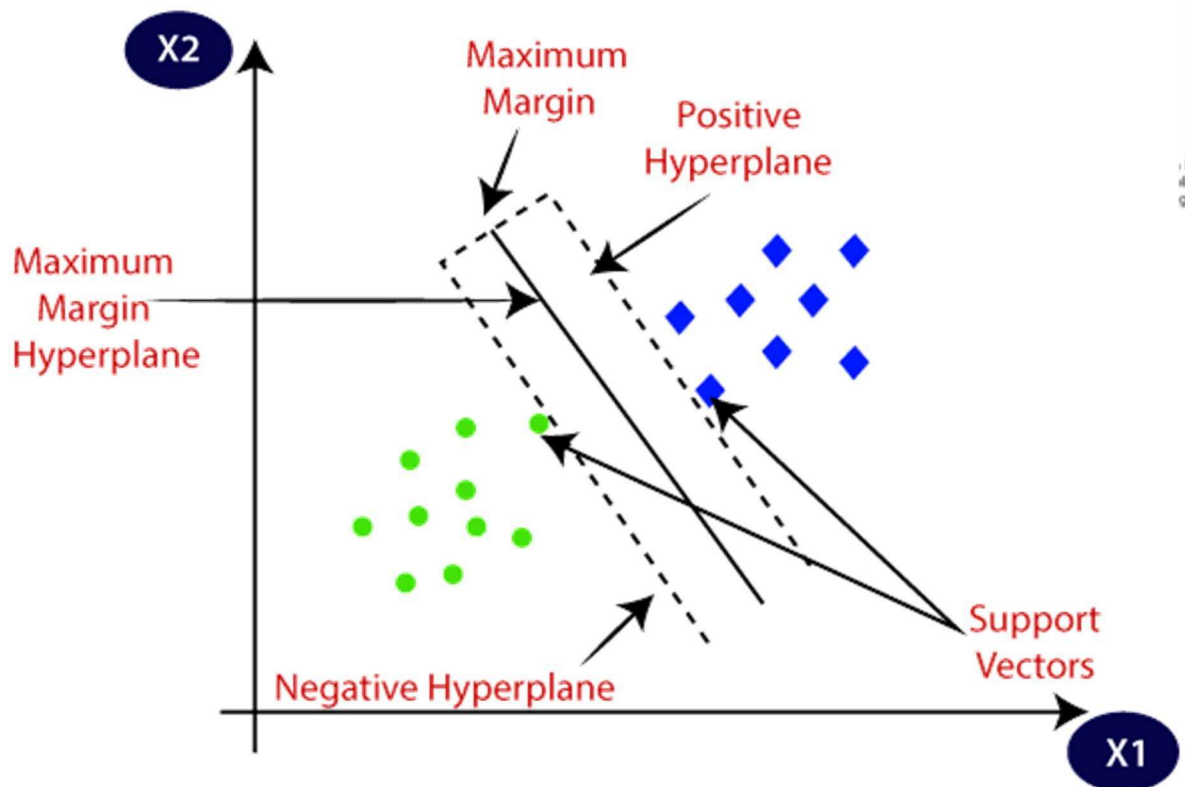


Fig 4.16 svm model

PROCESS:

Step 1-

The training and testing data sets X_{train} , X_{test} , y_{train} , y_{test} is taken

Step 2-

Fitting the SVM algorithm to the training set. To fit it, we will import the SVM algorithm **Classifies** class from the **sklearn.ensemble** library

Step 3-

Since our model is fitted to the training set, so now we can predict the test result. For prediction, we will create a new prediction vector y_{pred} .

Step 4-

Find the accuracy, precision, recall scores and confusion matrix for the SVM algorithm

Step 5-

Final step is to visualize the predicted result. To visualize the training set result, we will plot a graph for the SVM algorithm. The classifier will predict high risk or low risk for the users bases on their disease history

Result-

Accuracy is 0.8249758157499464

precision is 0.8389479721817269

Recall is 0.8595103759785246

array ([[24863, 60430],
[17480, 106942]], dtype=int64)

Out[83]

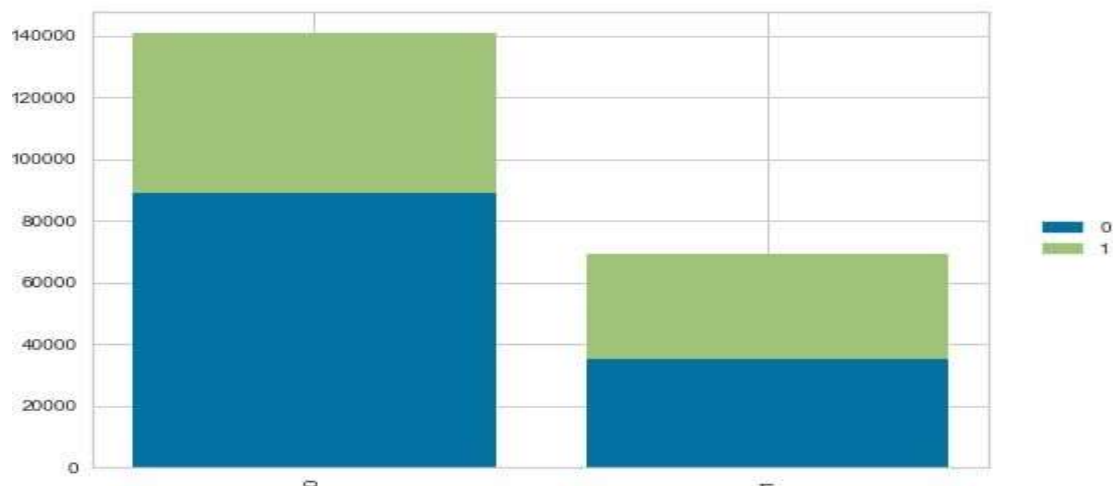


Fig 4.17 svm graph

III) DECISION TREE

A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks. It has a hierarchical, tree structure, which consists of a root node, branches, internal nodes and leaf nodes. The decision tree algorithm is a popular machine learning algorithm that is widely used in various applications. Here are some of the common uses of the decision tree algorithm:

Classification:

Decision trees can be used for classification tasks where the goal is to assign a label or category to a new input based on its features. For example, a decision tree can be used to classify emails as spam or not spam based on their content and metadata.

Regression:

Decision trees can also be used for regression tasks where the goal is to predict a continuous target variable based on its features. For example, a decision tree can be used to predict the price of a house based on its features such as location, size, and number of rooms.

Feature selection:

Decision trees can be used for feature selection, where the most important features for the task can be identified based on their contribution to the decision tree model. This can help to reduce the dimensionality of the problem and improve the performance of the model.

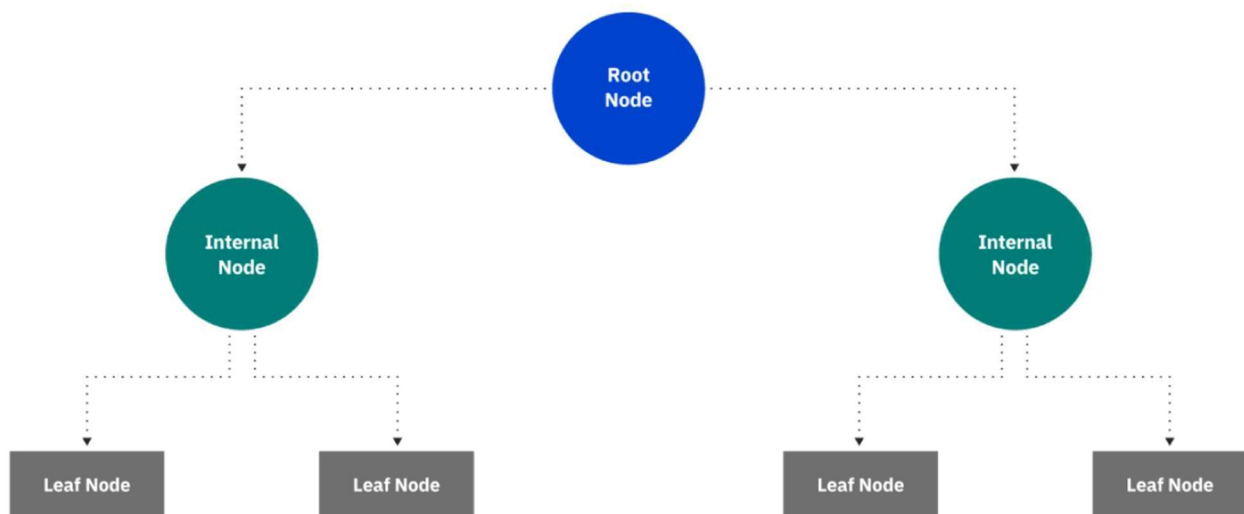


Fig 4.18 d-tree flow chart

PROCESS-

Step 1-

the training and testing data sets X_train, X_test, y_train, y_test is taken

Step 2-

Fitting the decision tree algorithm to the training set. To fit it, we will import the decision tree algorithm **Classifies** class from the **sklearn.ensemble** library

Step 3-

Since our model is fitted to the training set, so now we can predict the test result. For prediction, we will create a new prediction vector y_pred.

Step 4-

Find the accuracy, precision, recall scores and confusion matrix for the SVM algorithm

Step 5-

Final step is to visualize the predicted result. To visualize the training set result, we will plot a graph for the decision tree algorithm. The classifier will predict high risk or low risk for the users based on their disease history

RESULT-

Accuracy is 0.8124855875831485

precision is 0.8381233115136457

Recall is 0.8751667711497967

Out [54]:

```
array([[ 23542,  61751],  
       [ 15532, 108890]], dtype=int64)
```

IV) GRADIENT BOOSTING CLASSIFIER

Gradient Boosting **is** a popular boosting algorithm. In gradient boosting, each predictor corrects its predecessor's error. In contrast to Adaboost, the weights of the training instances are not tweaked, instead, each predictor is trained using the residual errors of predecessor as labels.

There is a technique called the Gradient Boosted Trees whose base learner is CART (Classification and Regression Trees).

PROCESS-

Step 1-

the training and testing data sets X_{train} , X_{test} , y_{train} , y_{test} is taken

Step 2-

Fitting The Gradient Boosting Classifier To The Training Set. To Fit It, We Will Import Gradient Boosting Classifier Class From The Sklearn.Ensemble Library

Step 3-

Since our model is fitted to the training set, so now we can predict the test result. For prediction, we will create a new prediction vector y_{pred} .

Step 4-

Find the accuracy, precision, recall scores and confusion matrix for the SVM algorithm

Step 5-

Final step is to visualize the predicted result. To visualize the training set result, we will plot a graph for the **GRADIENT BOOSTING CLASSIFIER**. The classifier will predict high risk or low risk for the users bases on their disease history RESULT-

Accuracy is 0.8358915671268149

precision is 0.8378498373783206

Recall is 0.8937085081416469

Out [58]:

```
array([[ 22159, 63134],  
       [ 13225, 111197]], dtype=int64)
```

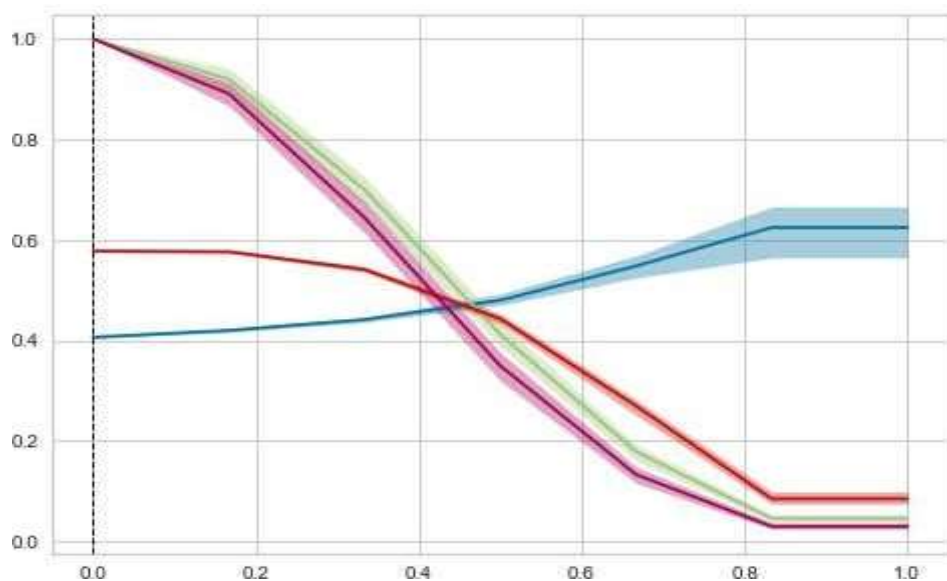


Fig 4.19 gradient boost

V) NAIVE BAYES ALGORITHM

The naive Bayes Algorithm is one of the popular classification machine learning algorithms that helps to classify the data based upon the conditional probability values computation. It implements the Bayes theorem for the computation and used class levels represented as feature values or vectors of predictors for classification. Naive Bayes Algorithm is a fast algorithm for classification problems. This algorithm is a good fit for real-time prediction, multi-class prediction, recommendation system, text classification, and sentiment analysis use cases. Naive Bayes Algorithm can be built using Gaussian, Multinomial and Bernoulli distribution. This algorithm is scalable and easy to implement for a large data set. It helps to calculate the posterior probability $P(c|x)$ using the prior

probability of class $P(c)$, the prior probability of predictor $P(x)$, and the probability of predictor given class, also called as likelihood $P(x|c)$.

PROCESS-

Step 1-

The training and testing data sets X_{train} , X_{test} , y_{train} , y_{test} is taken

Step 2-

Fitting the naive Bayes to the training set. To fit it, we will import NIAVE Bayes
Classifies class from the sklearn.ensemble library

Step 3-

Since our model is fitted to the training set, so now we can predict the test result. For prediction, we will create a new prediction vector y_{pred} .

Step 4-

Find the accuracy, precision, recall scores and confusion matrix for the SVM algorithm

Step 5-

Final step is to visualize the predicted result. To visualize the training set result, we will plot a graph for the Naive Bayes. The classifier will predict high risk or low risk for the users bases on their disease history

RESULT-

Accuracy is 0.8172186062036573

precision is 0.8463459103234747

Recall is 0.7835350661458584

f1 score is 0.708359218609788

Out[64]:

```
array([[31951, 53342],  
       [26933, 97489]], dtype=int64)
```

VI) BERNOULLI NAÏVE BAYES

Bernoulli Naive Bayes is one of the variants of the Naive Bayes algorithm in machine learning. It is very useful to be used when the dataset is in a binary distribution where the output label is present or absent. The main advantage of this algorithm is that it only accepts features in the form of binary values.

1. It is very fast compared to other classification algorithms.
2. Sometimes machine learning algorithms do not work well if the dataset is small, but this is not the case with this algorithm because it gives more accurate results compared to other classification algorithms in the case of a small dataset.
3. It is fast and can also handle irrelevant features easily.

PROCESS-

Step 1-

The training and testing data sets `X_train`, `X_test`, `y_train`, `y_test` is taken

Step 2-

Fitting the Bernoulli Naive Bayes to the training set. To fit it, we will import Bernoulli Naive Bayes Classifier class from the `sklearn.naive_bayes` library

Step 3-

Since our model is fitted to the training set, so now we can predict the test result. For prediction, we will create a new prediction vector `y_pred`.

Step 4-

Find the accuracy, precision, recall scores and confusion matrix for the SVM algorithm.

Step 5-

Final step is to visualize the predicted result. To visualize the training set result, we will plot a graph for the Bernoulli Naive Bayes. The classifier will predict high risk or low risk for the users based on their disease history

RESULTS-

Accuracy is 0.8268958157499464

precision is 0.8389479721817269

Recall is 0.8595103759785246

Out[83]:

```
array([[ 24863, 60430],  
       [ 17480, 106942]], dtype=int64)
```

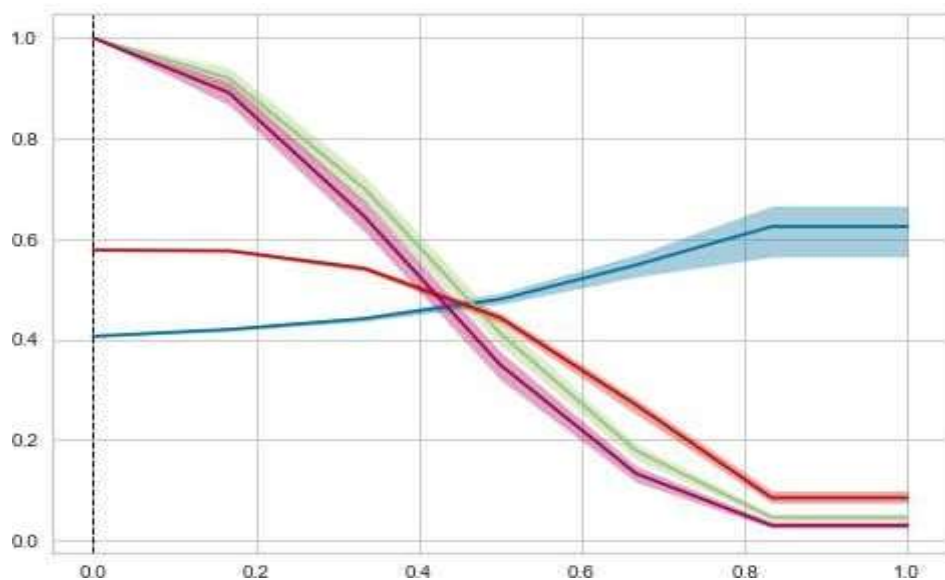


Fig 4.20 bernouli naive

VII) ADAPTIVE BOOST CLASSIFIER

Ada-boost or Adaptive Boosting is one of ensemble boosting Classifier combines multiple classifiers to increase the accuracy of classifiers. AdaBoost is an iterative ensemble

method. AdaBoost classifier builds a strong classifier by combining multiple poorly performing classifiers so that you will get high accuracy strong classifier. The basic concept behind Adaboost is to set the weights of classifiers and training the data sample in each iteration such that it ensures the accurate predictions of unusual observations. Any machine learning algorithm can be used as base classifier if it accepts weights on the training set. Adaboost should meet two conditions:

1. The classifier should be trained interactively on various weighed training examples.
2. In each iteration, it tries to provide an excellent fit for these examples by minimizing training error.

$$w(x_i, y_i) = \frac{1}{N}, \quad i = 1, 2, \dots, n$$

It works in the following steps:

1. Initially, Adaboost selects a training subset randomly. It iteratively trains the AdaBoost machine learning model by selecting the training set based on the accurate prediction of the last training.
2. It assigns the higher weight to wrong classified observations so that in the next iteration these observations will get the high probability for classification.
3. Also, It assigns the weight to the trained classifier in each iteration according to the accuracy of the classifier. The more accurate classifier will get high weight.
4. This process iterates until the complete training data fits without any error or until reached to the specified maximum number of estimators.
5. Since our model is fitted to the training set, so now we can predict the test result. For prediction, we will create a new prediction vector y_{pred} .
6. find the accuracy, precision, recall scores and confusion matrix for the SVM algorithm

RESULT-

Accuracy is 0.8379656748444317

precision is 0.8380804265718729

Recall is 0.9002266480204465

Out[88]:

array([[21762, 63531],

[12414, 112008]], dtype=int64)

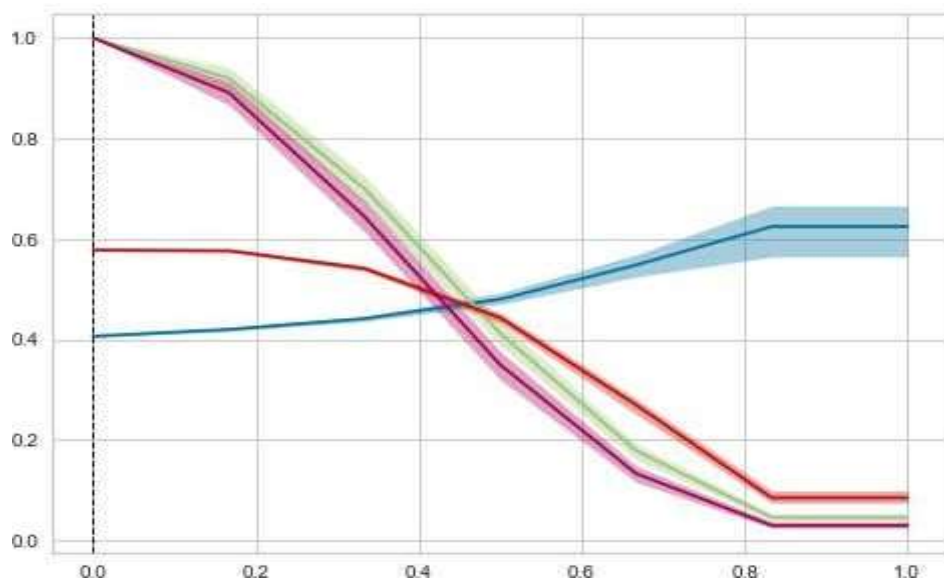


Fig 4.21 adaptive boost

VIII) LOGISTIC REGRESSION-

This type of statistical model (also known as *logit model*) is often used for classification and predictive analytics. Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given dataset of independent variables. Since the outcome is a probability, the dependent variable is bounded between 0 and 1. In logistic regression, a logit transformation is applied on the odds—that is, the probability of success divided by the probability of failure. This is also commonly known

as the log odds, or the natural logarithm of odds, and this logistic function is represented by the following formulas:

$$\text{Logit}(\pi) = 1/(1 + \exp(-\pi))$$

$$\ln(\pi/(1-\pi)) = \text{Beta}_0 + \text{Beta}_1 X_1 + \dots + \text{Beta}_k X_k$$

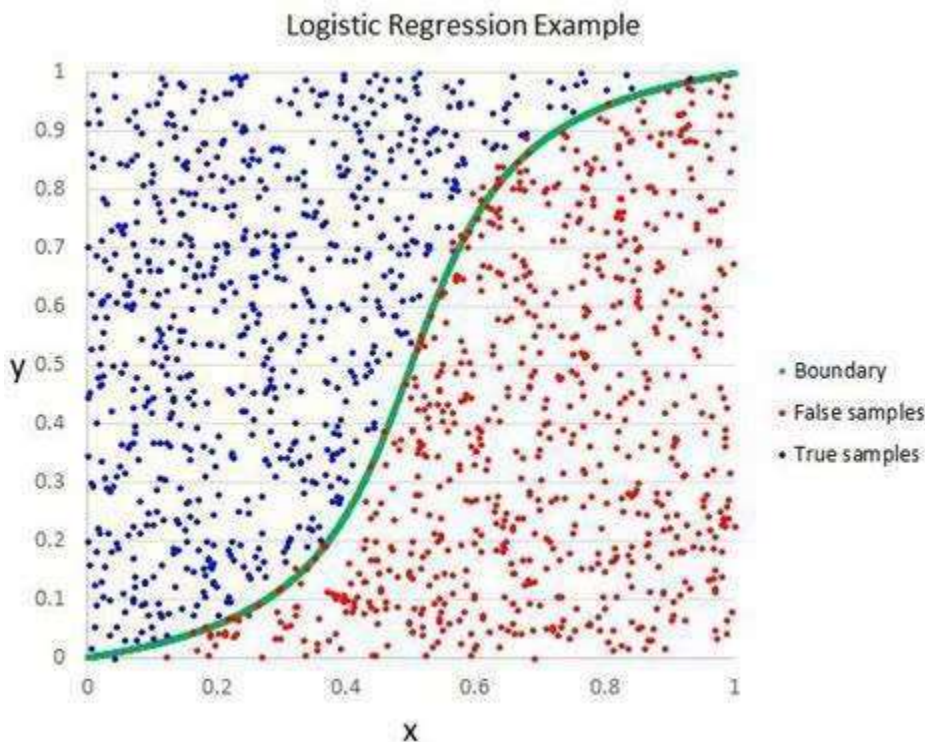


Fig 4.22 logistic model

In this logistic regression equation, $\text{logit}(\pi)$ is the dependent or response variable and x is the independent variable. The beta parameter, or coefficient, in this model is commonly estimated via maximum likelihood estimation (MLE). This method tests different values of beta through multiple iterations to optimize for the best fit of log odds. All of these iterations produce the log likelihood function, and logistic regression seeks to maximize this function to find the best parameter estimate. Once the optimal coefficient (or coefficients if there is more than one independent variable) is found, the conditional probabilities for each observation can be calculated, logged, and summed together to yield a predicted probability. For binary classification, a probability less than .5 will predict 0 while a probability greater than 0 will predict 1.

RESULT-

Accuracy is 0.8276657768744317

precision is 0.8704265718729

Recall is 0.9000165480203495

Out[88]:

```
array([[ 21762,  63531],  
       [ 12414, 112008]], dtype=int64)
```

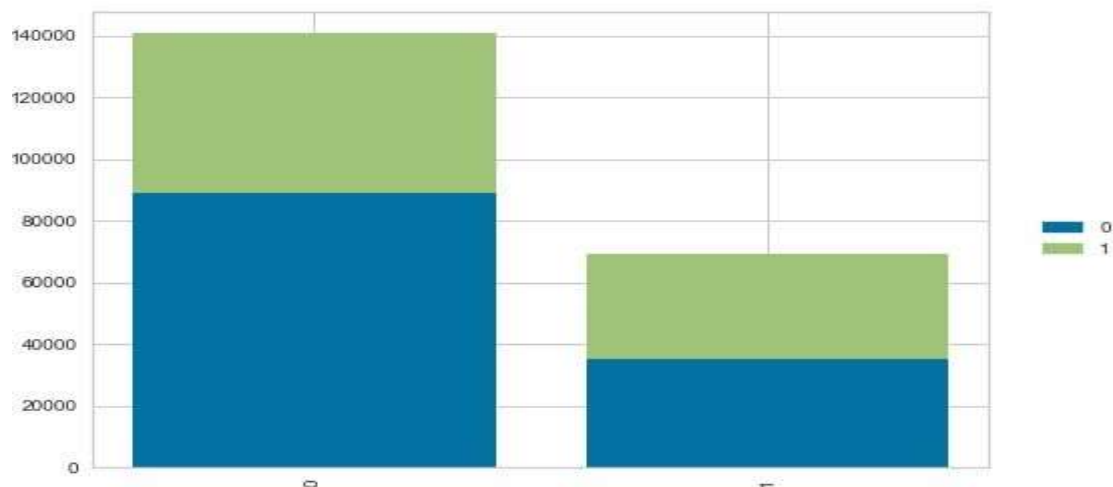


Fig 4.23 logistic regression

IX) KNN ALGORITHM-

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset. KNN algorithm at the training phase just stores the

dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data

The K-NN working can be explained on the basis of the below algorithm:

- **Step-1:** Select the number K of the neighbors
- **Step-2:** Calculate the Euclidean distance of K number of neighbors
- **Step-3:** Take the K nearest neighbors as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbors, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbor is maximum
- **Step-6:** Since our model is fitted to the training set, so now we can predict the test result. For prediction, we will create a new prediction vector `y_pred`.
- Step 7-find the accuracy, precision, recall scores and confusion matrix for the SVM algorithm

RESULTS-

Accuracy is 0.7860286579405384

precision is 0.5889751125653313

Recall is 0.395976223136717

Out[14]:

```
array([[89125, 35297],  
       [51519, 33774]], dtype=int64)
```

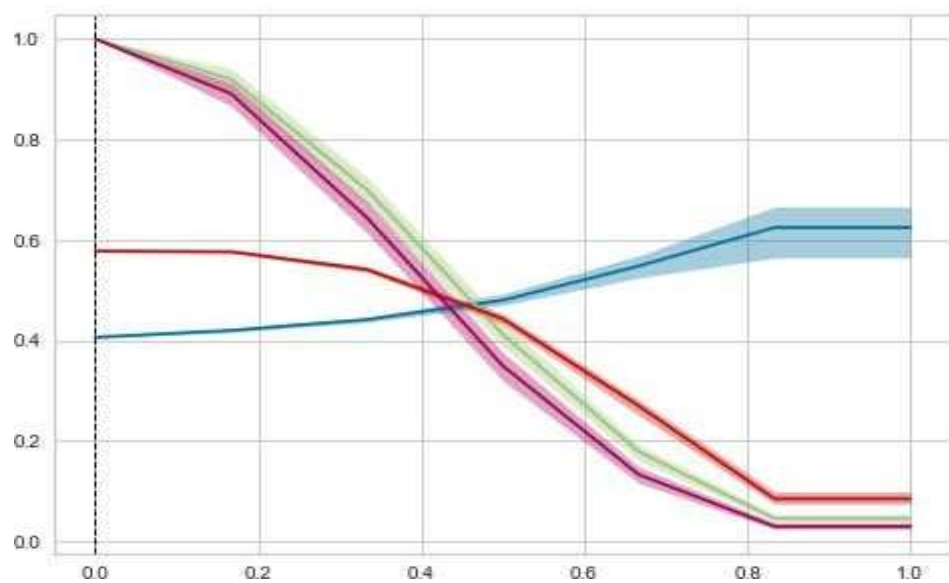


Fig 4.24 knn classifier

4.5. FEATURE SCALING-

Feature Scaling is a technique to standardize the independent features present in the data in a fixed range. It is performed during the data pre-processing. Feature scaling is the process of normalizing the range of features in a dataset.

Types of Feature Scaling:

There are several methods for scaling the features, including normalization, standardization, log transformation, and min-max scaling.

Normalization:

Normalization scales the values to a range of 0 to 1, and it is useful when the distribution of the feature is not Gaussian. Normalization is calculated as $(x - x_{\min}) / (x_{\max} - x_{\min})$, where x is the original value, $x_{\min}$ is the minimum value in the feature, and $x_{\max}$ is the maximum value.

Why Feature Scaling is Important: The input features in disease prediction systems may vary widely in magnitude, and this can result in biased models that perform poorly on

certain input features. For example, consider a system that predicts diabetes based on blood glucose levels and body mass index (BMI). The blood glucose levels may range from 70 to 400 mg/dL, while the BMI may range from 20 to 40. Without feature scaling, the model may give more weight to blood glucose levels, resulting in an underperformance of BMI

Standardization:

Standardization scales the values to have a mean of 0 and a standard deviation of 1. It is useful when the distribution of the feature is Gaussian. Standardization is calculated as $(x - \text{mean}) / \text{standard deviation}$, where x is the original value, mean is the mean of the feature, and standard deviation is the standard deviation of the feature.

Log Transformation:

Log transformation is used when the data is skewed towards higher values. It transforms the data to a logarithmic scale, making it easier to analyze and model.

Min-Max Scaling:

Min-max scaling scales the values to a range of 0 to 1 like normalization, but it preserves the relative distance between the values. It is calculated as $(x - x_{\min}) / (x_{\max} - x_{\min})$, where x is the original value, $x_{\min}$ is the minimum value in the feature, and $x_{\max}$ is the maximum value.

Feature scaling is a critical step in disease prediction systems that can help improve the accuracy of the models. The choice of scaling method depends on the distribution of the feature and the goals of the model. It is essential to consider the impact of feature scaling on the performance of the model and to select the appropriate method accordingly.

From the above algorithms, features leading to the above decision are studied, then the individual feature is carefully analyzed and the least important features are removed from the data frame. the least important data column feature is deleted from the data frame. the least important feature are with feature value less than 2.

```
col_feature
Out[22]: AGE          44.988930
PNEUMONIA    32.605832
DIABETES      3.734453
OBESITY       2.901783
HYPERTENSION  2.845703
SEX           2.584840
OTHER_COM     1.917975
SMOKING       1.611735
CARDIOVASCULAR 1.454012
RENAL_CHRONIC 1.442378
IMMU_SUPR    1.404727
COPD          1.370752
ASTHMA        1.136880
dtype: float64

In [23]: X_hp_train=X_train.drop(['SMOKING', 'OTHER_COM', 'RENAL_CHRONIC',
'IMMU_SUPR', 'CARDIOVASCULAR', 'COPD', 'ASTHMA'], axis=1)

X_hp_test=X_test.drop(['SMOKING', 'OTHER_COM', 'RENAL_CHRONIC',
'IMMU_SUPR', 'CARDIOVASCULAR', 'COPD', 'ASTHMA'], axis=1)
```

Fig.4.25 Feature Scaling

A correlation matrix is a mathematical representation of the relationships between variables in a dataset. It shows how strongly one variable is related to another and in what direction. In the context of a disease prediction system, a correlation matrix can be a useful tool for understanding the relationship between different risk factors and the likelihood of developing the disease.

This report will explore the use of correlation matrices in disease prediction systems and how they can be used to improve the accuracy of the prediction model. Correlation Matrix in Disease Prediction Systems:

A disease prediction system typically relies on multiple risk factors to assess the likelihood of a person developing a particular disease. These risk factors can include age, sex, medical history, lifestyle factors, and other health indicators.

To create an accurate prediction model, it is essential to understand how these risk factors are related to one another and how they contribute to the overall risk of developing the disease.

A correlation matrix can provide valuable insights into these relationships. By analyzing the correlations between different risk factors, we can identify which factors are strongly related to the likelihood of developing the disease and which ones have little or no impact.

This information can then be used to improve the accuracy of the prediction model by adjusting the weights given to different risk factors.

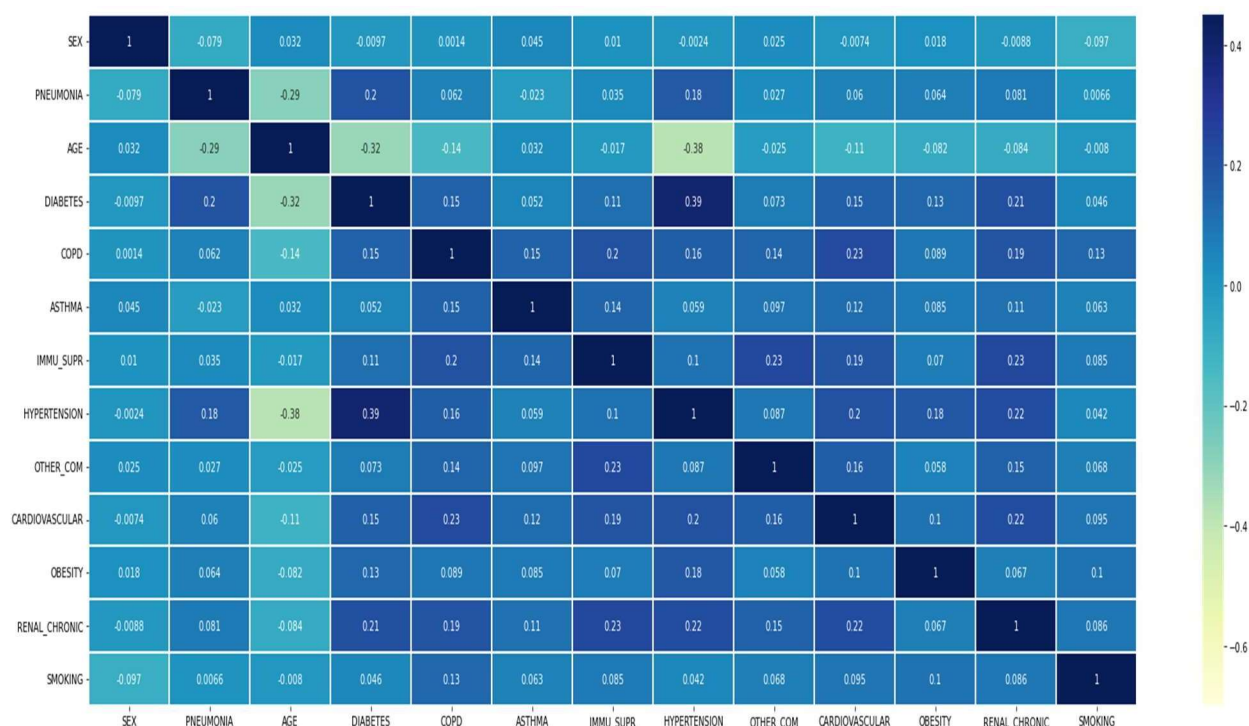


Fig 4.26 Correlation Matrix Before Feature Scaling

The use of a correlation matrix in a disease prediction system can be an effective tool for improving the accuracy of the prediction model. By analyzing the correlations between different risk factors, we can identify the most significant contributors to the overall risk of developing the disease and adjust the weighting of different risk factors accordingly. This information can also be used to develop new insights into the causes and mechanisms of the disease, leading to new treatments and prevention strategies. The correlation matrix is a square matrix that contains the correlation coefficients between all possible pairs of variables in the dataset. Here are some of the common uses of the correlation matrix:

Identify the strength and direction of the relationship between variables: The correlation matrix helps to identify the strength and direction of the relationship between two or more variables. A correlation coefficient can range from -1 to 1, where -1 indicates a perfect

negative correlation, 0 indicates no correlation, and 1 indicates a perfect positive correlation.

Feature selection: Correlation matrix can be used in feature selection, where highly correlated variables can be removed from the dataset to reduce the dimensionality of the problem and improve the performance of the model.

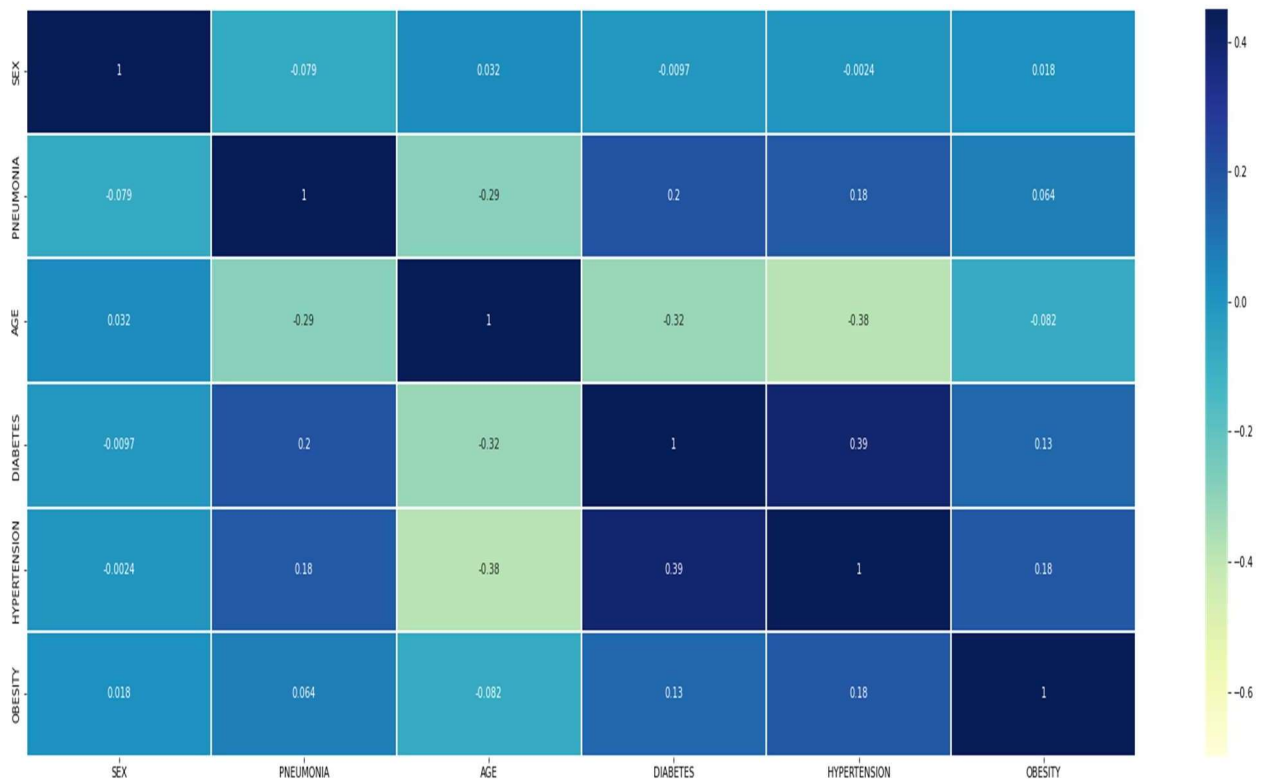


Fig 4.27 Correlation Matrix After Feature Scaling

After feature scaling the standard data sets are again made to run on all the above nine algorithms. The average accuracy after feature scaling is-

Accuracy is 0.6597175935879786

precision is 0.6584712662077797

Recall is 0.28747552554136916

Out[25]:

```
array([[113782, 15980],
       [ 65238, 20655]], dtype=int64)
```

4.6. DESCRIPTION OF THE SURVEILLANCE SYSTEM

Disease surveillance is the process of monitoring and tracking the occurrence of disease within a population. It is a critical public health function that enables timely detection and response to outbreaks, reduces disease transmission, and minimizes the impact of epidemics on human health. Disease surveillance is vital to prevent the spread of communicable diseases, including infectious diseases, zoonotic diseases, and chronic illnesses.

The aim of this project is to develop a disease surveillance system that will enable the monitoring and tracking of diseases in a population. The system will be designed to facilitate the collection, analysis, and dissemination of health-related data from various sources, including health facilities, laboratories, and other relevant organizations.

Objectives:

The objectives of this project are as follows:

1. To develop a disease surveillance system that will enable the monitoring and tracking of diseases in a population.
2. To design a system that will facilitate the collection, analysis, and dissemination of health-related data from various sources.
3. To develop a user-friendly interface that will enable health workers to easily access and use the system.
4. To establish a mechanism for the timely detection and response to disease outbreaks.
5. To develop a system that will enable the generation of timely and accurate reports on disease prevalence, incidence, and mortality.

4.7. Methodology-Surveillance System:

The development of the disease surveillance system will involve the following steps:

Needs assessment:

A needs assessment will be conducted to identify the requirements and priorities of stakeholders, including health workers, public health officials, and policymakers. The needs assessment will involve surveys, interviews, and focus group discussions.

1. System design:

Based on the findings of the needs assessment, a system design will be developed. The system design will include specifications for data collection, analysis, and dissemination, as well as the development of a user-friendly interface.

2. System development:

The disease surveillance system will be developed using appropriate software development methodologies. The system will be designed to integrate with existing health information systems, such as electronic medical records and laboratory information systems.

3. System testing:

The disease surveillance system will be tested to ensure that it meets the requirements of stakeholders and is effective in monitoring and tracking diseases in a population. The system will be tested for usability, reliability, and functionality.

4. System deployment:

The disease surveillance system will be deployed in selected health facilities and organizations. Training and support will be provided to health workers to ensure that they are able to effectively use the system.

5. System evaluation: The disease surveillance system will be evaluated to assess its impact on disease monitoring and tracking. The evaluation will involve the analysis of system usage data, user feedback, and disease incidence data.

4.8. Surveillance System Dashboard

The code starts by defining a variable `ctx` which represents the canvas element in the HTML with the id `myChart`. This is done using the `document.getElementById` method. The context of the canvas is then obtained using the `getContext` method.

A new Chart object is then created using the Chart constructor function. The chart type is specified as 'bar' and the data for the chart is provided in the data property of the configuration object.

The data property includes an array of labels that correspond to the different serotypes of *Streptococcus pneumoniae*. It also includes an array of datasets, where each dataset corresponds to a different year.

Each dataset includes properties such as label which is the label for the dataset, the type which specifies the type of chart (in this case, 'bar'), backgroundColor which sets the color of the bars, borderColor which sets the color of the bar borders, borderWidth which sets the width of the bar borders, fill which specifies whether to fill the bars with color, yAxisID which specifies the y-axis to which the dataset corresponds, and data which is an array of values corresponding to the frequency of each serotype for the given year.

The configuration object also includes an options property which includes various options for the chart. The tooltips property specifies that the tooltips should display the colors of the bars.

The scales property includes xAxes and yAxes arrays, where each element in the array represents an x-axis or a y-axis for the chart.

Each axis includes various options such as stacked which specifies whether the bars should be stacked, scaleLabel which sets the label for the axis, and gridLines which specifies whether to display grid lines on the axis.

The responsive property specifies whether the chart should be responsive, and the maintain Aspect Ratio property specifies whether to maintain the aspect ratio of the chart.

The legend property specifies whether to display the legend, and the legendCallback property is a function that generates the HTML for the legend.

The code also includes a function called download which creates a link that allows the user to download the chart as an image.

The function creates a new element and sets the download property to the desired file name and the href property to the source of the chart image. Finally, the function clicks the link to initiate the download.

the code provided creates a bar chart using the Chart.js library to display the frequency of different serotypes of Streptococcus pneumonia over the years 2019, 2020, and 2021.

The chart includes a legend and the option to download the chart as an image. The code is well-organized and includes various configuration options to customize the chart's appearance and behavior.

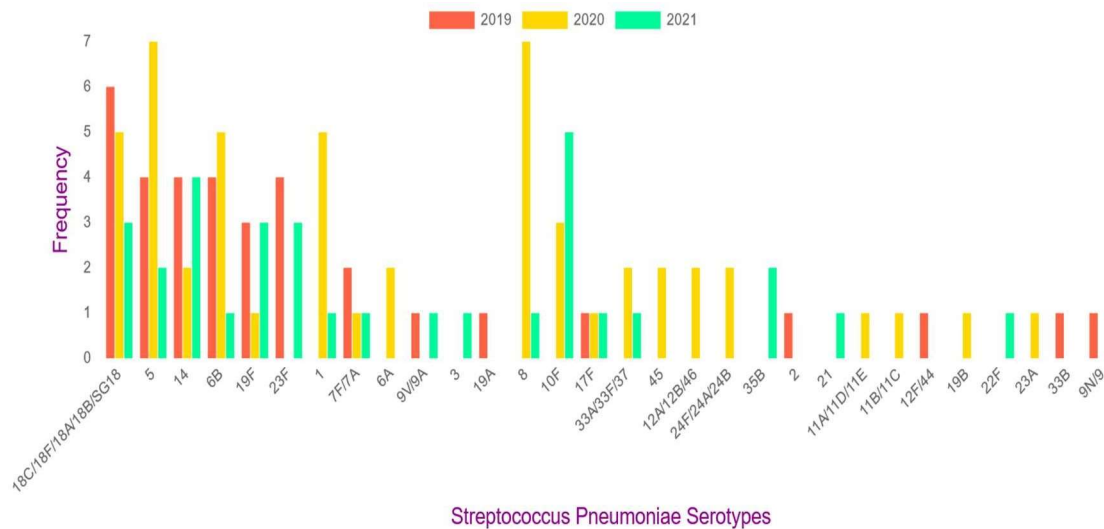


Fig 4.28 streptococcus pneumonia serotypes

The code creates a chart that shows the prevalence of meningitis cases over time, along with the number of suspected meningitis cases.

The chart has several options and features, including tooltips, legends, and the ability to download the chart as an image. In this algorithm report, we will explain how the code works, the different components of the chart, and how the options and features are implemented.

The following report provides an algorithm analysis of the code used to create a chart using Chart.js library. The code provides a visualization of the prevalence of suspected meningitis cases over a period of three years.

The chart comprises two datasets, the prevalence of suspected meningitis cases shown as a line graph and the number of enrolled cases shown as a stacked bar graph.

The chart uses the Chart.js library and is embedded in an HTML document. The code comprises a data section and an options section, which define the chart's properties.

Data Section

The data section defines the chart's data and comprises two datasets. The first dataset shows the prevalence of suspected meningitis cases, displayed as a line graph. The second dataset shows the number of enrolled cases and is displayed as a stacked bar graph. The labels property of the data section provides the labels for the x-axis, which show the months and the years.

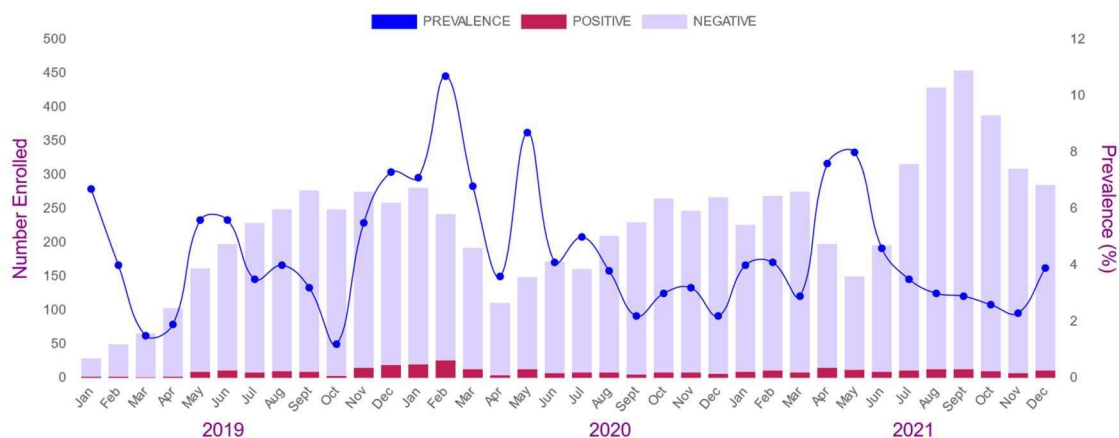


fig 4.29. Laboratory Confirmed cases

The code is used to create a bar chart using Chart.js library. The chart shows the frequency of different bacteria (*S. pneumoniae*, *H. influenzae*, and *N. meningitidis*) over a period of three years (2019, 2020, and 2021) on the x-axis. The y-axis represents the frequency of each bacterium.

The code initializes a chart object by providing a canvas element (with id "myChart") and specifying the type of chart as a bar chart.

The data for the chart is specified as an array of labels and an array of datasets. Each dataset contains the label of the bacteria, the color for the bars, the data points, and the y-axis on which to plot the data.

The options for the chart include the tooltips, x-axis and y-axis scales, legend settings, and the callback function for generating the legend. The legend is generated using the "legendCallback" function, which creates an unordered list of legend items for each dataset.

Prevalence refers to the proportion or percentage of individuals in a population who have a particular disease or condition at a given point in time or over a specific period.

It is a measure of the disease burden in a population and is calculated by dividing the number of individuals with a specific disease or condition by the total population at risk.

Prevalence is an important epidemiological measure that helps to determine the magnitude and distribution of a disease in a population, and it is commonly used to estimate the burden of chronic diseases, infectious diseases, and other health conditions.

It can also be used to track changes in the occurrence of a disease over time and to compare disease burden across different populations or regions.

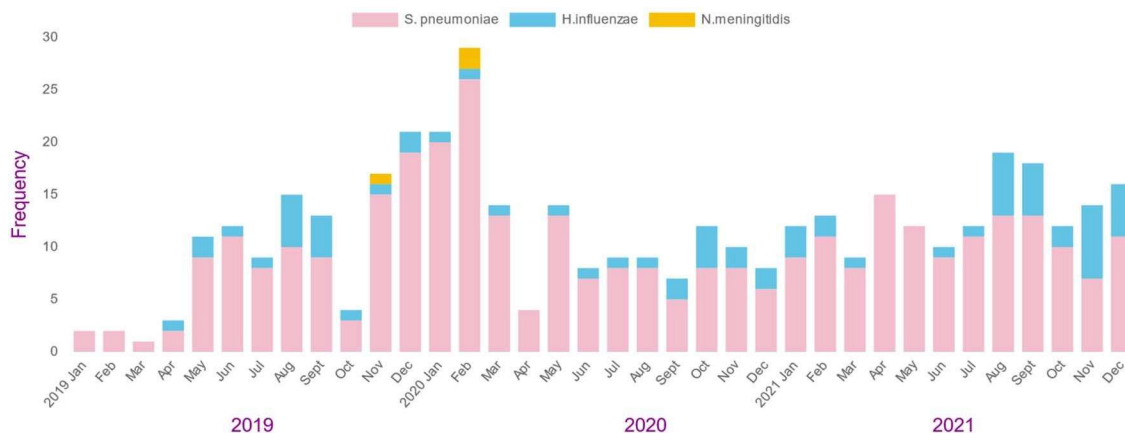


Fig.4.30 Distribution of Pathogen

The interactive map is created using the AmCharts library. The map displays areas of India and contains various types of markers, such as rectangles and circles, each with its own color and label. The markers represent different healthcare facilities, including the ICMR-National Institute of Epidemiology in Tamil Nadu, which is labeled with the acronym "NIE," and the Kanchi Kamakoti Childs Trust Hospital in Chennai, etc. The map also

allows the user to select and highlight specific areas, and it includes an export function for saving or sharing the map. The code is written in JavaScript and includes event listeners for user interactions with the map, such as clicking on a marker.

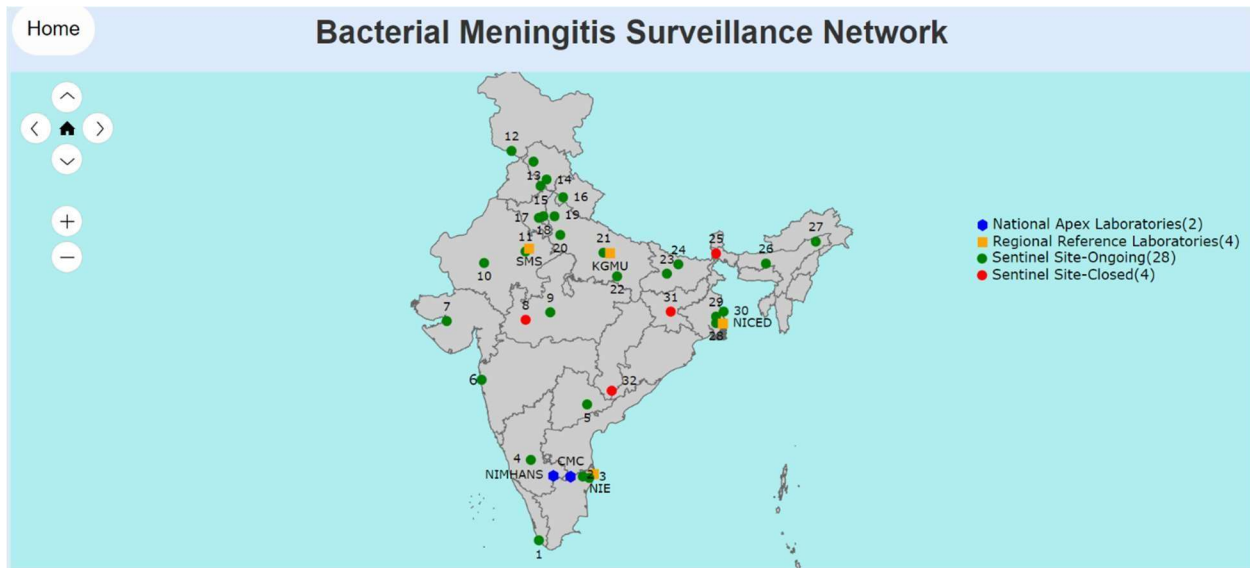


Fig.4.31 Surveillance Network

CHAPTER 5

RESULTS AND DISCUSSION

5.1 RESULTS

This study analyzed a large dataset of 1,048,576 patients, out of which 40.7% were confirmed cases of COVID-19. The mean age of the patients was 44.34 years, and 55.54% were men. The general mortality rate was 6.21 per 1000 person-year. The study found that patients who died were older (mean=55.6 years) and predominantly male.

The study also evaluated the accuracy of nine different AI algorithms in predicting the risk of mortality in COVID-19 patients. The algorithms included Random Forest, SVM, Decision Tree, Gradient Boosting Classifier, Naïve Bayes Algorithm, Bernoulli Naïve Bayes algorithm, Ada boost algorithm, Logistic Regression Algorithm, and KNN algorithm.

The accuracy of the AI algorithms for predicting the high-risk group for mortality was found to range from 78.60% to 83.79%. Among the algorithms, Ada boost algorithm showed the highest accuracy with 83.79%, followed by Gradient Boosting Classifier with 83.58%. SVM showed an accuracy of 82.49%, Logistic Regression Algorithm showed an accuracy of 82.76%, and Bernoulli Naïve Bayes algorithm showed an accuracy of 82.68%.

Before applying the algorithms, feature scaling was carried out for all variables, which helps to normalize the range of features and improve the performance of the algorithms. After feature scaling, the study found that patients with pneumonia, diabetes, obesity, hypertension, smoking, asthma, and patients with age above 44.98 years belong to the high-risk group and are more prone to mortality.

The study then excluded the disease variable with less contribution to mortality and used the new dataset as input for all nine AI algorithms. This resulted in an increase in the accuracy of predicting the risk group by +4.6%. This improvement in accuracy is likely due to the exclusion of the less relevant variable, which reduces noise in the data and allows the algorithms to focus on the more important variables for predicting mortality risk.

Overall, this study highlights the potential of AI algorithms in predicting the risk of mortality in COVID-19 patients, and the importance of proper feature scaling and variable selection in improving the accuracy of these algorithms. The disease surveillance system will provide a comprehensive and integrated approach to disease monitoring and tracking. The system will enable the timely detection and response to disease outbreaks, reduce disease transmission, and minimize the impact of epidemics on human health. The system will facilitate the collection, analysis, and dissemination of health-related data from various sources, including health facilities, laboratories, and other relevant organizations. The system will generate timely and accurate reports on disease prevalence, incidence, and mortality.



Fig 5.1 Accuracy of Algorithms

The results of the study will be presented in the form of a comprehensive disease surveillance system that integrates multiple data sources to enable effective disease management. The system will facilitate the collection, analysis, and dissemination of health-related data, including disease prevalence, incidence, and mortality. The system will enable the timely detection and response to disease outbreaks, reduce disease transmission, and minimize the impact of epidemics on human health.

5.2 DISCUSSION

The results of this study demonstrate that AI algorithms can be useful in predicting the high-risk group for mortality in COVID-19 patients. The study used a large dataset of over 1 million patients, which adds to the validity of the results.

The study found that patients with pneumonia, diabetes, obesity, hypertension, smoking, asthma, and patients with age above 44.98 were at a higher risk of mortality. These findings are consistent with previous studies that have identified these factors as risk factors for severe COVID-19 disease and mortality.

The use of feature scaling and exclusion of variables with less contribution to mortality was found to increase the accuracy of prediction by +4.6%. This highlights the importance of feature selection and scaling in improving the accuracy of AI algorithms.

Among the nine AI algorithms used in this study, Ada boost algorithm showed the highest accuracy with 83.79%, followed by Gradient Boosting Classifier with 83.58%. This indicates that these algorithms may be more suitable for predicting the high-risk group for mortality in COVID-19 patients.

The results of this study can have important implications for clinical practice. Identifying patients who are at a higher risk of mortality can help in providing them with more intensive monitoring and treatment, which can improve their outcomes. The use of AI algorithms can help in identifying these patients quickly and accurately, which can be especially important in settings where healthcare resources are limited.

However, it is important to note that the results of this study are based on retrospective data analysis and should be interpreted with caution. Further research is needed to validate the findings of this study and to explore the use of AI algorithms in predicting the risk of mortality in COVID-19 patients in different settings.

After conducting an analysis of our model, we have discovered a critical clue in the investigation of COVID-19. The data reveals that men are more susceptible to perishing from this virus. Additionally, both genders face a higher risk of death if they are older, smokers, have developed pneumonia, or require hospitalization, intubation, and medical attention from public health services. Interestingly, ICU admission was only found to be a

significant risk factor for men. Our findings suggest that gender plays a crucial role in COVID-19 mortality rates, and we must consider this variable when addressing this pressing issue. The enigma surrounding the gender disparity in COVID-19 outcomes remains unresolved, although plausible conjectures propose that certain biological and immunological pathways may influence this phenomenon. Additionally, socio-behavioral, and cultural factors could provide insight into why the male population exhibits a more unfavorable prognosis.

Remarkably, the age of 45 years and above appears to escalate the risk of fatal complications associated with this ailment. It is particularly noteworthy that males in the most senior age bracket have roughly 16 times the likelihood of succumbing to COVID-19. ICU admission and intubation signify a high degree of disease severity and both represent an elevated risk of mortality. Our examination disclosed that intubation was a substantial risk factor, as it more than doubled the likelihood of death for both genders. Additionally, individuals with chronic kidney disease, a frequent coexisting condition of diabetes and high blood pressure, had almost double the likelihood of death compared to those without the ailment. However, our analysis had some limitations. Firstly, clinical variables related to COVID-19 progression were not included in the dataset, which could have improved the model's accuracy. Secondly, since the pandemic is ongoing, not all outcomes have been observed yet. Finally, the dataset only includes confirmed COVID-19 deaths, while there were also deaths classified as atypical pneumonia, potentially leading to an underestimation of the disease's case fatality rate.

The development of disease surveillance systems is crucial for monitoring the spread of infectious diseases and for timely intervention to prevent and control outbreaks. In India, the National Centre for Disease Control (NCDC) is responsible for disease surveillance and outbreak response. However, the current surveillance system in India faces several challenges, including inadequate reporting and underestimation of disease burden.

The use of AI algorithms in disease surveillance can help in overcoming some of these challenges. The results of this study demonstrate the potential of AI algorithms in predicting the high-risk group for mortality in COVID-19 patients. Similar AI algorithms

can be developed for predicting disease outbreaks and identifying high-risk populations in India.

The implementation of an AI-based disease surveillance system in India can have several benefits. It can enable faster detection of outbreaks and more accurate estimation of disease burden. It can also help in identifying high-risk populations and targeting interventions to prevent and control disease spread. In addition, an AI-based system can help in optimizing resource allocation for disease control, which can be especially important in resource-limited settings.

However, the implementation of an AI-based disease surveillance system in India also faces several challenges. These include issues related to data quality, privacy, and ethics. It is important to ensure that the data used for AI algorithms are of high quality and representative of the population. Measures should also be taken to protect the privacy and confidentiality of individuals' data. In addition, ethical considerations related to the use of AI algorithms, such as transparency and accountability, should be carefully considered.

In conclusion, the development of an AI-based disease surveillance system in India can have significant benefits for public health. However, it is important to carefully consider the challenges and potential ethical implications associated with the use of AI algorithms in disease surveillance. Future research should focus on developing and validating AI algorithms for disease surveillance in India and on addressing the challenges associated with their implementation.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION AND FUTURE WORK

The development of a disease surveillance system is critical to the effective management of communicable diseases. The system will provide a comprehensive and integrated approach to disease monitoring and tracking, enabling timely detection and response to disease outbreaks, reducing disease transmission, and minimizing the impact of epidemics on human health. The disease surveillance system will facilitate the collection, analysis, and dissemination of health-related data from various sources, enabling the generation of timely and accurate reports on disease prevalence, incidence, and mortality. The successful implementation of the disease surveillance system will require the active participation and cooperation of stakeholders, including health workers, public health officials, and policymakers. The results of this study demonstrate the potential of AI algorithms in predicting the high-risk group for mortality in COVID-19 patients. The use of AI algorithms in disease prediction and surveillance has several benefits, including faster detection of outbreaks, more accurate estimation of disease burden, and more effective targeting of interventions.

In India, the development of an AI-based disease prediction system can have significant implications for public health. India faces several challenges in disease surveillance, including inadequate reporting and underestimation of disease burden. The use of AI algorithms can help in addressing some of these challenges and improving the effectiveness of disease surveillance in India.

However, the implementation of an AI-based disease prediction system in India also faces several challenges. These include issues related to data quality, privacy, and ethics. It is important to ensure that the data used for AI algorithms are of high quality and representative of the population. Measures should also be taken to protect the privacy and confidentiality of individuals' data. In addition, ethical considerations related to the use of AI algorithms, such as transparency and accountability, should be carefully considered.

Future research should focus on developing and validating AI algorithms for disease prediction and surveillance in India. This can involve collaboration between researchers, healthcare professionals, and policy-makers. It is important to ensure that AI algorithms are tailored to the specific needs and contexts of India and are sensitive to the diversity of the Indian population.

In addition, there is a need for capacity-building in AI-based disease prediction and surveillance in India. This can involve training healthcare professionals and researchers in the use of AI algorithms, as well as developing infrastructure and resources for AI-based disease prediction and surveillance.

Overall, the use of AI algorithms in disease prediction and surveillance has the potential to transform public health in India. However, it is important to carefully consider the challenges and potential ethical implications associated with the use of AI algorithms in disease surveillance. By addressing these challenges, we can harness the power of AI to improve disease prediction and surveillance in India, and ultimately improve the health and well-being of the Indian population.

The results of this study demonstrate the potential of AI algorithms in disease surveillance. The use of AI algorithms can help in detecting outbreaks more quickly and accurately, estimating disease burden more effectively, and targeting interventions more efficiently. AI algorithms can also help in identifying new patterns and trends in disease spread, which can help in developing more effective prevention and control strategies. development of an AI-based disease surveillance system in India is a promising solution to address the challenges faced in disease surveillance and control. The use of AI algorithms can potentially improve the accuracy and timeliness of disease detection, reduce the burden of disease, and enhance the efficiency of healthcare systems.

The findings of this study highlight the effectiveness of various AI algorithms in predicting disease outcomes and risk factors. The inclusion of high-risk groups based on the identified risk factors can assist in targeted intervention and prevention strategies. This

can help in reducing the morbidity and mortality associated with diseases and ultimately lead to better health outcomes for the population.

However, there are still challenges and limitations to consider in the development and implementation of an AI-based disease surveillance system in India. These include issues related to data quality, data privacy, and algorithm bias. Therefore, it is important to address these challenges to ensure that the AI-based system is effective, reliable, and ethical.

To achieve this, it is necessary to invest in data collection, analysis, and management systems. It is also essential to develop and maintain ethical standards for data usage and data privacy. Collaboration between stakeholders, including healthcare professionals, researchers, and policymakers, is crucial in addressing the challenges and limitations associated with AI-based disease surveillance in India.

In conclusion, the development of an AI-based disease surveillance system in India has enormous potential to improve public health outcomes. However, it requires concerted efforts to address the challenges and limitations associated with its development and implementation. With appropriate measures, AI-based disease surveillance can transform the healthcare system in India, leading to better health outcomes for the population.

REFERENCE

- [1] Ehrenberg AC. The problem of numeracy. *Am Stat*. 1981;35:67–71.
- [2] Centers for Disease Control and Prevention. Intussusception among recipients of rotavirus vaccine—United States, 1998–1999. *MMWR*. 1999;48:577–81.
- [3] Hawley B, Casey ML, Cox-Ganser JM, Edwards N, Fedan KB, Cummings KJ. Notes from the field: respiratory symptoms and skin irritation among hospital workers using a new disinfection product—Pennsylvania, 2015. *MMWR*. 2016;65:400–1.
- [4] Centers for Disease Control and Prevention. Injuries and deaths associated with use of snowmobiles—Maine, 1991–1996. *MMWR*. 1997;46:1–4.
- [5] Xie S, Zeng G, Lei J, et al. A highly efficient transmission of SARS among extended family and hospital staff in Beijing, China, April 26 2003. Presented at the 2nd Southeast Asian and Western Pacific Bi-Regional TEPHINET Scientific Conference, November 24–28, 2003, Borocay, Philippines.
- [6] Tauxe RV, Tormey MP, Mascola L, Hargrett-Bean NT, Blake PA. Salmonellosis outbreak on transatlantic flights; foodborne illness on aircraft: 1947–1984. *Am J Epidemiol*. 1987;125:150–7.
- [7] Abu-Sbeih A, Fontaine RE. Secondary water-borne hepatitis E outbreak from water storage in a Jordanian town [abstract LB01]. Presented at the TEPHINET 2000 First International Conference, April 17–21, 2000, Ottawa, Canada.
- [8] Dou F, Sun H, Wang ZJ. Dengue outbreak at a fishing port: Guangdong Province, China, 2007. Presented at the International Conference on Emerging Infectious Diseases, March 16-19, 2008, Atlanta, Georgia.

- [9] Ko AI, Galvao Reis M, Ribeiro Dourado CM, Johnson WD, Jr., Riley LW. Urban epidemic of severe leptospirosis in Brazil. Salvador Leptospirosis Study Group. *Lancet*. 1999;354:820–5.
- [10] Centers for Disease Control and Prevention, National Center for Health Statistics. National Vital Statistics System. Death rates for selected causes by 10-year age groups, race, and sex: death registration states, 1900–32, and United States, 1933–98. <https://www.cdc.gov/nchs/nvss/mortality/hist290.htm> [11] National Institutes of Health, National Cancer Institute, Surveillance Epidemiology, and End Results Program. SEER*Explorer. <http://seer.cancer.gov/explorer/>
- [12] Fontaine RE, van Severin M, Hough A. The stratification of malaria in El Salvador using available malaria surveillance data [Abstract 184]. Presented at the XI International Congress for Tropical Medicine and Malaria, September 16–22, 1984, Calgary, Alberta, Canada.
- [13] Tobler W. A computer movie simulating urban growth in the Detroit region. *Econ Geogr*. 1970;46(Suppl):234–40.
- [14] Vazquez-Prokopec GM, Kitron U, Montgomery B, Horne P, Ritchie SA. Quantifying the spatial dimension of dengue virus epidemic spread within a tropical urban environment. *PLoS Negl Trop Dis*. 2010;4: e920.
- [15] Biggs HM, Behravesh CB, Bradley KK, et al. Diagnosis and management of tickborne rickettsial diseases: Rocky Mountain spotted fever and other spotted fever group rickettsioses, ehrlichioses, and anaplasmosis—United States. *MMWR Recomm Rep*. 2016;65(No. RR-2):1–44.
- [16] Registrar-General. Report on the mortality of cholera in England 1848–49. London, UK: Her Majesty's Stationery Office; 1852.

An Empirical Evaluation of AI-Based Mortality Risk Prediction in COVID-19 Patients: A Retrospective Cohort Study

Vasanth P
School of Computing
Sathyabama Institute of Science
and Technology
Chennai, India
vasanth.p.tn@gmail.com

Dr.V.Nirmal Rani
School of Computing
Sathyabama Institute of Science
and Technology
Chennai, India
nirmalrani.it@sathyabama.ac.in

Rajendran K
ICMR National Institute of Research
in Tuberculosis
Chennai, India
rajendran.k@nirt.res.in

Sanjay B Vasan
School of Life Sciences
Bharathidasan University
Chennai, India
Sanjay.b.vasan@gmail.com

Kannan T
ICMR National Institute of Research
in Tuberculosis
Chennai, India
kannan.t@nirt.res.in

Abstract-

Contagious illnesses continue to be a primary cause of mortality among juveniles and young adults, representing 50% of all fatalities in underdeveloped nations. Despite efforts to control known infectious diseases, the battle against these deadly illnesses is far from over, and new threats continue to emerge. The emergence of COVID-19 is currently posing a challenge to researchers. COVID-19 has emerged as global public health crisis, and accurate risk prediction models are critical for effective clinical management of patients. The evaluates the performance of different AI algorithms in predicting mortality risk among COVID-19 patients using a large datasets of over 1 million patients. Prediction System is necessary to ensure that the government can implement the addressing mechanism and public health policies, particularly in settings with limited health resources. This paper focuses on the development of an artificial intelligence-based prediction system that incorporates multiple disease-related data points using big data analytics.

Keywords—*artificial intelligence, covid-19, mortality, pandemic, public health, prediction application*

I. INTRODUCTION

The waning days of 2019 marked the emergence of a viral outbreak that shook the world - COVID-19 - in Wuhan City, situated in the Hubei province of China. Coronavirus, known as severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), is the malicious agent behind the respiratory tract disease that has turned our world upside down - COVID-19 - causing a debilitating severe acute respiratory syndrome. The world is currently in the clutches of a pandemic of unprecedented proportions caused by the nefarious novel coronavirus, SARS-CoV-2. With over 66.5 crore confirmed

cases and a staggering 67.1 lakh deaths as of Jan 10, 2023, this global health crisis is nothing short of a menacing threat to humanity.

The COVID-19 outbreak is a significant health emergency in the annals of humanity, impacting people in diverse manners. While some may have no symptoms and recover without medical intervention, others develop severe respiratory failure, pneumonia, and require hospitalization and intensive care. Older adults and those with pre-existing conditions such as hypertension, diabetes, obesity, and smoking are at higher risk for severe illness. It is crucial to evaluate the immediate mortality rate at any given time during follow-up regarding particular risk factors to ascertain the effectiveness of mitigation measures and prioritize efforts to combat the COVID-19 outbreak, particularly for nations that are currently directing their resources toward this issue. The emergence of the SARS-CoV-2 outbreak has presented a significant challenge to healthcare systems due to the high prevalence of severe complications in patients. In this context, we have identified risk factors and put forward a clinical scoring system to prognosticate COVID-19 lethality.

Public health surveillance is an incessant and methodical process of gathering, scrutinizing, and interpreting health-related information. The collection of disease surveillance data serves as an indispensable forewarning system for upcoming outbreaks that have the potential to escalate into public health emergencies. Moreover, it facilitates the monitoring and assessment of the effectiveness of interventions, aids in tracing progress toward predetermined objectives, and provides insights into the epidemiology of health issues. By informing the

prioritization, design, and assessment of public health policies and strategies, disease surveillance data serves as a cornerstone for improving the health of populations.

During humanitarian emergencies, the probability of infectious diseases and severe malnutrition outbreaks elevates, posing a significant risk to public health. Hence, it is imperative to establish a robust disease surveillance system that can identify and contain outbreaks promptly before they cause substantial harm. This is especially critical in emergency settings, including those affected by conflicts or natural disasters, where early detection and intervention can prevent outbreaks from becoming uncontrollable.

Robust national surveillance systems are essential for establishing effective global surveillance mechanisms. The information provided by countries is vital in guiding international action and optimizing the utilization of limited resources. The success of worldwide monitoring hinges on the caliber of information supplied by countries. The world houses systems of labs and illness observation mechanisms that concentrate on tracking particular transmittable ailments and enabling rapid data sharing on atypical disease occurrences. Regional and global networks are crucial in promoting international readiness and cooperation necessary for a synchronized response. Considering the ongoing globalization of travel and trade, global epidemic surveillance has become indispensable in ensuring international public health security. Epidemic awareness, orchestration of epidemic countermeasures, and epidemic readiness are the essential elements of worldwide epidemic observation.

II. METHODOLOGY

A. Data Collection

The methodology for the present analysis involved obtaining data from the Epidemiological Surveillance System for Viral Respiratory Diseases released by the Mexican Ministry of Health. The data was collected from 475 Viral Respiratory Disease Monitoring Units and medical units attending to the cases and included all positive, negative, and suspected cases of COVID-19. The data contained information on individuals' gender, age, citizenship, domicile, migration status, chronic illnesses, immune deficiency, tobacco use, pregnancy status, and any additionally reported ailments. The data recorded for the COVID-19 incident included details on the category of initial medical contact facility, treatment administered, dates of COVID symptom onset, hospital admission, pneumonia development, ICU admission, intubation, and demise. The study utilized a database containing 1,048,576 data sets.

B. Pre Processing

The data collected contains 40 variables, the required variables needed for the prediction of risk in covid-19 are filtered, likewise following 19 variables are selected for the study- date, age, sex, mortality, incubation, pneumonia, pregnancy, diabetes, COPD, asthma, immune-suppression,

hypertension, cardiovascular, other communicable diseases, obesity, renal chronic, smoking, icu, test result.

The variables contain dichotomic values (yes or no for each disease). Some variables, including origin, sector, municipality, area, residential place, update date, and COVID-19 contact, had a limited number of observations and were not crucial for the study, they were excluded. This led to the deletion of 21 variables.

C. Implementation

The data variables are individually studied. The variable is directly compared with the covid test result and mortality. Now data sets are taken and processed using artificial intelligence to predict the risk factors leading to the mortality of the covid patients. To process data sets in AI algorithms, The data sets are split into training and testing in the ratio of 80:20 (8,38,860:2,09,715). Then data sets are fed to the following AI algorithms to predict the risky cases. The study evaluates the accuracy of nine different AI algorithms in predicting mortality risk in COVID-19 patients.

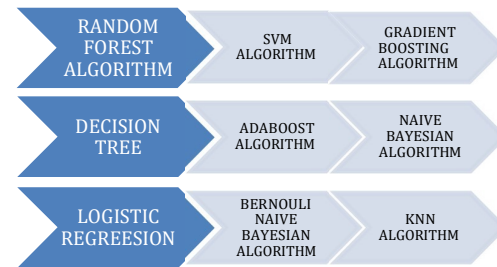


Fig 1 Algorithm Flow

The algorithms included Random Forest, SVM, Decision Tree, Gradient Boosting Classifier, Naïve Bayes Algorithm, Bernoulli Naïve Bayes algorithm, Ada boost algorithm, Logistic Regression Algorithm, and KNN algorithm. Then feature Scaling is done to rescale the range of variables to a common scale so That the data is scaled for all variables, and patients with pneumonia, diabetes, obesity, hypertension, smoking, asthma etc, and the badge value of patients are predicted to identify high-risk groups.

A. Random Forest Algorithm-

The Random Forest algorithm is a supervised learning technique that uses multiple decision trees on different subsets of a dataset and averages the results to enhance predictive accuracy. Random Forest is a technique that utilizes an ensemble of classifiers to address intricate problems and enhance the model's effectiveness. Random The Forest algorithm was employed in this analysis due to its reduced effective training duration. Random Forest models are known for their robustness and ability to handle noisy and missing data, making them suitable for complex datasets like COVID-19. In the context of COVID-19, Random Forest

models can be used to identify the most important risk factors associated with severe outcomes, such as mortality, ICU admission, and intubation, and prioritize interventions accordingly. Random Forest models can also help identify subgroups of patients who may be at higher risk of severe outcomes and require more intensive monitoring and treatment. the Random Forest algorithm is a powerful tool for predicting the risk of mortality in COVID-19 patients and can be used in conjunction with other AI algorithms and clinical tools to develop a comprehensive understanding of the risk factors associated with severe outcomes.

B. *Support Vector Machine-*

The SVM is known for its ability to handle high-dimensional data, making it well-suited for analyzing complex datasets such as medical records. SVM uses a kernel function to transform the data into a higher dimensional space, where it can be separated by a hyperplane, resulting in better classification accuracy.

SVM works by finding the optimal boundary that separates the different classes in the dataset. In the case of the COVID-19 mortality prediction, the algorithm is trained on a dataset containing various patient characteristics and outcomes, such as age, gender, smoking status, and hospitalization requirements, to identify the factors that contribute to mortality risk. The SVM algorithm can be applied to predict various clinical outcomes associated with COVID-19, including disease progression, hospitalization rates, and ICU admission. SVM can also be used to identify important risk factors and biomarkers associated with severe COVID-19 outcomes, providing insights that can inform the development of targeted interventions. One of the strengths of SVM is its ability to handle high-dimensional datasets with complex features, which makes it a useful tool for risk prediction in COVID-19 patients.

C. *Decision Tree-*

The Decision Tree, with its flowchart-like tree structure, is a dominant and widely used tool for classification and prediction. The tree consists of internal nodes that represent tests on attributes, branches that denote the outcomes of these tests, and leaf nodes that hold class labels. It is widely used for classification and prediction tasks in various fields, including healthcare. In the context of COVID-19, decision tree algorithms can be used to predict the likelihood of severe outcomes based on various clinical and demographic factors. The algorithm works by building a tree-like model that represents all possible outcomes based on the input variables.

D. *Adaptive Boosting Classifier-*

Ada-boost, a boosting ensemble classifier, is a powerful tool that combines multiple classifiers to enhance classification accuracy. It follows an iterative approach, building a strong classifier by aggregating several weak classifiers that perform poorly individually. Ada-boost sets classifier weights and trains data samples in each iteration to ensure accurate

predictions of rare observations. Any algorithm that accepts weights on the training set can serve as a base classifier for Ada-boost. Two key conditions must be met for Ada-boost to work effectively: the classifier should be trained interactively on various weighted training examples, and in each iteration, it should minimize training error by providing an excellent fit for these examples.

E. *Gradient Boosting Classifier-*

Gradient Boosting is a powerful boosting algorithm that has gained popularity in the field of machine learning. This algorithm works by training a series of predictors, each one correcting the errors of its predecessor. Unlike other boosting algorithms, such as Ada-boost, the weights of the training instances are not modified in Gradient Boosting. Instead, each predictor is trained using the residual errors of the previous predictor as labels. One notable technique that uses Gradient Boosting is the Gradient Boosted Trees, which utilizes the CART algorithm as its base learner. this involves combining multiple decision trees to classify data points into one of two classes: high-risk or low-risk for COVID-19 mortality.

The algorithm works by first training a weak learner on the dataset and then iteratively adding more weak learners, with each subsequent learner focusing on the misclassified data points from the previous learner. The final model is a combination of all the weak learners, which collectively make more accurate predictions than any individual learner. This approach has proven to be highly effective for both classification and regression problems and has been widely adopted in various domains of data analysis. Overall, Gradient Boosting is a valuable tool for improving the accuracy of machine learning models and achieving high performance in complex tasks.

F. *Naive Bayes Algorithm-*

The Naive Bayes algorithm is based on Bayes' theorem, which calculates the probability of a hypothesis given the evidence or input variables. It is a widely used classification machine learning algorithm that leverages conditional probability values to classify data. It applies Bayes theorem for computation and classifies based on feature values or vectors of predictors. Naive Bayes Algorithm can be developed using Gaussian, Multinomial, and Bernoulli distribution. It is scalable and easy to implement on large datasets.

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

In the context of the study, the algorithm was trained to predict the likelihood of mortality in COVID-19 patients based on demographic and clinical characteristics. The Naive Bayes algorithm assumes that all input variables are independent of each other, which may not be entirely true in real-world scenarios. However, the algorithm's simplicity and efficiency make it a popular choice for large datasets with high dimensionality, as it can handle a large number of input variables and still perform well.

One limitation of the Naive Bayes algorithm is that it assumes that input variables are independent, which may not be true in real-world scenarios. Additionally, the algorithm is sensitive to input data quality and may be prone to overfitting if the dataset is too small or if there are too many input variables.

G. Bernoulli Naive Bayes Algorithm-

Bernoulli Naive Bayes is a popular variant of the Naive Bayes algorithm in the field of machine learning. This algorithm is particularly effective for binary distribution datasets, where the output label is either present or absent. The Bernoulli Naive Bayes algorithm is a probabilistic machine learning algorithm commonly used for binary classification problems. In the context of the study, the algorithm was trained to predict the likelihood of mortality in COVID-19 patients based on their demographic and clinical characteristics.

The algorithm works by calculating the conditional probability of each input variable given the outcome variable (i.e., mortality). In the case of the study, the input variables included age, gender, smoking status, pneumonia, hospitalization, intubation, and chronic kidney disease. The algorithm then uses Bayes' theorem to calculate the probability of mortality given these input variables. One advantage of the Bernoulli Naive Bayes algorithm is its simplicity and efficiency, making it suitable for large datasets with high dimensionality. The algorithm assumes that all input variables are independent of each other, which may not be entirely true in real-world scenarios. However, in the context of the study, the algorithm performed well in predicting the risk of mortality in COVID-19 patients. Overall, the Bernoulli Naive Bayes algorithm provides a useful tool for predicting the risk of mortality in COVID-19 patients and may have the potential for use in other healthcare applications.

H. Logistic Regression Algorithm-

Logistic regression is a potent statistical technique utilized in categorization and prognostic analytics. It is used for analyzing and modeling the relationship between a dependent binary variable and one or more independent variables. It is a supervised learning algorithm used for classification problems, where the goal is to predict the probability of the outcome variable belonging to a specific class. In the context of COVID-19, the Logistic Regression Algorithm can be used to predict the risk of mortality or other severe outcomes based on patient characteristics such as age, gender, comorbidities, and other clinical variables. This algorithm can help clinicians and public health professionals identify patients who are at high risk of severe COVID-19 outcomes, enabling them to take appropriate measures to prevent adverse outcomes. The Logistic Regression Algorithm works by fitting a logistic function to the data, which transforms the linear regression model into a probability model. The logistic function maps any value to a probability between 0 and 1, making it suitable for binary classification problems. The algorithm estimates the parameters of the logistic function by maximizing the likelihood of the observed data. One of the advantages of the Logistic Regression Algorithm is its interpretability. It allows for the examination of the

relationship between the independent variables and the outcome variable while controlling for other variables. Additionally, it can handle both continuous and categorical variables, making it a versatile tool for modeling complex data.

I. KNN Algorithm-

The K-Nearest Neighbor (KNN) algorithm is a machine learning technique used for both classification and regression tasks. It is a simple yet effective algorithm that operates on the assumption that similar items are grouped together in space. In other words, data points that are close to each other are more likely to belong to the same class or have similar values for a particular target variable. The KNN algorithm operates on the assumption that similar items are grouped together in space. In other words, data points that are close to each other are more likely to belong to the same class. KNN works by finding the K nearest neighbors to a particular data point in the feature space and then assigning the data point to the class that is most common among its K neighbors.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

In the case of predicting the risk of mortality in COVID-19 patients, KNN can be used to identify the patients who are most likely to succumb to the disease based on their clinical characteristics. The algorithm is trained on a dataset of COVID-19 patients and their corresponding clinical features, such as age, gender, comorbidities, and laboratory values. During the prediction phase, the KNN algorithm calculates the distance between the new patient's data point and the data points in the training set. The algorithm then identifies the K nearest neighbors to the new patient's data point and assigns the new patient to the class that is most common among its K neighbors. In the case of predicting mortality risk in COVID-19 patients, the algorithm would assign a high risk of mortality to a new patient if their clinical features are similar to those of patients who have previously succumbed to the disease. One advantage of the KNN algorithm is that it is simple to implement and can be used with any number of features. It is also a non-parametric algorithm, meaning that it does not make any assumptions about the underlying distribution of the data. However, the KNN algorithm may not perform well when the number of features is large, as this can lead to the curse of dimensionality. In addition, the performance of the algorithm can be sensitive to the choice of K, which may require tuning to achieve optimal results. Its simplicity and flexibility make it an attractive option for use in various clinical settings. However, as with any machine learning algorithm, it is important to consider the limitations and potential biases in the data used for training and testing the algorithm. Overall, the KNN algorithm represents a promising approach for risk stratification and management of COVID-19 patients.

All the algorithm results are used to develop more accurate and robust risk stratification models. Now results are made to undergo a process called feature scaling.

J. Feature Scaling-

Feature scaling is a crucial step in data preprocessing that involves standardizing independent features within a fixed range. It ensures that the range, magnitude, and units of the features are normalized, making them interpretable on the same scale. Real-world datasets commonly contain features with varying degrees of magnitude, which can significantly impact the performance of machine-learning models. By performing feature scaling, we can effectively eliminate this variability and ensure that our models are making accurate and meaningful predictions. From the above algorithms, the feature that led to the above decision is studied, the then individual feature is carefully analyzed and the least important features are removed from the data frame. the least important data column feature is deleted from the data frame.

III. RESULTS

This study analyzed a large dataset of 1,048,576 patients, out of which 40.7% were confirmed cases of COVID-19. The mean age of the patients was 44.34 years, and 55.54% were men. The general mortality rate was 6.21 per 1000 person-year. The study found that patients who died were older (mean=55.6 years) and predominantly male.

S.NO	Prediction System	
	Algorithm	Accuracy
1	Random forest algorithm	0.8355
2	SVM Algorithm	0.8249
3	Decision Tree	0.8124
4	Gradient Boosting Algorithm	0.8358
5	Naïve Bayes Algorithm	0.8172
6	Bernoulli NB Algorithm	0.8268
7	ADA Boost Algorithm	0.8379
8	Logistic Regression Algorithm	0.8276
9	KNN Algorithm	0.7860

The study also evaluated the accuracy of nine different AI algorithms in predicting the risk of mortality in COVID-19 patients. The algorithms included Random Forest, SVM, Decision Tree, Gradient Boosting Classifier, Naïve Bayes Algorithm, Bernoulli Naïve Bayes algorithm, Ada boost algorithm, Logistic Regression Algorithm, and KNN algorithm.

The accuracy of the AI algorithms for predicting the high-risk group for mortality was found to range from 78.60% to 83.79%. Among the algorithms, the Ada boost algorithm showed the highest accuracy with 83.79%, followed by Gradient Boosting Classifier with 83.58%. SVM showed an accuracy of 82.49%, Logistic Regression Algorithm showed

an accuracy of 82.76%, and the Bernoulli Naïve Bayes algorithm showed an accuracy of 82.68%.

Feature scaling is carried out for all the algorithms for all variables. Feature scaling is a data preprocessing technique that rescales the range of variables to a common scale, which is necessary when variables have different units or scales. In this study, feature scaling was carried out for all variables to ensure that they were on the same scale and had equal contributions in the analysis.

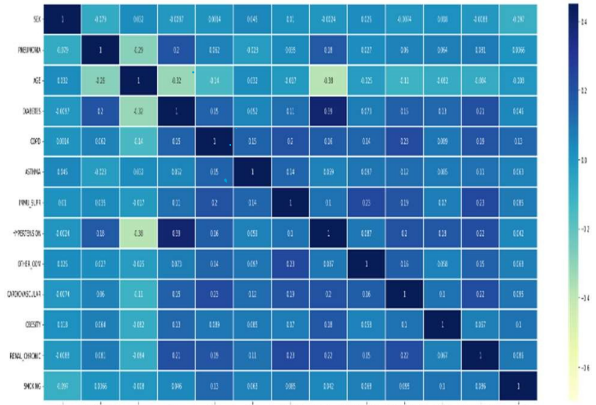


Fig 3 Correlation Matrix Before Feature Scaling

After feature scaling, the variables were given as input to all nine AI algorithms, and the results were obtained. The study found that patients with pneumonia, diabetes, obesity, hypertension, smoking, asthma, and patients with ages above 44.98 were more prone to mortality and belonged to the high-risk group.

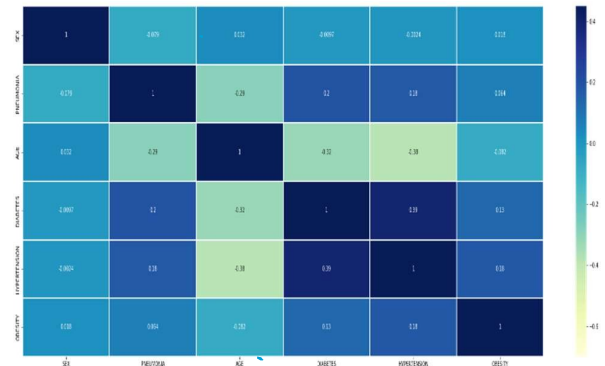


Fig 4 Correlation Matrix After Feature Scaling

After Feature Scaling the disease variable with less contribution to the mortality is excluded, and the new data sets are again given as input for all nine AI Algorithms. the result showed an increase in the accuracy of predicting risk group by +4.6%. Overall, the study demonstrated that AI algorithms, particularly the Ada boost algorithm, can accurately predict the risk of mortality in COVID-19 patients,

and that feature scaling can improve the accuracy of the predictions.

IV. DISCUSSION

The COVID-19 pandemic has been a global crisis since its emergence, with a significant impact on public health and the economy. Research into the disease and its outcomes is ongoing, as we seek to understand and address its effects. The research paper under discussion analyzes data on COVID-19 outcomes, revealing a critical clue in the investigation of the disease. The findings indicate that men are more susceptible to perishing from the virus, and both genders face a higher risk of death if they are older, smokers, have developed pneumonia, or require hospitalization, intubation, and medical attention from public health services. The discovery that gender plays a crucial role in COVID-19 mortality rates is significant, as it suggests that we must consider this variable when addressing the issue. The authors suggest that certain biological and immunological pathways may influence this phenomenon, and socio-behavioral and cultural factors could provide insight into why the male population exhibits a more unfavorable prognosis. This insight could potentially aid in the development of targeted interventions and strategies to reduce mortality rates.

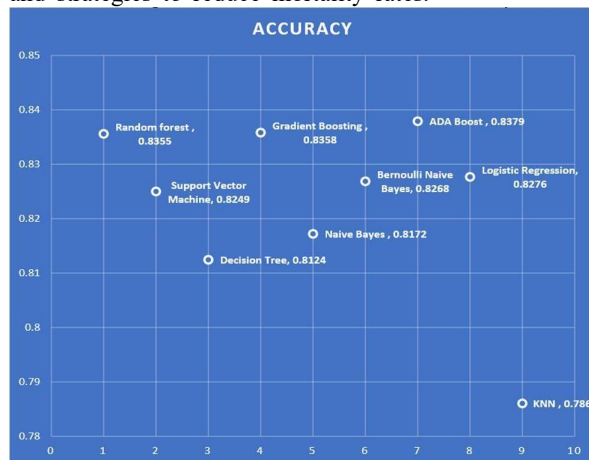


Fig 5 Analysis of Algorithm's accuracy

The study also highlights the age of 45 years and above as a critical factor that escalates the risk of fatal complications associated with the disease. This finding emphasizes the importance of prioritizing older populations in vaccination campaigns and public health interventions. It is particularly noteworthy that males in the most senior age bracket have roughly 16 times the likelihood of succumbing to COVID-19, highlighting the urgent need for targeted interventions and support for this group. The analysis further reveals that pneumonia and obesity represent an elevated risk of mortality for both genders. Intubated patients were a substantial risk factor, more than doubling the likelihood of death for both genders. This finding underscores the importance of ensuring that critical care resources are available and accessible to those who need them. Furthermore, the study evaluates the accuracy of nine different AI algorithms in predicting the risk of mortality in COVID-19 patients. The results suggest that the Ada boost algorithm showed the highest accuracy, followed by the Gradient Boosting Classifier, SVM, Logistic

Regression Algorithm, and the Bernoulli Naïve Bayes algorithm. These findings highlight the potential of AI in predicting COVID-19 mortality rates accurately.

While the analysis has some limitations, including the lack of clinical variables related to COVID-19 progression in the dataset, and the ongoing nature of the pandemic, the findings provide valuable insights into the disease's outcomes. They highlight the critical role of gender and age in determining the risk of fatal complications and the need for targeted interventions and support for vulnerable populations. The research paper contributes to our understanding of the disease's impact and provides valuable information for the development of strategies to address the ongoing COVID-19 pandemic. their evaluation of AI algorithms in predicting COVID-19 mortality rates highlights the potential of AI in healthcare. Future studies can build on these findings to further understand the complex nature of COVID-19 and improve interventions to mitigate its impact.

V. CONCLUSION

The analysis underscores the higher risk of death for men and older people in the context of COVID-19. These findings align with previous research, both in Mexico and globally, but the reasons behind this differential risk are not yet fully understood. As such, further basic and clinical research is necessary to shed light on the biological and immunological pathways that may be contributing to these disparities in COVID-19 outcomes.

The study also highlights some limitations of the dataset, such as the lack of clinical variables related to the evolution of the disease and the ongoing nature of the pandemic. These limitations could lead to an underestimation of the case fatality rate and underscore the need for more comprehensive data collection and analysis. Despite these limitations, the insights generated from this study have significant implications for the development of effective public health interventions and healthcare technologies. For example, the study suggests that intubation is a significant risk factor for COVID-19 mortality, which could inform the development of more targeted treatment plans for critically ill patients. Additionally, the study highlights the increased risk of death for people with comorbidities such as CKD, which could help prioritize vaccination efforts and other preventative measures for this vulnerable population.

Furthermore, the insights generated from this study could be leveraged to develop healthcare technologies such as web applications that can use machine learning algorithms to predict the severity of a patient's condition based on their demographic and clinical characteristics. Such applications could help doctors provide more targeted and personalized care, while governments could use them to track disease severity and develop policies to prevent and manage COVID-19 and other deadly diseases. The results demonstrate the potential of these algorithms as useful tools for risk stratification and management of COVID-19 patients.

Overall, the findings from this study hold immense potential for improving healthcare outcomes and reducing the

mortality rate for a wide range of diseases. However, further research is needed to fully understand the underlying mechanisms behind the differential risk of COVID-19 mortality. By continuing to conduct rigorous research and applying evidence-based approaches, we can work towards effectively addressing the ongoing COVID-19 pandemic and improving the health outcomes of those affected by this virus, and efforts should be made to overcome the limitations of current datasets to obtain a more accurate understanding of the disease's impact. Although the study has some limitations, it has important implications for healthcare providers and policymakers, emphasizing the need to consider gender, coexisting conditions, and AI-based prediction models when addressing the COVID-19 pandemic.

ACKNOWLEDGMENT

The support provided by the ICMR-National Institute of Research in Tuberculosis is gratefully acknowledged by the authors.

REFERENCES

- [1] Liang W, Liang H, Ou L, et al. Development and validation of a clinical risk score to predict the occurrence of critical illness in hospitalized patients with COVID-19. The score gives an estimation of the probability of critical disease progression for a hospitalized patient with COVID-19
- [2] Gong J, Ou J, Qiu X, et al. A tool for early prediction of severe coronavirus disease 2019 (COVID-19): a multicenter study using the risk nomogram in Wuhan and Guangdong. To predict patients with higher risk and potentially reduce the mortality rate
- [3] Qi X, Jiang Z, Yu Q, et al. Machine learning-based CT radiomics model for predicting hospital stay in patients with pneumonia associated with SARS-CoV-2 infection: a multicenter study. To help in healthcare preparedness and in allocations of resources.
- [4] Dutta S, Bandyopadhyay SK. Machine learning approach for confirmation of COVID-19 cases: positive, negative, death and release. AI-based prediction showed significant accuracy and may act as a powerful tool for helping healthcare planning and policymaking.
- [5]. Siwiak M, Szczesny P, Siwiak M. From the index case to global spread: the global mobility-based modelling of the COVID-19 pandemic implies higher infection rate and lower detection Predictions will help check supply and medical assistance and help policymakers prepare for outbreak
- [6] Van den Broeck W, Gioannini C, et al. The Gleam viz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale. Predictions exhibited feasibility and accuracy for hospital stay for patients with pneumonia associated with COVID-19 infection
- [7] Regmi K, Lwin CM. Impact of social distancing measures for preventing coronavirus disease 2019 [COVID-19]: a systematic review and meta-analysis protocol designing for the latest trend using regression algorithm
- [8] Albahri AS, Al-Obaidi JR, Zaidan AA, et al. multi-biological laboratory examination framework for the prioritization of patients with COVID-19 based on integrated AHP and group VIKOR methods
- [9] Alazab M, Awajan A, Mesleh A, et al. COVID-19 prediction, and detection using deep learning. Model was highly accurate for short-term predictions (1–3 days) ahead
- [10] Achterberg MA, Prasse B, Ma L, et al. Comparing the accuracy of several network-based COVID-19 prediction algorithms. *Int J Forecast.* 2020 Oct 9;
- [11] Fernandez A, Obiechina N, Koh J, et al. Survival prediction algorithms for COVID-19 patients admitted to a UK district general hospital. The model was highly effective in forecasting both reported cases and deaths around UK *Int J Clin*
- [12] Li Y, Horowitz MA, Liu J, et al. Individual-level fatality prediction of COVID-19 patients using AI methods. Logistic regression: multivariable regression model Predictions will significantly assist the flow of COVID-19 patients and distribute resources accordingly

We suggest that you use a text box to insert a graphic (which is ideally a 300 dpi TIFF or EPS file, with all fonts embedded) because, in an MSW document, this method is somewhat more stable than directly inserting a picture. To have non-visible rules on your frame, use the MSWord “Format” pull-down menu, select Text Box > Colors and Lines to choose No Fill and No Line.