A Project Report
on
A Survey on Predictive Healthcare Analytics Based on AI/ML

Submitted in partial fulfilment of the requirements
for the award of the degree of

Bachelor of Technology
in
Information Technology
by

Udit Maurya
Roll No.: 2100970130117

Ritesh Kumar
Roll No.: 2100970130093

Vishal Yadav
Roll No.: 2100970130127

Group no.: 24IT732

Under the Supervision of

Dr. Javed Miya

Galgotias College of Engineering & Technology
Greater Noida 201306
Uttar Pradesh, INDIA

Affiliated to

Dr. A.P.J. Abdul Kalam Technical University

Lucknow
May 2025

GALGOTIAS COLLEGE OF ENGINEERING & TECHNOLOGY
GREATER NOIDA - 201306, UTTAR PRADESH, INDIA.

DECLARATION

We hereby that the project work presented in this project report entitled “**A Survey on Predictive Healthcare Analytics Based on AI/ML Approaches”** in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Information Technology, submitted to A.P.J Kalam Technical University, Lucknow is based on my work carried out at Department of Information Technology, Galgotias College of Engineering and Technology, Greater Noida. The work carried in the report is original and project work reported in this report has not been submitted by us for the award of any other degree or diploma.

Signature:
Name: Udit Maurya
Roll No: 2100970130117

Signature:
Name: Ritesh Kumar
Roll No: 2100970130093

Signature:
Name: Vishal Yadav
Roll No: 2100970130127

Date:
Place: Greater Noida

GALGOTIAS COLLEGE OF ENGINEERING & TECHNOLOGY
GREATER NOIDA - 201306, UTTAR PRADESH, INDIA.

CERTIFICATE

This is to certify that the project report entitled "**A SURVEY ON PREDUCTIVE HEALTHCARE ANALYTICS BASED ON AI/ML APPROACHES"** submitted by **UDIT MAURYA (2000970130117), RITESH KUMAR (2100970130093), VISHAL YADAV (2100970130127)** to the Galgotias College of Engineering & Technology, Uttar Pradesh in Partial fulfillment for the award of Degree of Bachelor of Technology in Information Technology is a bonafide record of the project work carried out by them under my supervision during the year 2024 – 2025.

Dr. Javed Miya	**Dr. Sanjeev Kumar Singh**
(Professor)	**(HOD IT)**
Deptt. of IT	

ACKNOWLEDGEMENT

First of all, we would like to express our gratitude to our beloved **Chairman, Shri. Suneel Galgotia and CEO, Shri. Dhruv Galgotia** for providing necessary facilities to carry out and finish the project successfully. We are grateful to our **Director, Dr. Vikram Bali** for his support and encouragement.

Our special thank goes to **Dr. Sanjeev Kumar Singh, Head of the Department, Information and Technology,** for providing an environment that encouraged us in working towards our goal and supported in completion of our project. This project gave us an opportunity to apply creative and critical thinking skills.

We would like to present our deep sense of gratitude to our project guide, **Dr. Javed, Professor, Department of Information and Technology** who has always been a source of motivation and firm support for carrying out the project. The supervision and support that she gave truly helped in the progression of the project. Her cooperation is much indeed appreciated. We are highly obliged to his for his valuable advices and moral support during researchperiod.

Finally, our greatest and special gratitude goes to our family for their love and support.

Udit Maurya

(2100970130117)

Ritesh Kumar

(2100970130093)

Vishal Yadav

(2100970130127)

ABSTRACT

The integration of Artificial Intelligence (AI) within predictive healthcare analytics heralds a promising era, poised to revolutionize patient care, clinical decision-making, and the allocation of healthcare resources. By meticulously analysing vast repositories of healthcare data, AI algorithms showcase an unparalleled ability to extract invaluable insights and predictions. This potential extends beyond merely improving patient outcomes to optimizing cost-effectiveness and refining interventions aimed at bolstering public health. Nonetheless, the successful incorporation of AI in healthcare analytics necessitates a vigilant approach to various challenges and ethical considerations. These encompass safeguarding patient privacy, ensuring fairness and equity in AI-driven healthcare decisions, and nurturing trust among stakeholders. To unlock the full potential of intelligence, it is important to address these ethical issues while maintaining ethical standards and social values. This research aims to conduct a comprehensive exploration of the multifaceted impact of AI integration in predictive healthcare analytics. Through an exhaustive examination of its implications, this study seeks not only to elucidate the transformative potential of AI in healthcare but also to contribute significantly to the collective understanding and future advancements of this pioneering technology within the healthcare sector.

Keywords: *Predictive Healthcare, Autonomic Computing, Artificial Intelligence, Machine Learning*

TABLE OF CONTENT

List of Tables

List of Figure

List of Abbreviations

AI	Artificial Intelligence
ML	Machine Learning
SVM	Support Vector Machine
KDE	Kernel Density Estimate
NBC	Naïve Bayes Classifier
RFC	Random Forest Classifier

CHAPTER 1

INTRODUCTION

1.1 HEALTHCARE PREDICTIVE ANALYTICS

Artificial Intelligence (AI) stands at the forefront of technological innovation, poised to transform numerous sectors, particularly healthcare [1].

Predictive analytics is indeed a critical discipline in the realm of data analytics, particularly in healthcare, where it holds immense potential to revolutionize patient care, operational efficiency, and disease management. Here's a breakdown of its significance across various healthcare domains:

Clinical Research: Predictive analytics enables researchers to analyse vast amounts of healthcare data to identify patterns, correlations, and predictive factors related to diseases, treatments, and outcomes. This informs the design and execution of clinical trials, leading to more efficient research processes and accelerated discoveries. Development of New Treatments: By analysing patient data and clinical trial results, predictive analytics helps pharmaceutical companies and researchers identify potential targets for new treatments and therapies. This data-driven approach enhances the efficiency of drug development pipelines and facilitates the discovery of innovative therapies. Discovery of New Drugs: According to [22], predictive analytics in healthcare is vital for identifying patterns, correlations, and predictive factors related to diseases, treatments, and outcomes. It informs the design and execution of clinical trials, leading to more efficient research processes and accelerated discoveries. Prediction and Prevention of Diseases: Healthcare analytics enables early identification of individuals at risk of developing certain diseases or conditions based on their medical history, genetics, and lifestyle factors. This proactive approach allows healthcare providers to implement preventive interventions and personalized healthcare strategies to mitigate disease risk and improve patient outcomes.

Clinical Decision Support: Predictive analytics tools provide healthcare professionals with real-time insights and decision support to aid in clinical decision-making. By analysing patient data and clinical guidelines, these systems help clinicians make informed decisions about diagnosis, treatment plans, and patient management, leading to better outcomes and enhanced patient safety.

Quicker, More Accurate Diagnosis: Predictive analytics enhances diagnostic accuracy by analysing patient data and identifying subtle patterns or anomalies indicative of disease. This facilitates early detection and diagnosis of medical conditions, leading to timely interventions and improved patient outcomes. High Success Rates of Surgeries and Medications: Predictive analytics helps optimize surgical outcomes and medication efficacy by predicting patient responses and identifying factors that contribute to treatment success or failure. This personalized approach ensures that patients receive the most effective treatments tailored to their individual characteristics and needs. . Privacy, data security, ethical implications, and the indispensable requirement for human oversight in AI-driven decision-making processes are vital considerations. These factors necessitate a comprehensive evaluation of AI's impact on healthcare analytics, encompassing limitations, challenges, and ethical dimensions [7].

Automation of Hospital Administrative Processes: Healthcare analytics streamlines administrative processes within hospitals and healthcare organizations by automating tasks such as scheduling, billing, and inventory management. This improves operational efficiency, reduces administrative burdens on staff, and enhances overall healthcare delivery.

In summary, predictive analytics has a profound impact on various aspects of healthcare, ranging from clinical research and treatment development to disease prevention, clinical decision support, and operational efficiency. By harnessing the power of data analytics, the healthcare industry can unlock valuable insights to improve patient outcomes, optimize resource allocation, and drive innovation in patient care.

Predictive analytics in healthcare offers numerous benefits that significantly improve patient care and operational efficiency. Here are some key advantages:

1. Improved Patient Care: Predictive analytics leverages various data sources, including medical history, demographics, and comorbidities, to provide healthcare professionals with valuable insights. These insights guide decision-making, leading to better, data-driven care strategies tailored to each patient's needs.

2. Identification of At-Risk Patients: Predictive analytics can identify patients at higher risk of adverse health outcomes, allowing for early interventions to prevent complications. For example, it can predict the likelihood of hospitalization for patients with cardiovascular disease based on factors such as age, chronic illnesses, and medication adherence. Proactive care for at-risk patients reduces the burden on healthcare systems and improves patient outcomes.

3. Proactive Care Management: By predicting disease likelihood and identifying at-risk populations, healthcare organizations can implement proactive care management strategies. This may involve targeted interventions, and patient education programs aimed at preventing disease progression and reducing hospital readmissions.

4. Optimized Resource Allocation: Predictive analytics helps healthcare providers allocate resources more efficiently by anticipating patient needs and demand for services. This includes optimizing staffing levels, bed utilization, and medical supply inventory to ensure timely and effective care delivery.

5. Enhanced Population Health Management: Predictive analytics enables healthcare organizations to monitor and manage the health of populations more effectively. By analyzing trends and patterns in health data, providers can identify population health risks, implement preventive measures, and allocate resources strategically to address community health needs.

6. Reduced Healthcare Costs: By identifying at-risk patients early and preventing costly complications, predictive analytics can help reduce healthcare costs. Proactive interventions, such as medication adherence programs and chronic disease management, can lead to fewer hospitalizations, emergency room visits, and other expensive healthcare services.

7. Improved Clinical Outcomes: Predictive analytics supports evidence-based decision-making, leading to improved clinical outcomes for patients. By identifying optimal treatment pathways, predicting treatment responses, and avoiding adverse events, healthcare professionals can achieve better patient outcomes and satisfaction.

Overall, predictive analytics in healthcare empowers providers with actionable insights that improve patient care, optimize resource utilization, and drive positive health outcomes across populations. By harnessing the power of data and technology, healthcare organizations can transform the way they deliver care and address the evolving needs of patients and communities.

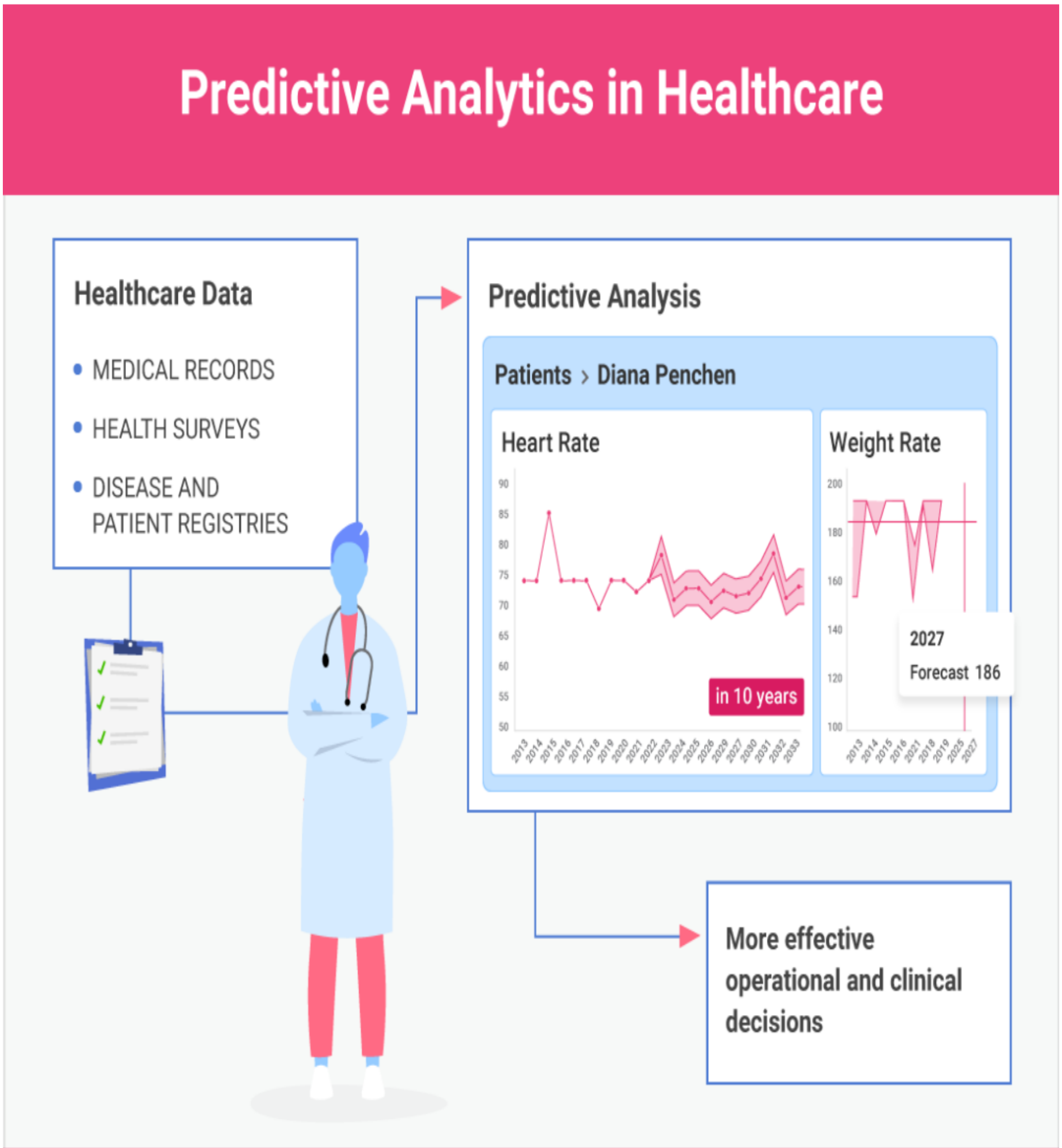


Fig 1.1: Predictive Healthcare Analytics

1.2 HISTORY OF PREDICTIVE ANALYTICS

The origin of healthcare analytics can indeed be traced back to the mid-20th century, particularly the 1950s, which saw the emergence of early attempts to apply computational methods to medical diagnosis and treatment. This period coincided with the advent of computers and the growing interest in leveraging them for various scientific and practical applications, including healthcare.

During the 1950s, medical records were predominantly paper-based, and computational analytics was limited by the available technology. However, there was a notable interest among researchers in developing automated applications to aid in medical diagnosis and treatment.

One significant milestone reflecting this interest is the publication of the article "Reasoning Foundations of Medical Diagnosis" in the journal Science in 1959. This article, authored by Robert S. Ledley and Lee B. Lusted, delved into the mathematical underpinnings of medical diagnosis. While the terminology and symbols used may differ from modern biostatistics, the concepts presented laid the groundwork for future developments in healthcare analytics.

Ledley and Lusted's work explained how physicians make medical diagnoses using mathematical reasoning, providing insights into the processes involved in clinical decision-making. Although their approach may have been rudimentary by today's standards, it demonstrated an early recognition of the potential for computational methods to enhance medical practice.

Overall, the 1950s marked the beginning of efforts to apply computational and analytical techniques to healthcare, setting the stage for the subsequent evolution of healthcare analytics into the sophisticated field it is today.

In the 1970s, with the increasing accessibility of computers in academic research centers, there was a surge of interest in developing Medical Diagnostic Decision Support (MDDS) systems. These systems aimed to provide comprehensive diagnostic assistance by utilizing patient information input into computer programs. One of the most prominent systems from this era was INTERNIST-1, developed by researchers at the University of Pittsburgh.

The INTERNIST-1 system, described as an "experimental program for computer-assisted diagnosis in general internal medicine," was a result of over 15 person-years of work and extensive collaboration with physicians. Its knowledge base encompassed 500 diseases and 3,500 clinical manifestations across various medical specialties. Users could input positive and negative findings for a patient, and the system would generate a list of potential diagnoses, which would evolve as new information was added. Despite initial enthusiasm and capturing the attention of the medical community, INTERNIST-1 failed to gain widespread adoption. Its diagnostic recommendations were ultimately outperformed by those of leading physicians. Several factors contributed to its demise, along with the failure of MDDS systems in general:

1. Performance Limitations: While INTERNIST-1 showcased promising capabilities, its diagnostic accuracy fell short when compared to expert clinicians. This discrepancy undermined its utility and credibility among healthcare professionals.
2. Interface Design: INTERNIST-1 lacked an intuitive visual interface, as graphical user interfaces (GUIs) akin to modern systems were not yet prevalent. The absence of user-friendly interaction may have hindered its acceptance and usability among clinicians.
3. Technological Constraints: The absence of advanced machine learning techniques limited the system's ability to adapt and improve over time. Without modern data-driven algorithms, MDDS systems struggled to keep pace with the complexity of medical diagnosis and decision-making.

While INTERNIST-1 and similar MDDS systems represented pioneering efforts in applying computational methods to healthcare, their limitations underscored the challenges of integrating technology into clinical practice during the nascent stages of digital health innovation. Nonetheless, these early endeavors laid the groundwork for subsequent advancements in healthcare informatics and decision support systems.

1.3 IMPORTANCE OF PREDICTIVE ANALYTICS

1. Early Disease Detection: Predictive analytics can help identify individuals at risk of developing certain diseases or conditions based on various factors such as genetics, lifestyle, and medical history. Early detection allows for timely interventions, potentially preventing the progression of diseases or minimizing their impact.
2. Personalized Medicine: By analyzing large datasets of patient information, predictive analytics can help healthcare providers tailor treatment plans to individual patients. This personalized approach ensures that patients receive the most effective treatments while minimizing adverse effects and unnecessary procedures.
3. Resource Optimization: Predictive analytics can forecast patient influxes, disease

outbreaks, and resource demands, enabling healthcare facilities to allocate staff, equipment, and other resources more efficiently. This leads to reduced wait times, improved patient satisfaction, and cost savings for healthcare organizations.

4. Risk Stratification: Healthcare providers can use predictive analytics to stratify patients based on their risk levels for various outcomes such as hospital readmission, complications, or mortality. This allows for targeted interventions and monitoring for high-risk patients, ultimately improving their outcomes and reducing healthcare costs associated with avoidable complications.

5. Chronic Disease Management: Predictive analytics can assist in managing chronic conditions by predicting exacerbations or complications before they occur. This proactive approach enables healthcare providers to intervene early, adjust treatment plans, and educate patients on self-management strategies, leading to better disease control and improved quality of life for patients.

6. Population Health Management: Predictive analytics can help public health officials and policymakers identify trends, hotspots, and high-risk populations for certain diseases or health issues. This information facilitates the development and implementation of targeted interventions, preventive measures, and public health campaigns to improve overall population health.

7. Research and Development: Predictive analytics can aid researchers in identifying patterns, correlations, and potential biomarkers associated with diseases, treatments, and outcomes. This information can accelerate the discovery of new treatments, drugs, and medical technologies, leading to advancements in healthcare and better patient care.

Overall, predictive healthcare analytics empowers healthcare providers, policymakers, and researchers with valuable insights that can enhance patient care, optimize healthcare delivery, and drive innovation in the healthcare industry.

1.4 APPLICATION OF PREDICTIVE ANALYTICS

Predictive analytics encompasses a suite of statistical analytics approaches that blend techniques from data mining, predictive modeling, AI, and machine learning. Its aim is to leverage extensive datasets containing established facts within a particular domain or situation to estimate probabilities for various potential future scenarios. Predictive analytics functions as a probabilistic tool to enhance decision-making within intricate systems that produce a diverse array of apparently unpredictable results.

Predictive analytics finds utility across diverse sectors and disciplines, such as:

1. Marketing
2. Finance
3. Security
4. Insurance
5. Law Enforcement
6. Business Administration
7. Social Media
8. Healthcare

Despite the array of applications, predictive analytics follows a
uniform procedure across various domains. Analysts leverage
extensive datasets of historical events to assign probability scores to
factors influencing the outcome of intricate events. This method
typically involves a standardized three-step approach.
1. Logistic Regression2. Time Series Analytics3. Decision TreesHere
are five promising healthcare application for predictive analytics:

1. Clinical Prediction:

Healthcare providers can utilize machine learning algorithms to analyze patient data and past medical outcomes, enabling them to forecast future trends in clinical settings. By leveraging analytics to identify probabilities, care providers can proactively anticipate the onset of various medical conditions.

During the Covid-19 pandemic, predictive analytics played a significant role in clinical assessments. One notable example is the US National COVID Cohort Collaborative, which applied machine learning models to patient data from 174,568 prior Covid hospitalizations. This approach aimed to map severity trajectories over time. By leveraging the study's findings to identify high-risk patients early, healthcare providers successfully reduced the mortality rate among patients under their care from 16.4% in March 2020 to 8.6% by September of the same year.

2. Assessing Patient Engagement and Behavior:

Time and resources lost due to missed appointments and disregarded treatment regimens pose significant challenges for healthcare organizations, hindering their ability to allocate resources effectively. Implementing predictive analytics and leveraging behavioural science with patient data has demonstrated success in identifying individual at high risk for no-shows or non-adherence to treatment plans. Armed with these insights, administrators can proactively schedule interventions or increase the frequency of patient contact to improve overall engagement levels.

3. Early Intervention for Diseasing Progression and Comorbidities:

Physicians have long recognized the critical importance of early intervention in managing progressive diseases and conditions with comorbidities. However, effective early intervention has often relied on educated guesses rather than precise data analytics, particularly when dealing with large datasets that may contain outliers.

In recent times, healthcare institutions have embraced predictive analytical methods to sift through vast amounts of patient data, particularly in cases of progressive diseases. This approach has yielded promising outcomes, especially in risk assessment for cardiovascular diseases and complications related to diabetes. By enhancing treatment effectiveness for prevalent chronic conditions, healthcare organizations can create a ripple effect of long-term benefits. Fewer patients progress to critical stages of illness, allowing care providers to allocate resources more efficiently to address other healthcare needs.

4. Resource Allocation and Acquisition:

Healthcare organizations operate within intricate systems managing extensive networks of resources. Even slight misallocations can have significant repercussions, particularly for national-scale organizations. By leveraging predictive analytics to discern seasonal or regional patterns in resource utilization, organizations can cultivate more adaptable resource management strategies, proactively addressing future demands. This proactive approach enables organizations to optimize their resource allocation, ultimately improving efficiency and effectiveness in healthcare delivery.

5. Hospital Overstays and Readmission:

Healthcare organizations face the challenge of managing critical yet limited resources, such as hospital beds, to deliver optimal care to the broader patient community. In traditional systems, hospital overstays often accumulate, leading to increased costs, longer patient wait times, and heightened risks of secondary infections.

Conversely, premature patient discharges may elevate the likelihood of readmission, compromising patient outcomes. Predictive analytics offers a solution by analyzing patient data, including age, medical history, and condition, to determine the average duration of hospitalization. Armed with this insight, organizations can adjust treatment plans and resource allocation more effectively, identifying outliers early through established criteria. This proactive approach enables healthcare organizations to optimize the utilization of beds and rooms while minimizing the risks associated with both prolonged stays and premature discharges.

1.5 MOTIVATION

The motivation for conducting a survey on predictive healthcare analytics using AI/ML techniques stems from the urgent need to revolutionize healthcare delivery, improve patient outcomes, and optimize resource allocation. Predictive analytics, fueled by advancements in AI and ML, has the potential to transform the healthcare landscape by leveraging vast amounts of data to predict patient outcomes, identify high-risk individuals, and personalize treatment.

Furthermore, in the context of healthcare, predictive analytics holds immense promise for improving clinical decision-making, enhancing population health management, and reducing healthcare costs. By analyzing diverse data sources, including electronic health records (EHRs), medical imaging, genomic data, and real-time physiological measurements, predictive models can forecast disease onset, stratify patient risk, and optimize treatment protocols. This proactive approach to healthcare enables early intervention, preventive care, and targeted interventions, ultimately leading to better health outcomes and improved quality of life for patients.

A survey on predictive healthcare analytics using AI/ML lies in its potential to revolutionize healthcare delivery, improve patient outcomes, and address pressing healthcare challenges. By synthesizing existing methodologies, identifying key findings, and highlighting future directions, this survey aims to provide valuable insights for researchers, healthcare professionals, policymakers, and industry stakeholders.

CHAPTER 2

LITERATURE REVIEW

The field of predictive healthcare analytics, empowered by Artificial Intelligence (AI) and Machine Learning (ML) techniques, has witnessed exponential growth and garnered considerable attention in recent years. This literature review aims to provide a comprehensive overview of the current methodologies, applications, challenges, and future directions in predictive healthcare analytics using AI/ML.

Methodologically, supervised learning techniques such as logistic regression, decision trees, and neural networks have been extensively applied for predictive modeling in healthcare. These techniques enable the prediction of clinical outcomes, disease diagnosis, and treatment response by learning patterns from labeled data. Deep learning algorithms, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have demonstrated remarkable success in medical imaging analytics, clinical natural language processing, and predictive modeling from electronic health records (EHRs). Moreover, unsupervised learning methods, such as clustering and anomaly detection, play a crucial role in uncovering hidden patterns and insights from unlabeled healthcare data, facilitating patient segmentation, population health management, and outlier detection.

In terms of applications, predictive healthcare analytics has been applied across various domains, including disease diagnosis and prognosis, personalized medicine, and healthcare resource allocation. Predictive models assist in the early detection and accurate diagnosis of diseases, estimate disease progression, and predict patient outcomes, thereby informing treatment decision-making and care planning. Additionally, AI/ML-based predictive analytics enable the delivery of personalized healthcare interventions tailored to individual patient characteristics and genetic profiles, improving treatment efficacy and patient outcomes. Furthermore, predictive analytics facilitates efficient resource allocation and healthcare resource management by forecasting patient demand, predicting healthcare utilization rates, and identifying high-risk patient populations, thereby optimizing operational efficiency and reducing costs.

However, predictive healthcare analytics is not without challenges and considerations. Data quality, interoperability, and privacy concerns pose significant challenges in leveraging healthcare data for predictive modeling. Ethical and regulatory compliance, including patient privacy, data security, and algorithmic bias, must be carefully addressed to ensure the responsible and ethical use of healthcare data and AI/ML algorithms. Moreover, the black-box nature of some AI/ML models presents challenges in model interpretability and transparency, hindering clinician acceptance and patient trust.

Looking ahead, future directions and emerging trends in predictive healthcare analytics include the integration of multimodal data, the development of interoperable AI ecosystems, and advancements in explainable AI and model interpretability. By embracing these trends and fostering interdisciplinary collaboration, the healthcare industry can harness the power of AI/ML to improve patient outcomes, enhance operational efficiency, and advance population health management.

By conducting an exhaustive review of existing literature, meticulous data analytics, and scrutinizing real-world case studies, this study aims to offer meaningful insights into how artificial intelligence can reshape the landscape of healthcare analytics and drive transformative changes in healthcare delivery methods [5].

Several researchers have proposed frameworks and methodologies aimed at improving healthcare systems and analytics:

1. Information Disclosure, Examination and Expectation in Medical services utilizing Information Mining and Analytics" by Akshay Raul et al.: This paper suggests a framework designed to create public awareness about alternative medications for specific drugs and their availability within a region. The proposed system assists patients in understanding substitute medications recommended by their doctors, utilizing information mining and analytics [10].

2. Medical care Information Investigation utilizing Dynamic Opening Assignment in Hadoop by Aditi Bansal and Priyanka Ghare: This work introduces a healthcare framework utilizing Hadoop and the Dynamic Hadoop Opening Assignment (DHSA) technique. The focus lies on improving the performance of MapReduce jobs and maintaining the system efficiently [12].

3. Large Information Stream Figuring in Medical care Ongoing Examination by Van-Dai Ta, Chuan-Ming Liu, Generosity Wandile Nkabinde: This paper presents a traditional architecture for large-scale healthcare analytics[11]. It leverages open-source tools such as Hadoop, Apache Tempest, Kafka, and NoSQL Cassandra [19].

4. Security Challenges in Healthcare Systems: For a comprehensive review on security challenges in healthcare systems by Razaque, A., Amsaad, F., Khan, M. J., Hariri, S., Chen, S., Chen, S., & Ji, X. (2019). Survey: Cybersecurity vulnerabilities, attacks, and solutions in the medical domain. IEEE Access, 7, 168774–168797. This paper provides insights into cybersecurity vulnerabilities, attacks, and solutions specific to the medical domain, contributing to the enhancement of healthcare systems and the development of robust security measures [3].

5. Huge information examination in medical services: commitment and Potential by Wullianallur Raghupathi and Viju Raghupathi: The authors discuss the potential of big data analytics in healthcare and provide insights beneficial for healthcare professionals and researchers. They advocate for a general architecture employing open-source tools like Hadoop, Apache Tempest, Kafka, and NoSQL Cassandra [13].

However, these works primarily concentrate on extracting features from specific healthcare data sources or focus solely on batch-oriented processing methods, resulting in higher latency. There is a need to consider continuous data influx, varied data formats, and substantial volume in the proposed architecture. Furthermore, research on autonomic computing, inspired by human capabilities, has garnered attention. Khalid et al. categorized different types of autonomic architectures [17].

Naturally Inspired Architecture: Modelled after human body functionality for self-regulation.

Architecture for Large-Scale Distributed Systems: Developed by IBM and Microsoft for large-scale system control.

Moreover, Kumar and Sharma proposed a vulnerability detection model employing autonomic computing to minimize software vulnerabilities [20].

Pena et al. demonstrated a model-driven approach for self-strategy-based systems during runtime processes [18].

CHAPTER 3

DESIGNING AND METHODOLOGY

3.1 INTRODUCTION TO DESIGNING:

Gathering the Data: Data preparation is the primary step for any machine learning problem. We will be using a dataset from Kaggle for this problem. This dataset consists of two CSV files, one for training and one for testing. There are a total of 133 columns in the dataset, out of which 132 columns represent the symptoms, and the last column is the prognosis.

Cleaning the Data: Cleaning is the most important step in a machine learning project. The quality of our data determines the quality of our machine-learning model. So it is always necessary to clean the data before feeding it to the model for training. In our dataset, all the columns are numerical. The target column, i.e., prognosis, is of string type and is encoded to numerical form using a label encoder.

Model Building: After gathering and cleaning the data, it's ready to be used to train a machine learning model. We will be using this cleaned data to train the Support Vector Classifier, Naive Bayes Classifier, and Random Forest Classifier. We will use a confusion matrix to determine the quality of the models.

Inference: After training the three models, we will predict the disease for the input symptoms by combining the predictions of all three models. This makes our overall prediction more robust and accurate.

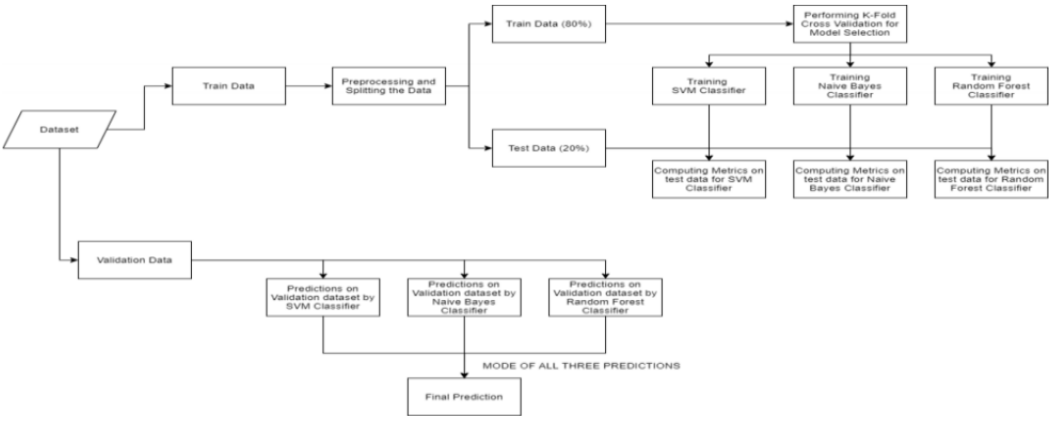

Dataset
Train Data
Preprocessing and Splitting the Data
Train Data (80%)
Test Data (20%)
Performing K-Fold Cross Validation for Model Selection
Training SVM Classifier
Training Naive Bayes Classifier
Training Random Forest Classifier
Computing Metrics on test data for SVM Classifier
Computing Metrics on test data for Naive Bayes Classifier
Computing Metrics on test data for Random Forest Classifier
Validation Data
Predictions on Validation dataset by SVM Classifier
Predictions on Validation dataset by Naive Bayes Classifier
Predictions on Validation dataset by Random Forest Classifier
MODE OF ALL THREE PREDICTIONS
Final Prediction

Fig 3.1: Training of Machine Learning

Make sure that the Training and Testing are downloaded and the train.csv, test.csv are put in the dataset folder. Open jupyter notebook and run the code individually for better understanding.

3.2 Reading the dataset

Firstly we will be loading the dataset from the folders using the pandas library. While reading the dataset we will be dropping the null column. This dataset is a clean dataset with no null values and all the features consist of 0's and 1s. Whenever we are solving a classification task it is necessary to check whether our target column is balanced or not. We will be using a bar plot, to check whether the dataset is balanced or not.

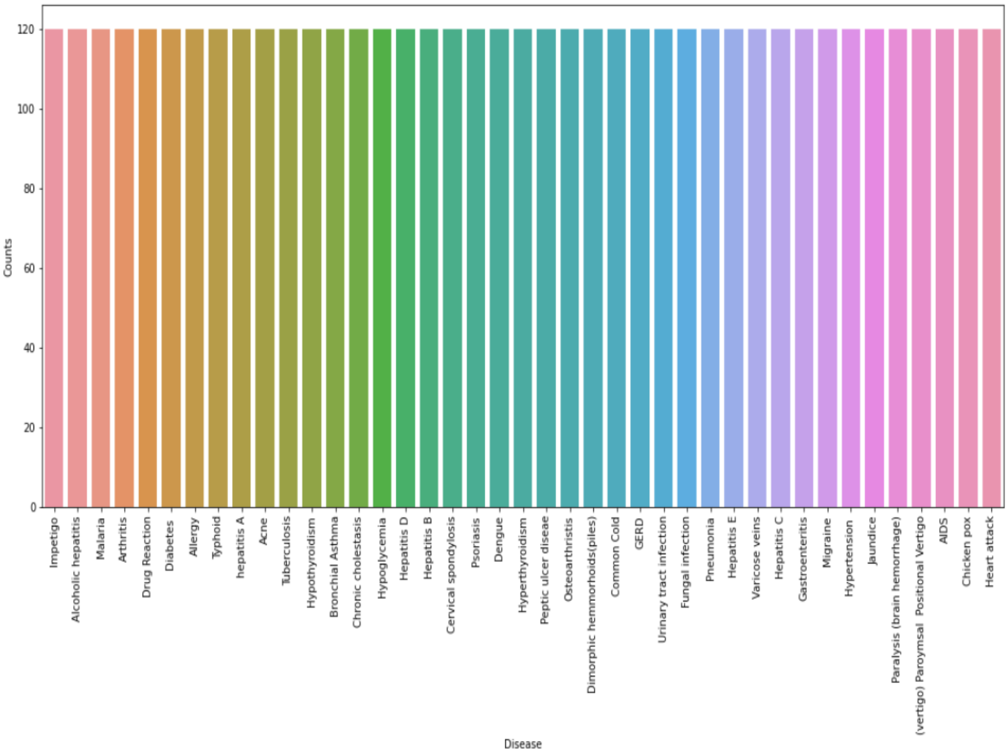
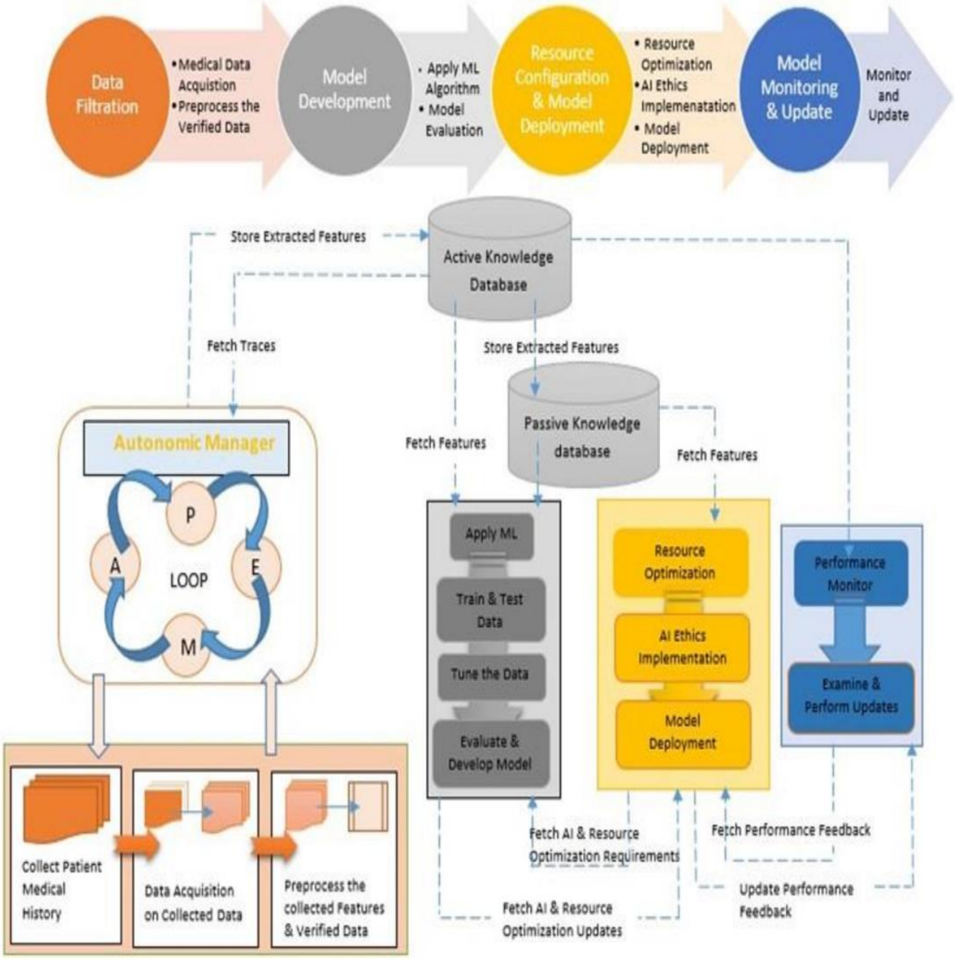


Fig 3.2: Balanced Data Sheet

From the above plot, we can observe that the dataset is a balanced dataset i.e. there are exactly 120 samples for each disease, and no further balancing is required. We can notice that our target column i.e. prognosis column is of object datatype, this format is not suitable to train a machine learning model. So, we will be using a label encoder to convert the prognosis column to the numerical datatype. Label Encoder converts the labels into numerical form by assigning a unique index to the labels. If the total number of labels is n, then the numbers assigned to each label will be between 0 to n-1.

3.3 Splitting the data for training and testing the model

Now that we have cleaned our data by removing the Null values and converting the labels to numerical format, It's time to split the data to train and test the model. We will be splitting the data into 80:20 format i.e. 80% of the dataset will be used for training the model and 20% of the data will be used to evaluate the performance of the models.



Data Filtration
•Medical Data Acquistion
•Preprocess the Verified Data
Model Development
• Apply ML Algorithm
• Model Evaluation
Resource Configuration & Model Deployment
• Resource Optimization
•AI Ethics Implemenatation
• Model Deployment
Model Monitoring & Update
Monitor and Update
Store Extracted Features
Active Knowledge Database
Fetch Traces
Store Extracted Features
Passive Knowledge database
Fetch Features
Fetch Features
Autonomic Manager
P
A
E
M
LOOP
Apply ML
Train & Test Data
Tune the Data
Evaluate & Develop Model
Resource Optimization
AI Ethics Implementation
Model Deployment
Performance Monitor
Examine & Perform Updates
Collect Patient Medical History
Data Acquisition on Collected Data
Preprocess the collected Features & Verified Data
Fetch AI & Resource Optimization Requirements
Fetch Performance Feedback
Update Performance Feedback
Fetch AI & Resource Optimization Updates

Fig 3.3: Proposed Methodology

3.4 Model Building

After splitting the data, we will be now working on the modeling part. We will be using K-Fold cross-validation to evaluate the machine-learning models. We will be using Support Vector Classifier, Gaussian Naive Bayes Classifier, and Random Forest Classifier for cross-validation. Before moving into the implementation part let us get familiar with k-fold cross-validation and the machine learning models.

1. K-Fold Cross-Validation**:** K-Fold cross-validation is one of the cross-validation techniques in which the whole dataset is split into k number of subsets, also known as folds, then training of the model is performed on the k-1 subsets and the remaining one subset is used to evaluate the model performance.

2. Support Vector Classifier: Support Vector Classifier is a discriminative classifier i.e. when given a labeled training data, the algorithm tries to find an optimal hyperplane that accurately separates the samples into different categories in hyperspace.

3. Gaussian Naive Bayes Classifier: It is a probabilistic machine learning algorithm that internally uses Bayes Theorem to classify the data points.

4. Random Forest Classifier: Random Forest is an ensemble learning-based supervised machine learning classification algorithm. In a random forest classifier, all the internal decision trees are weak learners, and the outputs of these weak decision trees are combined i.e. mode of all the predictions is as the final prediction.

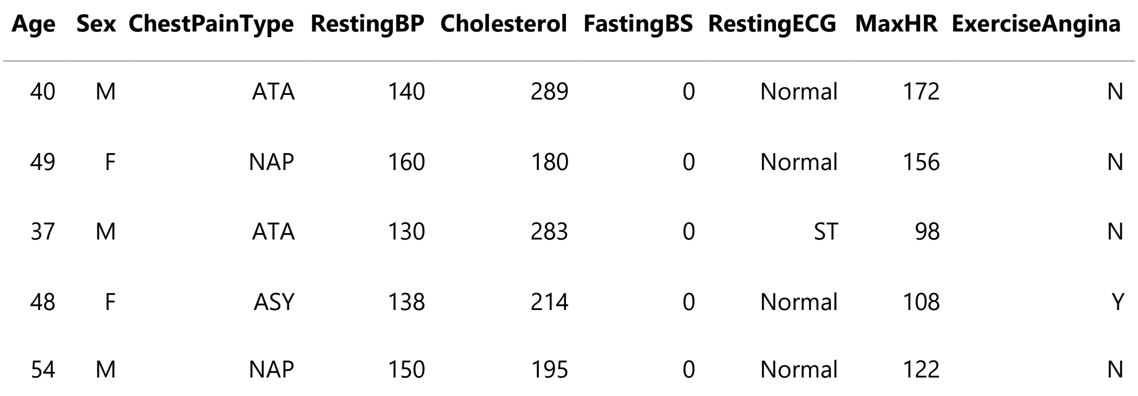
CHAPTER 4

IMPLEMENTATION

```
import os
import numpy as np import pandas as pd import warnings
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
warnings.filterwarnings("ignore")
pd.set_option("display.max_rows",None)
%matplotlib inline
from sklearn import preprocessing
import matplotlib
matplotlib.style.use('ggplot')
from sklearn.preprocessing import LabelEncoder
lab = LabelEncoder()
```

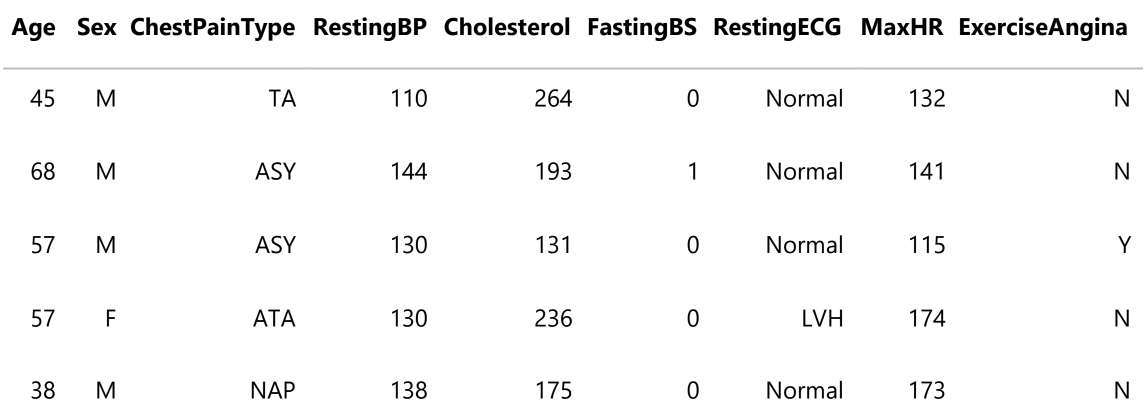
Read the CSV File or Data Collection

```
df =pd.read_csv("C:/Users/saifm/Downloads/heart.csv")
df.head()
```



Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina
40	M	ATA	140	289	0	Normal	172	N
49	F	NAP	160	180	0	Normal	156	N
37	M	ATA	130	283	0	ST	98	N
48	F	ASY	138	214	0	Normal	108	Y
54	M	NAP	150	195	0	Normal	122	N

Table 4.1: Age-Sex with Diseases



Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina
45	M	TA	110	264	0	Normal	132	N
68	M	ASY	144	193	1	Normal	141	N
57	M	ASY	130	131	0	Normal	115	Y
57	F	ATA	130	236	0	LVH	174	N
38	M	NAP	138	175	0	Normal	173	N

Table 4.2: Age-Sex with Diseases

```
<class 'pandas.core.frame.DataFrame'> RangeIndex: 918 entries, 0 to 917
Data columns (total 12 columns):
#     Column         Non-Null Count   Dtype
0     Age            918   non-null   int64
1     Sex            918   non-null   object
2     ChestPainType  918   non-null   object
3     RestingBP      918   non-null   int64
4     Cholesterol    918   non-null   int64
5     FastingBS      918   non-null   int64
6     RestingECG     918   non-null   object
7     MaxHR          918   non-null   int64
8     ExerciseAngina 918   non-null   object
9     Oldpeak        918   non-null   float64
10    ST_Slope       918   non-null   object
11    HeartDisease   918   non-null   int64
dtypes: float64(1), int64(6), object(5) memory usage: 86.2+ KB
```

Data Cleansing

```
df.isna().sum()
Age      0
```

```
Sex          0
ChestPainType     0
RestingBP         0
Cholesterol       0
FastingBS         0
RestingECG        0
MaxHR   0
ExerciseAngina    0
Oldpeak  0
ST_Slope          0
HeartDisease      0
dtype: int64
```

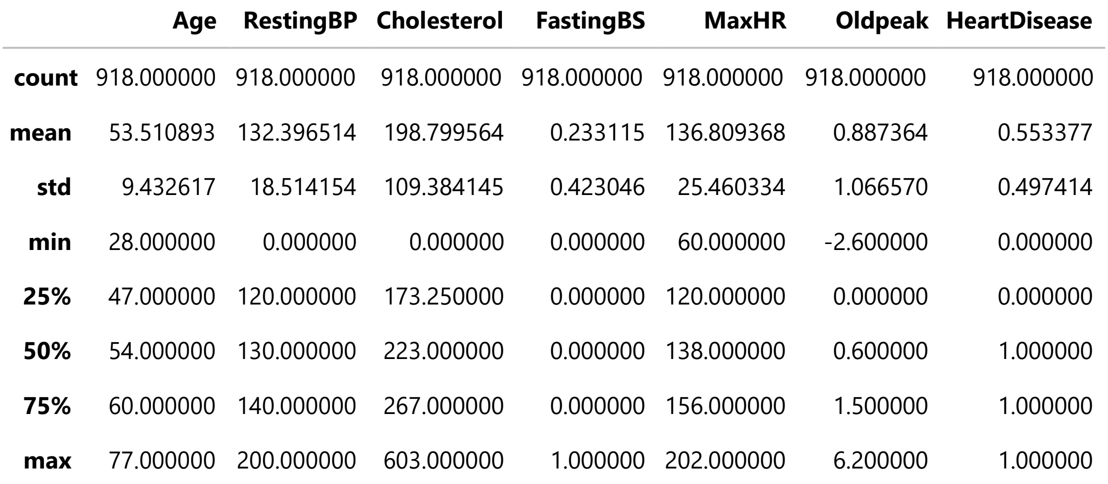
```
df1 = df.select_dtypes(exclude=object)
df2 = df.select_dtypes(include=object)
df3 = df1.drop('HeartDisease',axis=1)
```

```
RestingBP
Cholesterol FastingBS
MaxHR   int64
int64 int64 int64
Oldpeak
HeartDisease dtype: object   float64 int64
```

```
df2.dtypes()
```

```
Sex               object
ChestPainType     object
RestingECG        object
ExerciseAngina    object
 ST_Slope         object
dtype:            object
```

```
df1.describe() #this function gives the summary of the statistics
```

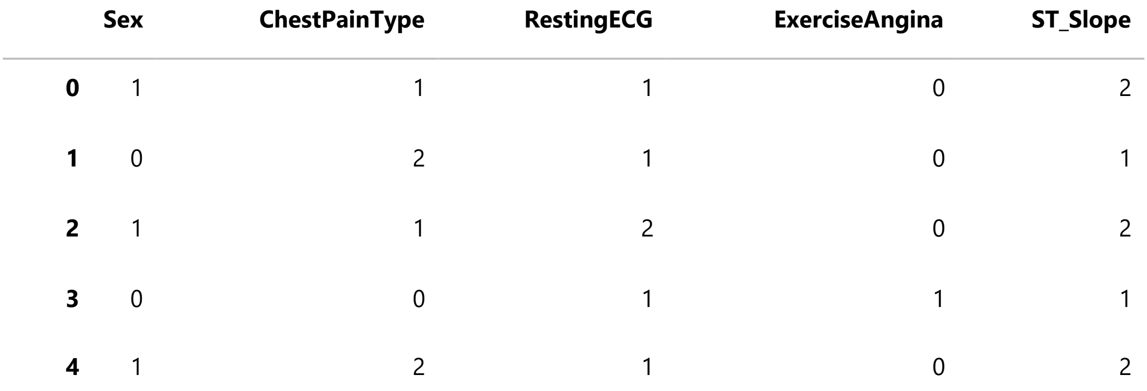


	Age	RestingBP	Cholesterol	FastingBS	MaxHR	Oldpeak	HeartDisease
count	918.000000	918.000000	918.000000	918.000000	918.000000	918.000000	918.000000
mean	53.510893	132.396514	198.799564	0.233115	136.809368	0.887364	0.553377
std	9.432617	18.514154	109.384145	0.423046	25.460334	1.066570	0.497414
min	28.000000	0.000000	0.000000	0.000000	60.000000	-2.600000	0.000000
25%	47.000000	120.000000	173.250000	0.000000	120.000000	0.000000	0.000000
50%	54.000000	130.000000	223.000000	0.000000	138.000000	0.600000	1.000000
75%	60.000000	140.000000	267.000000	0.000000	156.000000	1.500000	1.000000
max	77.000000	200.000000	603.000000	1.000000	202.000000	6.200000	1.000000

Table 4.3: Age – Diseases with certain counts

Processing Data

```
for i in range(0,df2.shape[1]):
        df2.iloc[:,i] = lab.fit_transform(df2.iloc[:,i])
df2.head()
```



	Sex	ChestPainType	RestingECG	ExerciseAngina	ST_Slope
0	1	1	1	0	2
1	0	2	1	0	1
2	1	1	2	0	2
3	0	0	1	1	1
4	1	2	1	0	2

Table 4.4: Age – Diseases without counts

```
df,ispa().sum()
Age                0
ResingBP           0
Cholestrol         0
FastingBS          0
Oldspeaks          0
Sex                0
ChestPainType      0
RestingECG         0
ST Slope           0
dtype:             int64
```

```
string_col=data.select_dtypes(include="object").columnsdata[string_col]=data[string
_col].
astype("string")
data.dtypes
```

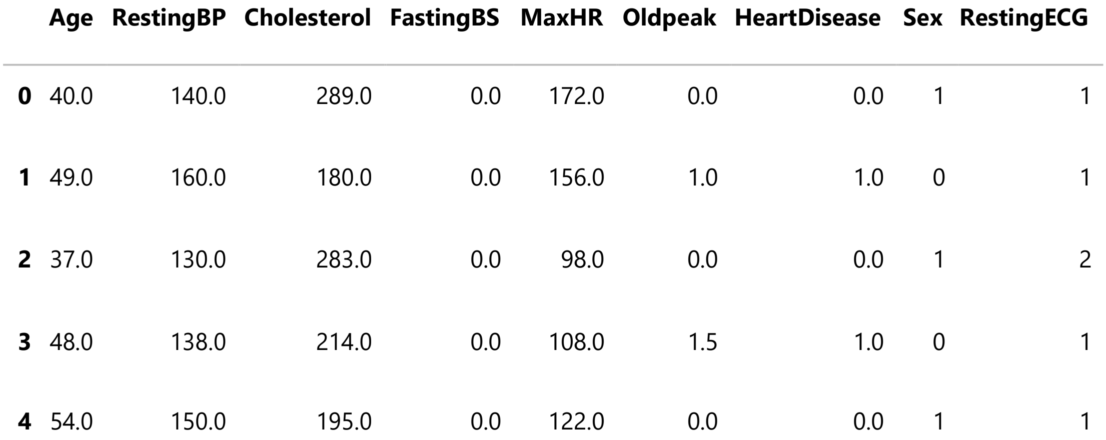
```
Age                int64
Sex                string[python]
ChestPainType      string[python]
Cholestrol         int64
RestingECG         string[python]
MaxHR              int64
ST Slope           string[python]
Heart Disease      int64
dtypes:            object
```

Getting the categorical columns

```
string_col = data.select_dtypes("string").columns.to_list()
num_col = data.columns.to_list()
print(num_col)
for col in string_col:
```

```
            num_col.remove(col)
num_col.remove("HeartDisease")
data.describe().T
heart_df = pd.read_csv('Datasets/heart.csv')
   st.title('Heart Checkup')
   st.sidebar.header('Patient Data')
   st.subheader('Training Data Stats')
   st.write(heart_df.describe())

   x = heart_df.drop(['target'], axis=1)
   y = heart_df['target']
   x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=0)
```



	Age	RestingBP	Cholesterol	FastingBS	MaxHR	Oldpeak	HeartDisease	Sex	RestingECG
0	40.0	140.0	289.0	0.0	172.0	0.0	0.0	1	1
1	49.0	160.0	180.0	0.0	156.0	1.0	1.0	0	1
2	37.0	130.0	283.0	0.0	98.0	0.0	0.0	1	2
3	48.0	138.0	214.0	0.0	108.0	1.5	1.0	0	1
4	54.0	150.0	195.0	0.0	122.0	0.0	0.0	1	1

Table 4.5: Final Age – Diseases Table

Exploratory Data Analytics

Correlation Matrix

```
px.imshow(df.corr(),title="Correlation Plot of the Heart Failure Prediction")
heart_df = pd.read_csv('Datasets/heart.csv')
st.title('Heart Checkup')
st.sidebar.header('Patient Data')
```

```
    st.subheader('Training Data Stats')
    st.write(heart_df.describe())
    st.title('Heart Checkup')
    st.sidebar.header('Patient Data')
    st.subheader('Training Data Stats')
    st.write(heart_df.describe())
    print(predictDisease("Itching,Skin Rash,Nodal Skin Eruptions"))
{'rf_model_prediction': 'Fungal infection', 'naive_bayes_prediction': 'Fungal infe ction',
'svm_model_prediction': 'Fungal infection', 'final_prediction': 'Fungal in fection'}
    x = heart_df.drop(['target'], axis=1)
    y = heart_df['target']
    x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=0

sns.heatmap(df.corr(),annot=True,linewidths=0.5)
plt.show()
```

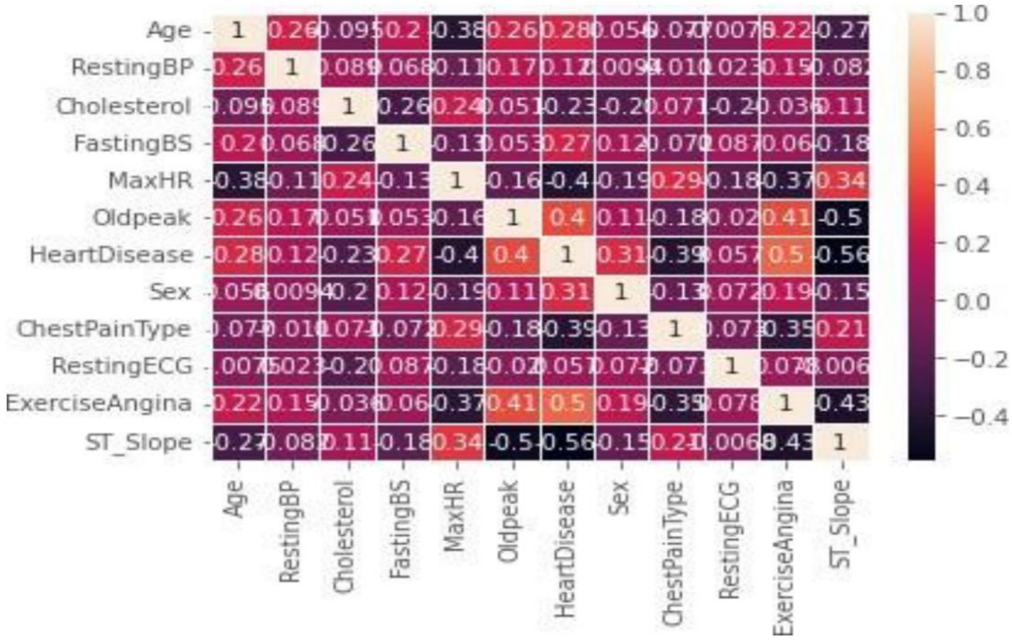


Fig 4.1: Heatmap for Heart Disease

```
df['HeartDisease'].value_counts()
# Shows the Distribution of Heart Diseases with respect to male and female
fig = px.histogram(data, x="HeartDisease", color="Sex",hover_data=data.columns,
title="Distribution of Heart Diseases", barmode="group")
fig.show()
```

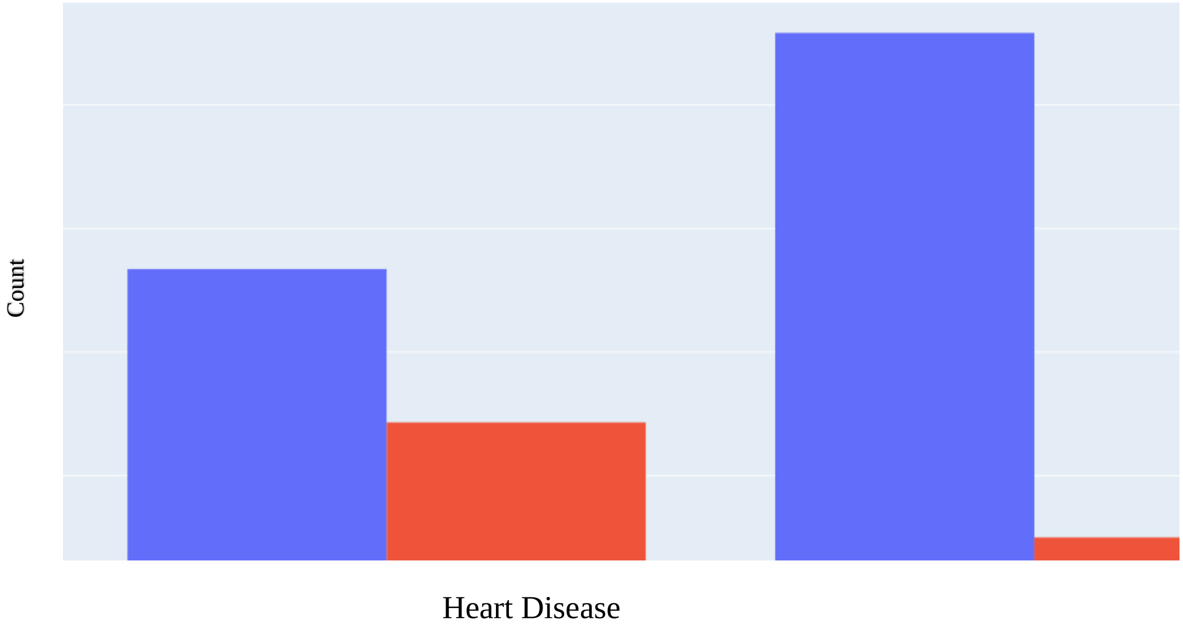


Fig 4.2: Distribution of Heart Disease

```
fig=px.histogram(data, x="ChestPainType", color="Sex",hover_data=data.columns,
title="Types of Chest Pain [1:Male,0:Female]")
fig.update_layout(bargap=0.2)
fig.show()
```

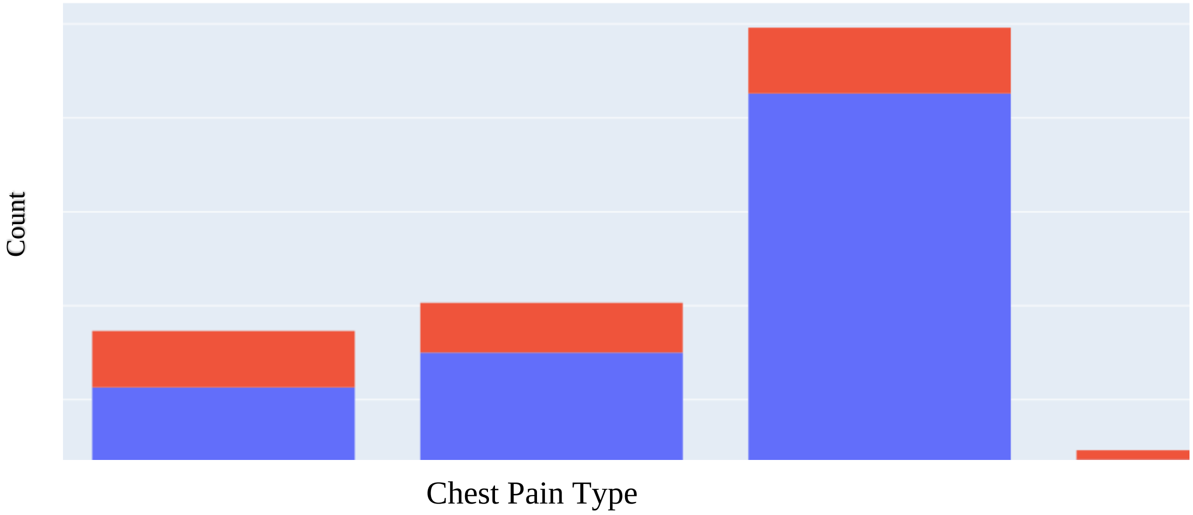


Fig 4.3: Types of Chest Pain [1-Male, 0-Female]

```
fig=px.histogram(data, x="Sex",hover_data=data.columns,
title="Sex Ratio in the Data")
fig.show()
```

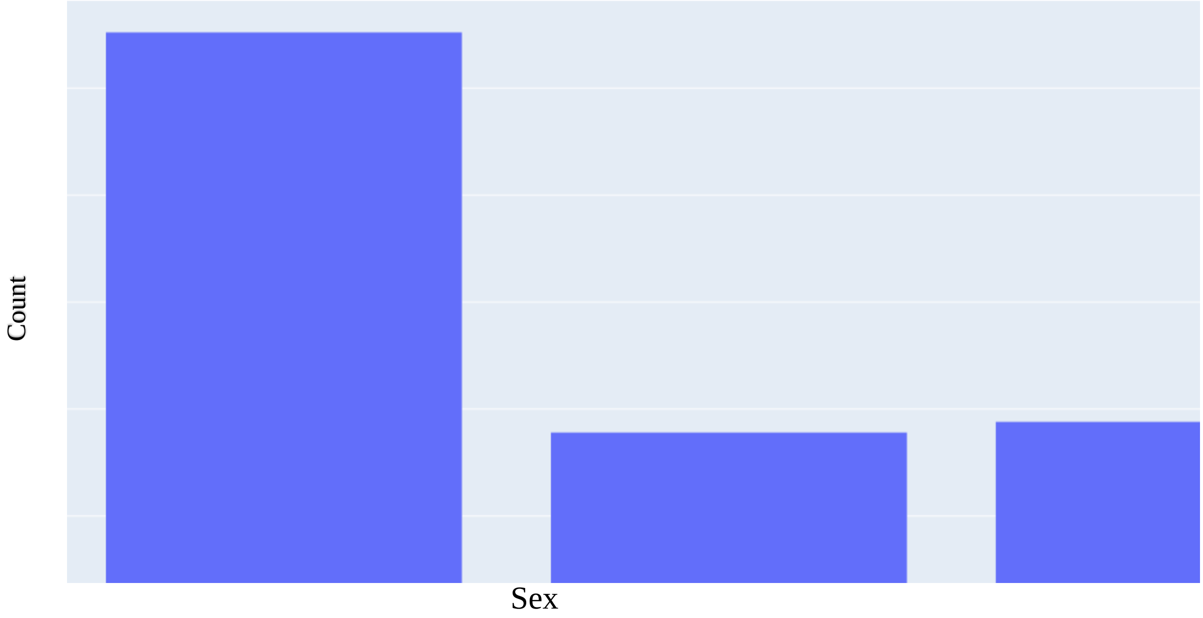


Fig 4.4: Distribution of RestingECG

```
plt.figure(figsize=(15,10))
sns.pairplot(df,hue="HeartDisease")
plt.title("Looking for Insites in Data") plt.legend("HeartDisease")
plt.tight_layout()
 plt.plot()
```

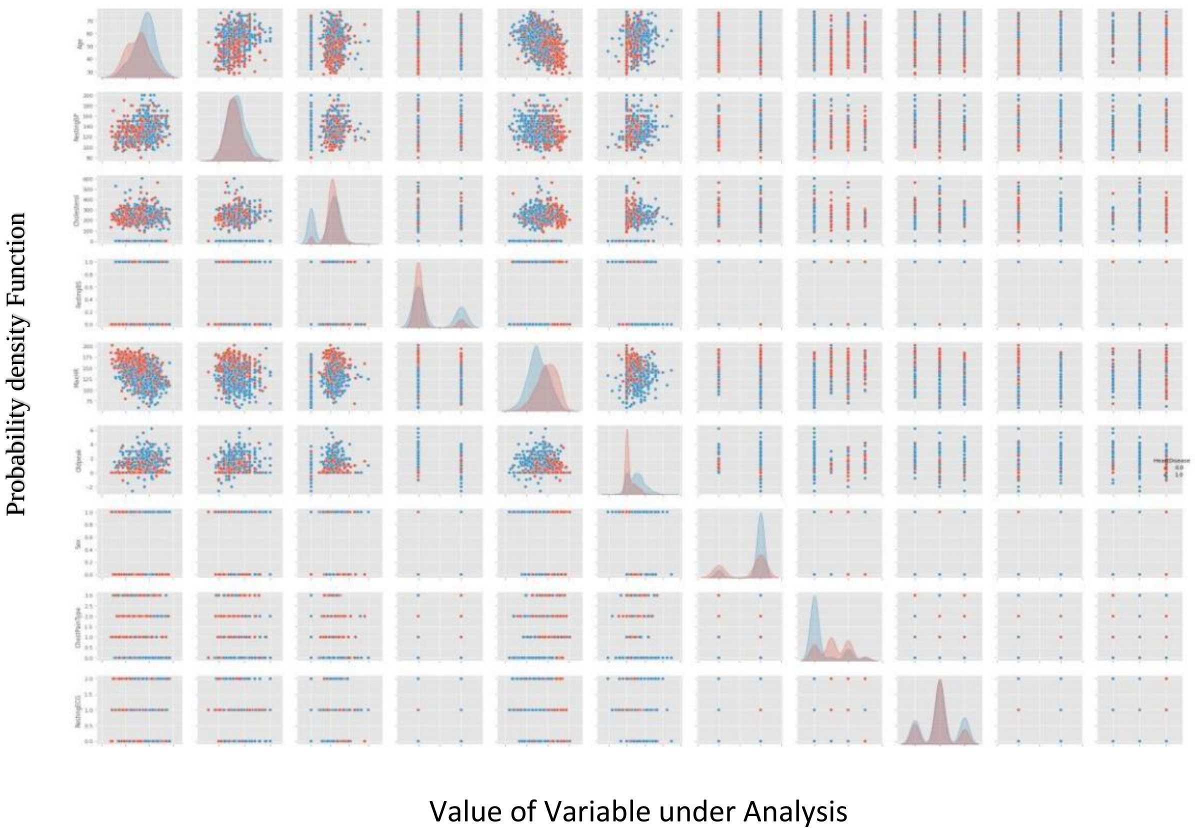
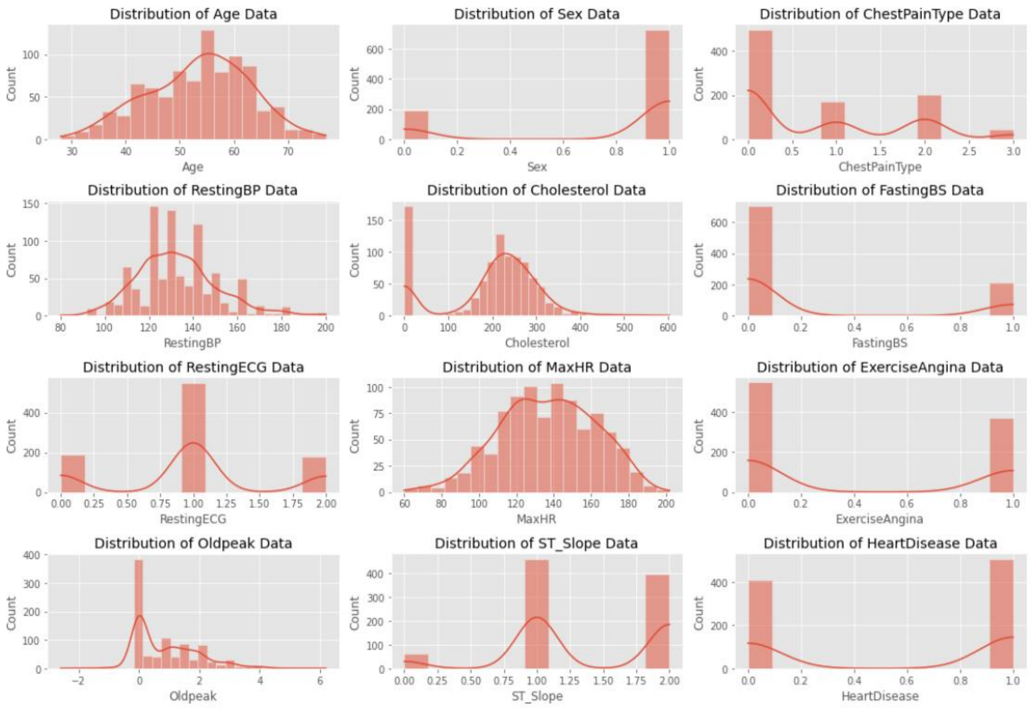


Fig 4.5: Kernel Density estimate KDE

```
plt.figure(figsize=(15,10))
for i,col in enumerate(data.columns,1): plt.subplot(4,3,i)
plt.title(f"Distribution of {col} Data")
sns.histplot(df[col],kde=True) plt.tight_layout()
plt.plot()
```

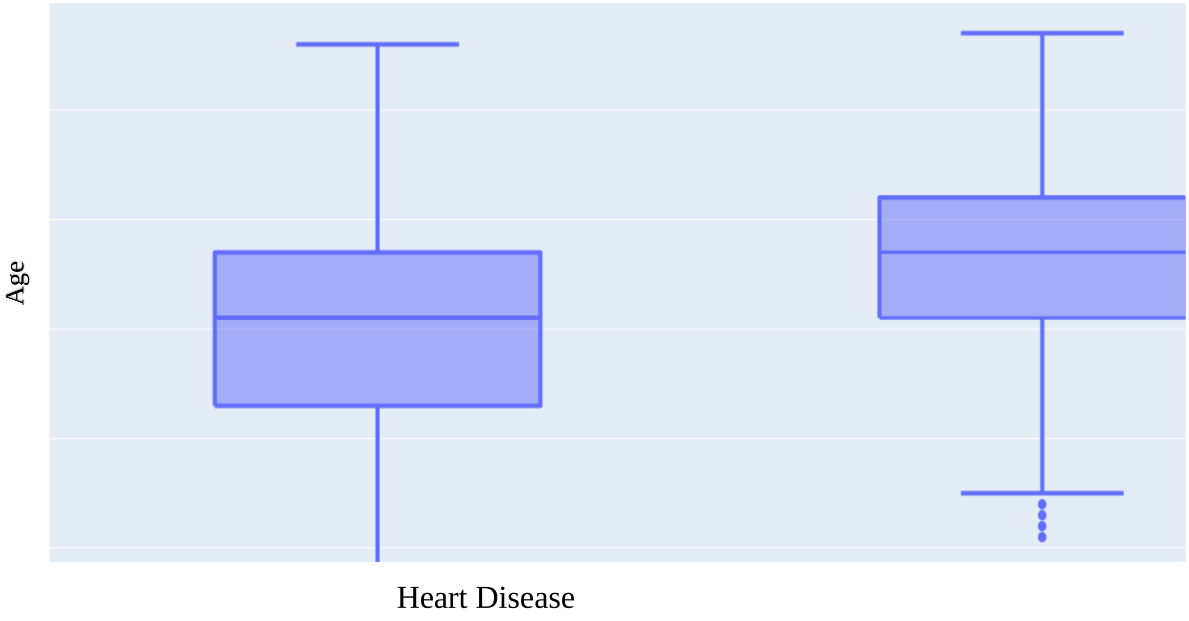
```
plt.figure(figsize=(10,15))
for i,col in enumerate(data.columns,4.3,i): plt.subplot(1)
plt.title(f"Distribution of {col} Data")
sns.histplot(df[col],kde=True) plt.tight_layout()
plt.plot()
```



Distribution Graph

Outliers

```
fig = px.box(data,y="Age",x="HeartDisease",title=f"Distribution of Age")
fig.show()
```



Heart Disease

Fig 4.7: Distribution of Age (Box Plot)

```
Fig=px.box(df,y="RestingBP",x="HeartDisease",title=f"DistributionofRestingBP"
,color="Sex")
fig.show()
```

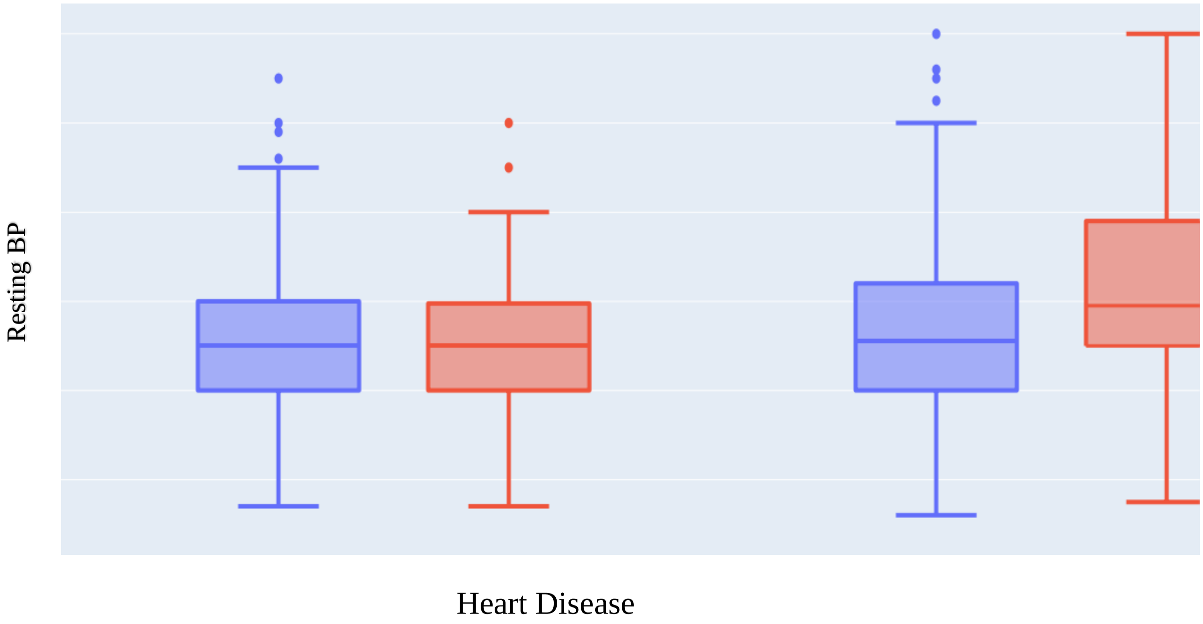


Fig 4.8: Distribution of RestingBP

```
fig = px.box(df,y="Cholesterol",x="HeartDisease",title=f"Distribution of Cholestrol")
fig.show()
```

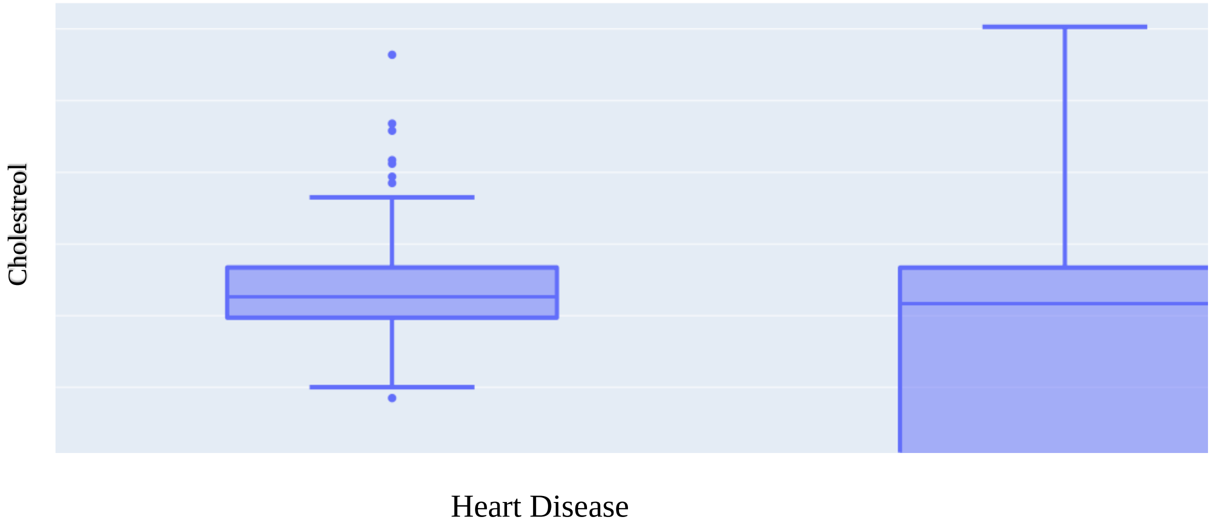


Fig 4.9: Distribution of Cholestrol

```
fig=px.box(df,y="Oldpeak",x="HeartDisease",title=f"Distribution of Oldpeak")
fig.show()
```

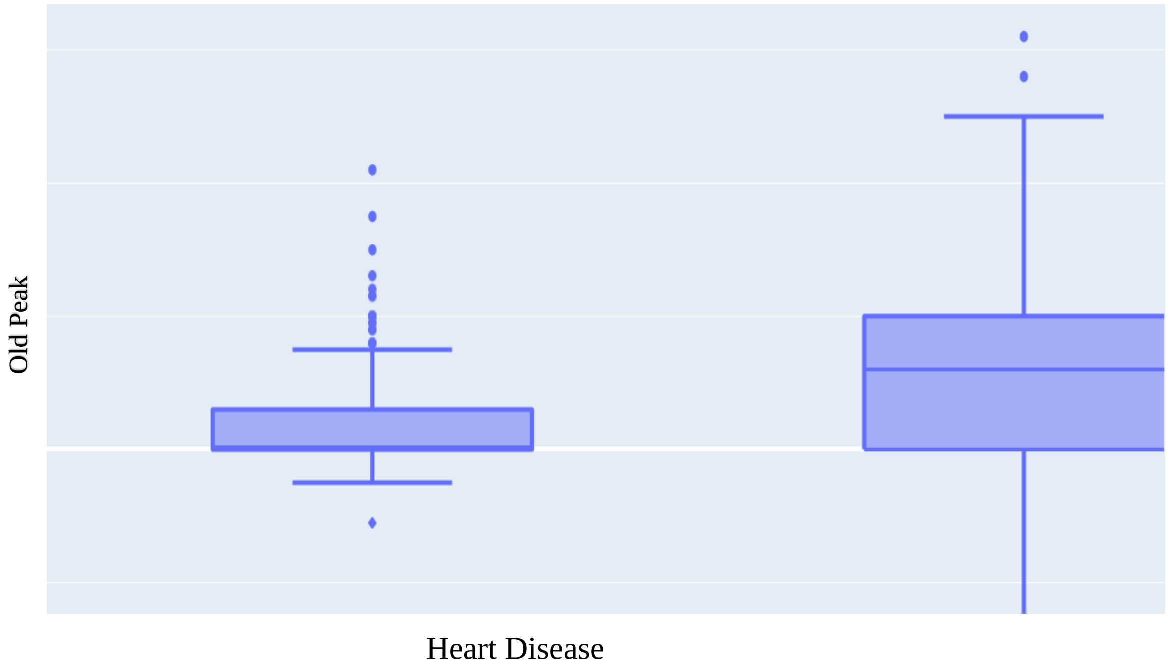


Fig 4.10: Distribution of OldPeak

```
fig=px.box(df,y="MaxHR",x="HeartDisease",title=f"Distribution of MaxHR")
fig.show()
```

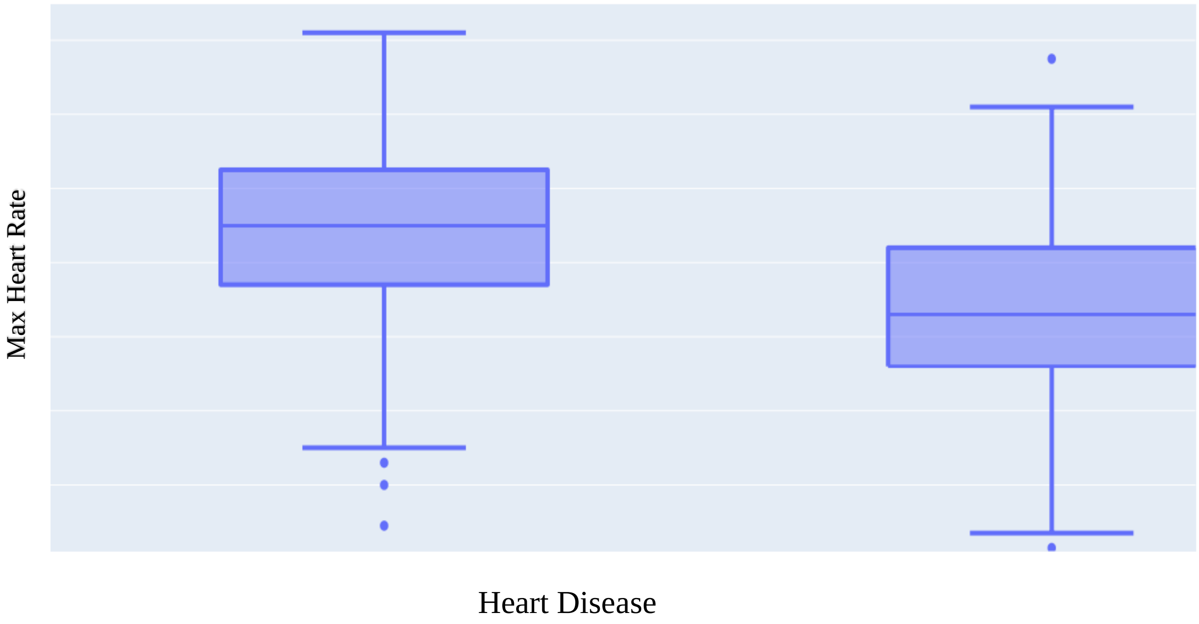


Fig 4.11: Distribution of MaxHR

```
x = pd.DataFrame({
#Distribution with lower outliers
'x1':np.concatenate([np.random.normal(20,2,1000),np.random.normal(1,2,25)]),
#Distribution with higher outliers
'x2':np.concatenate([np.random.normal(30,2,1000),np.random.normal(50,2,25)]),
})
np.random.normal

scaler = preprocessing.RobustScaler() robust_df = scaler.fit_transform(x)
robust_df = pd.DataFrame(robust_df,columns=['x1','x2'])

scaler = preprocessing.StandardScaler() standard_df = scaler.fit_transform(x)
standard_df = pd.DataFrame(standard_df,columns=['x1','x2'])

scaler = preprocessing.MinMaxScaler() minmax_df = scaler.fit_transform(x)
minmax_df = pd.DataFrame(minmax_df,columns=['x1','x2'])

fig, (ax1,ax2,ax3,ax4) = plt.subplots(ncols=4,figsize=(20,5)) ax1.set_title('Before Scaling')

sns.kdeplot(x['x1'],ax=ax1,color='r'),sns.kdeplot(x['x2'],ax=ax1,color='b')
ax2.set_title('After Robust Scaling')

sns.kdeplot(robust_df['x1'],ax=ax2,color='red')
sns.kdeplot(robust_df['x2'],ax=ax2,color='blue') ax3.set_title('After Standard Scaling')

sns.kdeplot(standard_df['x1'],ax=ax3,color='black')
sns.kdeplot(standard_df['x2'],ax=ax3,color='g')
ax4.set_title('After Min-Max Scaling')

sns.kdeplot(minmax_df['x1'],ax=ax4,color='black')
sns.kdeplot(minmax_df['x2'],ax=ax4,color='g')
```

```
plt.show()
```

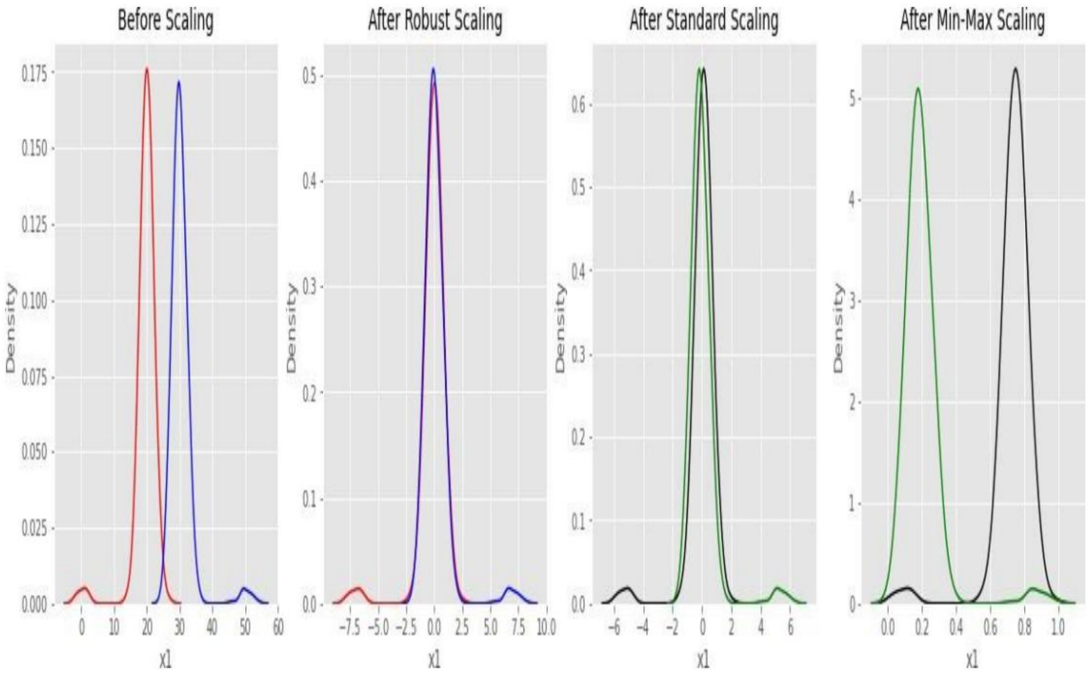


Fig 4.12: Robust Scaler

```
#Applying and Checking Accuracy and Precision of different Classification Models
from sklearn.naive_bayes import GaussianNB
from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
from sklearn.metrics import roc_curve
from sklearn.metrics import roc_auc_score
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import cross_val_score from sklearn.model_selection
import train_test_split from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
import xgboost as xgb
from sklearn.svm import SVC
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import precision_score

models = {
'Logistic_Regression':LogisticRegression(),
'Random_Forest':RandomForestClassifier(),
'XGBoost':xgb.XGBClassifier(), 'SVM':SVC(kernel = 'rbf'),
'KNN':KNeighborsClassifier(n_neighbors = 10),
'Naive_Bayes':GaussianNB()
}

for i in models:
obj = models[i]
obj.fit(train_X,train_Y)
obj_pred = obj.predict(test_X)
accuracy = accuracy_score(test_Y,obj_pred)
precision = precision_score(test_Y,obj_pred,zero_division=1)
print('Accuracy of '+i+': ',accuracy)
print('Precision of '+i+': ',precision)
# Reading the train.csv by removing the last column
# since it's an empty column
DATA_Path = "C:/Users/saifm/Downloads/Training data.csv"
data = pd.read_csv(DATA_Path).dropna(axis=1)
# Checking whether the dataset is balanced or not
disease_counts = data["prognosis"].value_counts()
temp_df = pd.DataFrame({
  "Disease": disease_counts.index,
  "Counts": disease_counts.values
})
plt.figure(figsize = (18,8))
sns.barplot(x = "Disease", y = "Counts", data = temp_df)
plt.xticks(rotation=90)
plt.show()
```

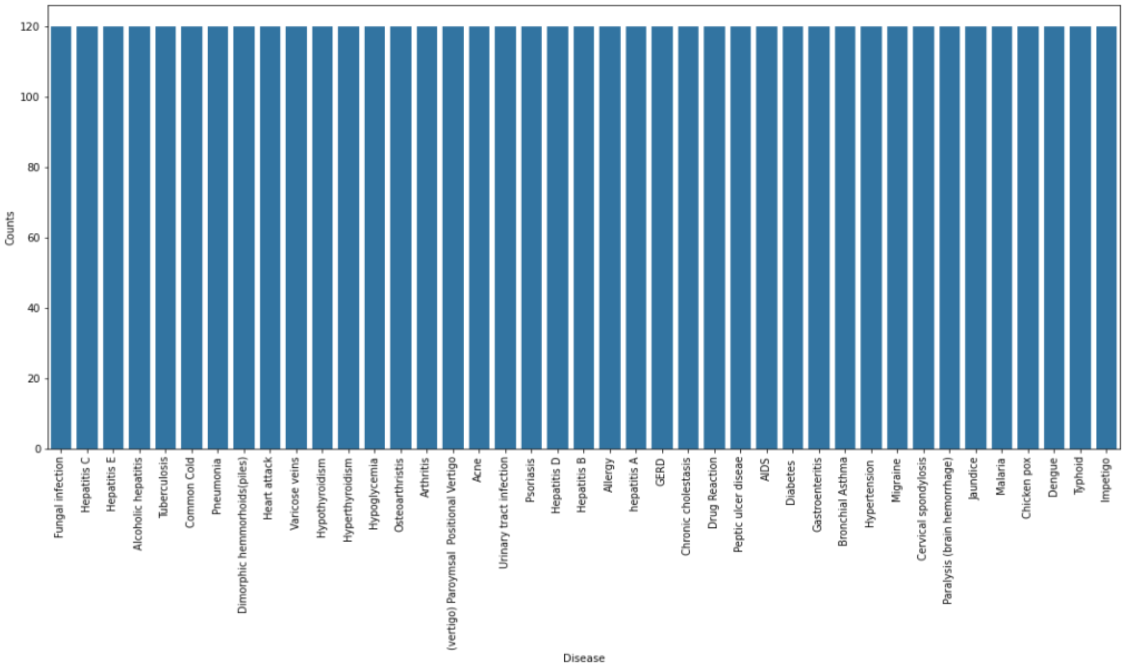


Fig 4.13: Disease – Counts Graph

```
# Encoding the target value into numerical value using LabelEncoder
encoder = LabelEncoder().data["prognosis"] =
encoder.fit_transform(data["prognosis"])
```

Splitting the data for training and testing the model

```
X = data.iloc[:,:-1]
y = data.iloc[:, -1]
X_train, X_test, y_train, y_test =train_test_split(
 X, y, test_size = 0.2, random_state = 24)
print(f"Train: {X_train.shape}, {y_train.shape}")
print(f"Test: {X_test.shape}, {y_test.shape}")
```

Model Building using Voting Classifier for model selection

```
# Defining scoring metric for k-fold cross validation
```

```
def cv_scoring(estimator, X_train, y_train):
 return accuracy_score(y_train, estimator.predict(X_train))
# Initializing Models
models = {
 "SVC":SVC(),
 "Gaussian NB":GaussianNB(),
 "Random Forest":RandomForestClassifier(random_state=18)
}
# Producing cross validation score for the models
for model_name in models:
 model = models[model_name]
 scores = cross_val_score(model, X_train, y_train, cv = 10,
 n_jobs = -1,
 scoring = cv_scoring)
 print("=="*30)
 print(model_name)
 print(f"Scores: {scores}")
 print(f"Mean Score: {np.mean(scores)}")
============================================================
SVC
Scores: [1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Mean Score: 1.0
============================================================
Gaussian NB
Scores: [1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Mean Score: 1.0
============================================================
Random Forest
Scores: [1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Mean Score: 1.0
```

Building robust classifier by combination of all models

```
# Training and testing SVM Classifier
svm_model = SVC()
svm_model.fit(X_train, y_train)
preds = svm_model.predict(X_test)

print(f"Accuracy on train data by SVM Classifier\
: {accuracy_score(y_train, svm_model.predict(X_train))*100}")

print(f"Accuracy on test data by SVM Classifier\
: {accuracy_score(y_test, preds)*100}")
cf_matrix = confusion_matrix(y_test, preds) plt.figure(figsize=(12,8))
sns.heatmap(cf_matrix, annot=True)
plt.title("Confusion Matrix for SVM Classifier on Test Data")
plt.show()
# Training and testing Naive Bayes Classifier
nb_model = GaussianNB()
nb_model.fit(X_train, y_train)
preds = nb_model.predict(X_test)
print(f"Accuracy on train data by Naive Bayes Classifier\
: {accuracy_score(y_train, nb_model.predict(X_train))*100}")

print(f"Accuracy on test data by Naive Bayes Classifier
: {accuracy_score(y_test, preds)*100}")
cf_matrix = confusion_matrix(y_test, preds) plt.figure(figsize=(12,8))
sns.heatmap(cf_matrix, annot=True)
plt.title("Confusion Matrix for Naive Bayes Classifier on Test Data")
plt.show()

models = {
 "SVC":SVC(),
 "Gaussian NB":GaussianNB(),
 "Random Forest":RandomForestClassifier(random_state=18)
```

```
# Training and testing Random Forest Classifier
rf_model = RandomForestClassifier(random_state=18) rf_model.fit(X_train, y_train)
preds = rf_model.predict(X_test)
print(f"Accuracy on train data by Random Forest Classifier\
: {accuracy_score(y_train, rf_model.predict(X_train))*100}")

print(f"Accuracy on test data by Random Forest Classifier\
: {accuracy_score(y_test, preds)*100}")
def cv_scoring(estimator, X_train, y_train):
 return accuracy_score(y_train, estimator.predict(X_train))
# Initializing Models
models = {
 "SVC":SVC(),
 "Gaussian NB":GaussianNB(),
 "Random Forest":RandomForestClassifier(random_state=18)
}
rf_model = RandomForestClassifier(random_state=18) rf_model.fit(X_train, y_train)
preds = rf_model.predict(X_test)
print(f"Accuracy on train data by Random Forest Classifier\
: {accuracy_score(y_train, rf_model.predict(X_train))*100}")

cf_matrix = confusion_matrix(y_test, preds) plt.figure(figsize=(12,8))
sns.heatmap(cf_matrix, annot=True)
plt.title("Confusion Matrix for Random Forest Classifier on Test Data")
plt.show()
```

Output-

```
Accuracy on train data by SVM Classifier: 100.0
Accuracy on test data by SVM Classifier: 100.0
```

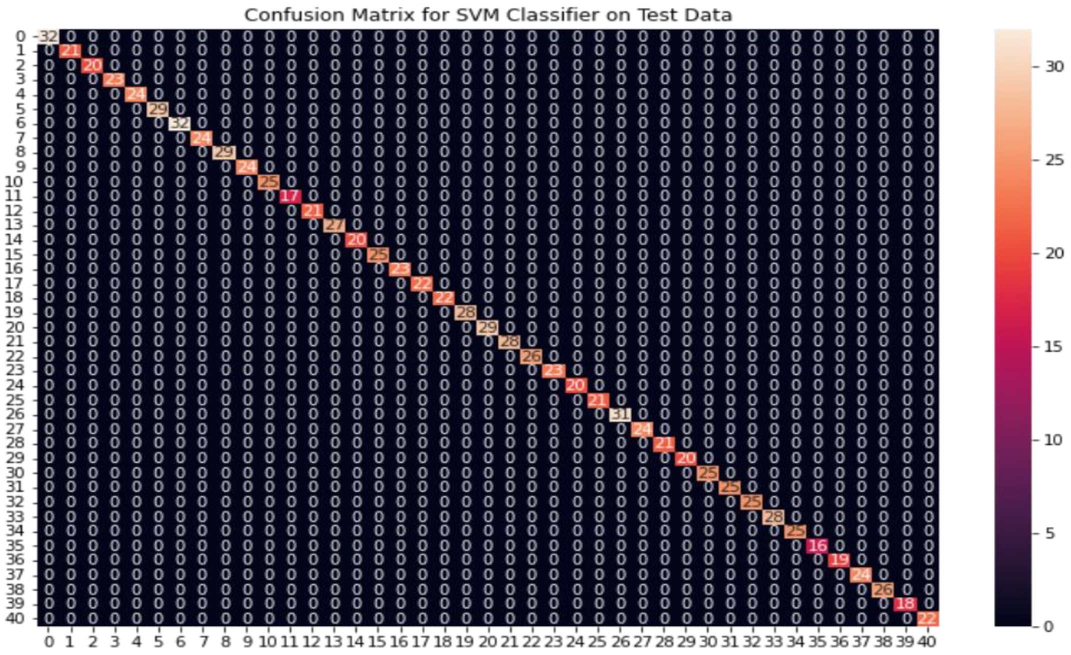


Fig 4.14: SVM Classifier

Output-

Accuracy on train data by Naïve Bayes Classifier: 94.0

Accuracy on test data by Naïve Bayes Classifier: 94.0

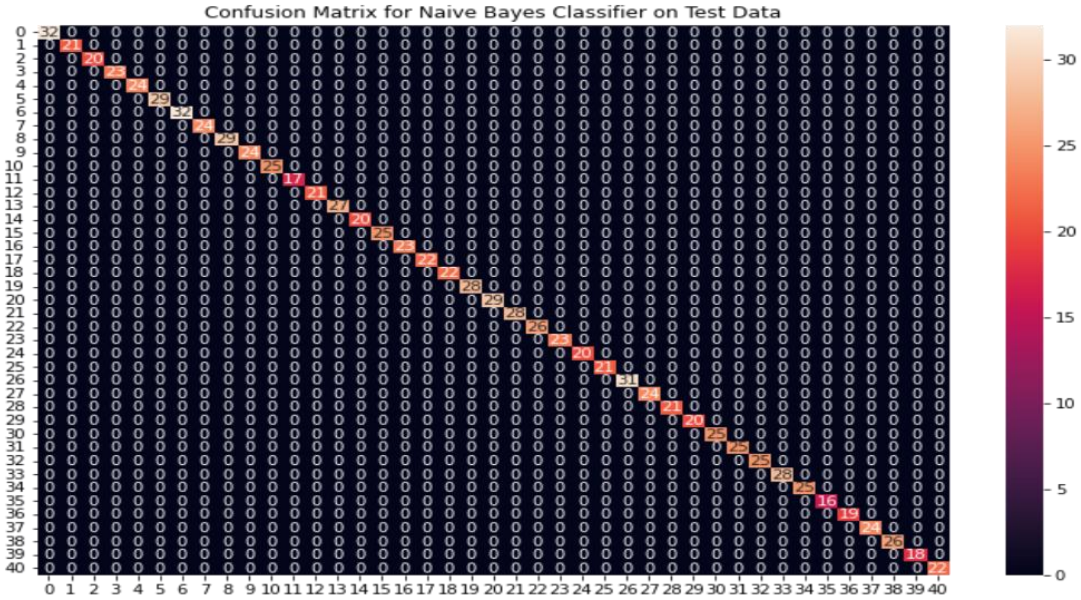
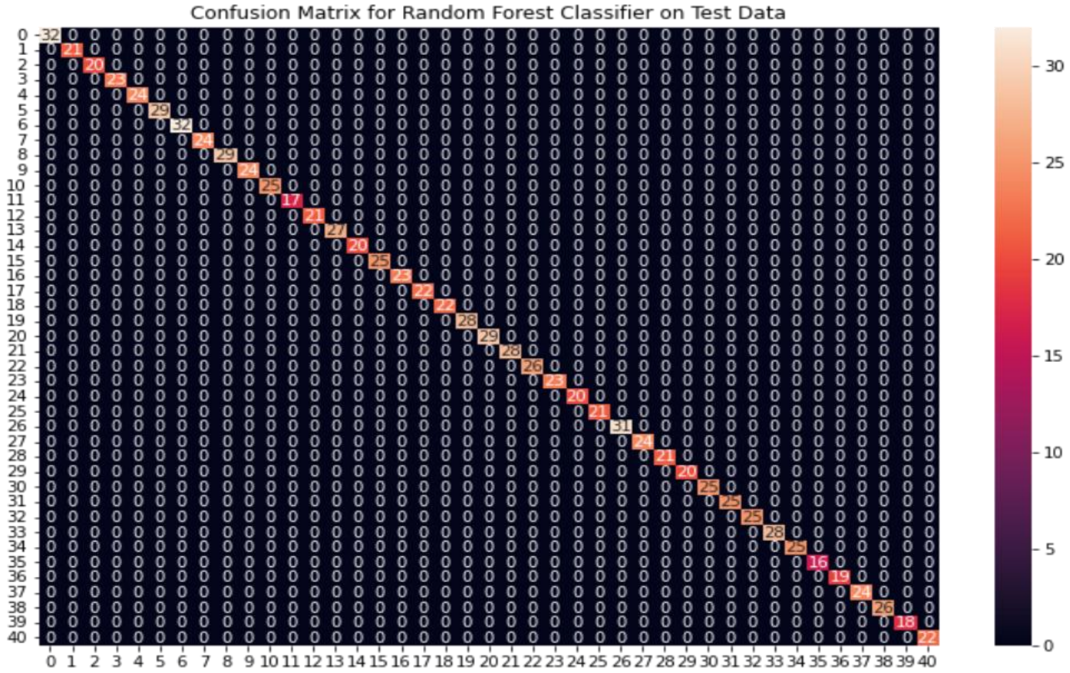


Fig 4.15: NB Classifier

Output-

Accuracy on train data by Random Forest Classifier: 94.0

Accuracy on test data by Random Forest Classifier: 94.0



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
0	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	31	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	25	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	25	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	25	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	25	0	0	0	0	0	0
35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0
36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0
37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0	0
38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	26	0	0
39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	18	0
40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22

Fig 4.16: RF Classifier

Fitting the model on whole data and validation on the test dataset

```
# Training the models on whole data
final_svm_model = SVC()
final_nb_model = GaussianNB()
final_rf_model = RandomForestClassifier(random_state=18)
final_svm_model.fit(X_train, y_train)
final_nb_model.fit(X_train, y_train)
final_rf_model.fit(X_train, y_train)
```

```
# Reading the test data

test_data = pd.read_csv("C:/Users/saifm/Downloads/Testing
data.csv").dropna(axis=1)

test_X = test_data.iloc[:, :-1]
test_Y = encoder.transform(test_data.iloc[:, -1])

# Making prediction by take mode of predictions # made by all the classifiers
svm_preds = final_svm_model.predict(test_X) nb_preds =
final_nb_model.predict(test_X) rf_preds = final_rf_model.predict(test_X)

final_preds = [mode([i,j,k])[0][0] for i,j,
k in zip(svm_preds, nb_preds, rf_preds)]

print(f"Accuracy on Test dataset by the combined model\
: {accuracy_score(test_Y, final_preds)*100}")

cf_matrix = confusion_matrix(test_Y, final_preds) plt.figure(figsize=(12,8))

sns.heatmap(cf_matrix, annot = True)
plt.title("Confusion Matrix for Combined Model on Test Dataset")
plt.show()

# creating input data for the models
input_data = [0] * len(data_dict["symptom_index"])
for symptom in symptoms:
index = data_dict["symptom_index"][symptom] input_data[index] = 1
# reshaping the input data and converting it # into suitable format for model predictions
input_data = np.array(input_data).reshape(1,-1)
```

Output-

Accuracy on train data by Combined Model Data

Accuracy on test data by Combined Model Data

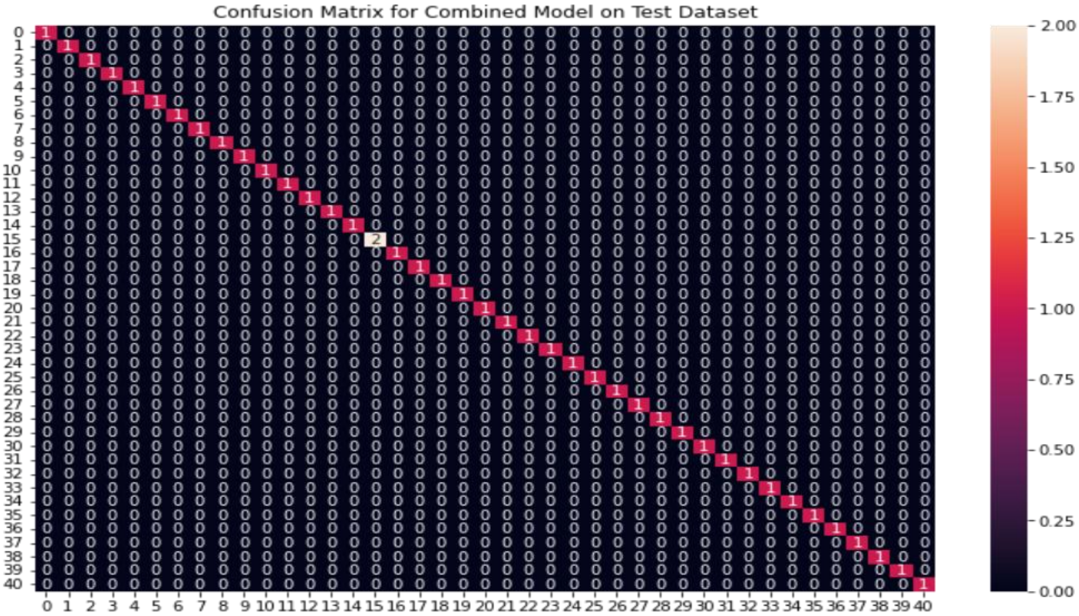


Fig 4.17: Confusion matrix heatmap

Creating a function that can symptoms as inputs and generate prediction

```
warnings.filterwarnings("ignore", category=UserWarning) symptoms =
X_train.columns.values

# Creating a symptom index dictionary to encode the # input symptoms into
numerical form
symptom_index = {}
for index, value in enumerate(symptoms):
symptom_index = ” ”.join([i.capitalize() for i in value.split(“_”)])
symptom_index[symptom] = index

data_dict = {
"symptom_index":symptom_index,
"predictions_classes":encoder.classes_
}

# Defining the Function
# Input: string containing symptoms separated by commas # Output: Generated
predictions by models
def predictDisease(symptoms):
symptoms = symptoms.split(",")

# creating input data for the models
input_data = [0] * len(data_dict["symptom_index"])
for symptom in symptoms:
index = data_dict["symptom_index"][symptom] input_data[index] = 1
# reshaping the input data and converting it # into suitable format for model predictions
input_data = np.array(input_data).reshape(1,-1)

# generating individual outputs
rf_prediction =
data_dict["predictions_classes"][final_rf_model.predict(input_dnb_prediction
=data_dict["predictions_classes"][final_nb_model.predict(input_d_svm_prediction
=data_dict["predictions_classes"][final_svm_model.predict(input)
# making final prediction by taking mode of all predictions
final_prediction = mode([rf_prediction, nb_prediction, svm_prediction])[0][0]
predictions = {
"rf_model_prediction": rf_prediction,
"naive_bayes_prediction": nb_prediction, "svm_model_prediction": svm_prediction,
"final_prediction":final_prediction
}
return predictions
# Testing the function
print(predictDisease("Itching,Skin Rash,Nodal Skin Eruptions"))
```

CHAPTER 5

RESULT DISCUSSION

5.1 Evaluation Matrices:

To Evaluate the Success of the Proposed Model, Several Metrices are considered:

Accuracy refers to the ratio of correctly predicted instances (both true positives and true negatives) to the total number of predictions made. While accuracy offers a general sense of performance, it may not fully reflect the model's reliability in imbalanced classification scenarios**.**

$$\text{Accuracy} = \frac{TP+TN}{FP+FN+TP+TN}$$

Precision on the other hand, measures the proportion of true positive predictions out of all positive predictions. This metric is particularly important in healthcare applications, where false positives may lead to incorrect treatments or unnecessary anxiety for patients.

$$\text{Precision} = \frac{TP}{FP + TP}$$

5.2 Comparative Model Performance

The Table below summarizes the Accuracy & Precision of Each Model:

Model	Accuracy (%)	Precision (%)
Logistic Regression	83.19	83.19
Random Forest	88.04	85.71
SVM	86.96	85.45
KNN	85.87	89.58
Naive Bayes	88.04	86.36

Table 4.66 Model Accuracy & Precision

CHAPTER 6

CONCLUSION

6.1 Integration of Autonomic Model

- The model integrates an autonomic computing module within the AI/ML framework to enable intelligent self-management of data preprocessing tasks.
- This module automates the data filtration process, which is a crucial preliminary step to ensure that only high-quality, complete, and relevant data is fed into machine learning algorithms.

6.2 Importance of Accurate Data in Healthcare Analysis

- In the healthcare domain, data integrity is critical. Inaccurate or incomplete data can lead to flawed predictions, potentially causing severe real-world consequences.
- By emphasizing the accuracy and completeness of healthcare datasets, the model proactively addresses one of the most common issues in medical machine learning applications.

6.3 Use of the MAPE-K Loop for Autonomic Management

The autonomic module operates based on the MAPE-K loop—a feedback control loop consisting of:

- Monitoring: Continuously observing data input for quality.
- Analysis: Evaluating data for inconsistencies or missing values.
- Planning: Determining appropriate filtration or correction strategies.
- Execution: Applying preprocessing operations autonomously.
- Knowledge: Retaining patterns for future optimization.
- This loop significantly reduces human intervention and promotes system self-sufficiency, aligning with the vision of autonomic computing systems.

6.4 Elevating System Maturity Toward Autonomicity

- The implementation of the autonomic module represents a step toward achieving autonomicity, wherein the AI/ML system can manage itself with minimal external input.
- This advancement increases the system's maturity and reliability, particularly in real-time and high-stakes environments like healthcare diagnostics.

CHAPTER 7

FUTURE SCOPE

Developing an AI/ML-Based Healthcare Prediction Model: Methodology

7.1) INCORPORATING AI TOOLS AND SYSTEM:

Approach: The strategy involves employing AI wrappers as intermediaries to seamlessly incorporate various AI tools and systems into the workflow. These wrappers serve as interfaces, abstracting the intricacies of interacting with different AI platforms to ensure smooth execution of machine learning tasks.

Method: Implement robust monitoring mechanisms to track the progress of AI tasks in real-time. By deploying advanced algorithms, we can proactively detect anomalies and deviations from expected behaviour. This enables us to intervene before tasks fail or fall outside predefined criteria. Additionally, incorporate mechanisms for the graceful termination of tasks to optimize resource utilization and minimize disruptions. The pivotal role of AI in real-time analytics of intricate healthcare data is instrumental in facilitating early disease prevention, refining diagnostic accuracy, and customizing personalized treatment strategies [2].

Implementation: Develop a flexible and adaptive system capable of interpreting natural language commands or configurations to determine the most suitable AI platforms for execution. This may involve leveraging natural language processing (NLP) techniques to extract relevant information from user input and map it to corresponding AI tools and systems. Additionally, consider integrating machine learning models trained on historical usage patterns to recommend optimal configurations based on specific task requirements and constraints. Ensure seamless integration with existing infrastructure and technologies to facilitate easy deployment and scalability.

7.2) SETTING UP INFRASTRUCTURE AND CONFIGURATION:

Infrastructure: Structure Establishing a master worker knot is a pivotal step, particularly in the environment of pall or distributed calculating systems. still, it's essential to design the structure with scalability and adaptability in mind. Factors similar as fault forbearance, cargo balancing, and data redundancy should be considered to ensure the system can handle the computational demands of AI/ ML tasks effectively. Technologies like containerization, similar as Docker or Kubernetes, can be explored to simplify deployment and operation across distributed surroundings.

Management: Operation piecemeal from registering bumps in the computing resource depository, effective operation processes are necessary to oversee resource application and performance. Access programs should be precisely defined and executed to cover sensitive data and coffers, with monitoring and waking mechanisms in place to track operation criteria and identify implicit issues. robotization workflows can streamline resource provisioning, configuration operation, and scaling, reducing homemade outflow and icing effective operations.

Integration: flawless integration with being structure and services is pivotal to influence investments and capabilities effectively. This includes integrating with IAM systems for centralized authentication and authorization, as well as logging and auditing systems for compliance and governance. Interoperability with data storehouse and processing systems should also be established to grease data exchange and analytics. espousing structure as Code principles can automate structure provisioning and configuration, enhancing reproducibility and thickness across surroundings.

7.3) INTEGRATING AUTONOMIC COMPUTING TECHNIQUE:

Architecture: The integration of autonomic computing ways involves breaking down the system armature into modular factors, each addressing specific aspects similar as machine literacy processes and the activation of AI ways[9]. This segmentation allows for better operation and optimization of individual factors, easing more effective resource allocation and task prosecution.

Management: To effectively integrate autonomic computing ways, the autonomic director plays a central part in overseeing tone- configuration, optimization, mending, and data securing within the AI/ ML armature. It stoutly adjusts system parameters and configurations grounded on changing conditions, icing optimal performance and adaptability. also, the autonomic director tools mechanisms for data integrity and security, securing sensitive information throughout the AI/ ML workflow[6].

7.4) DATA FILTRATION MODULE:

Data- Driven Learning Central to our path is the use of engine literacy algorithms trained on strictly curated datasets sourced from secure databases. These datasets form the foundation for constructing a robust prophetic model. Optimizing Resource Efficiency, we prioritize resource effectiveness by enforcing rigid protocols, potentially involving ways like model pruning or containerization, all while icing ethical considerations are consummate. Ethical AI Integration Ethical principles are integrated throughout the evolution process, icing compliance with data sequestration regulations and mollifying impulses through the use of loveliness criteria and explainability ways. Nonstop enhancement Model evolution and resource configuration are viewed as a nonstop circle, with perceptivity from model interpretation informing adaptations to resource allocation strategies. This iterative process ensures ongoing optimization and ethical adherence.

7.5) MODEL DEVELOPMENT AND CONFIGURATION

Foundational literacy A Data - Driven Approach using Secure Data We will use engine literacy algorithms to authority our model by employing data from well- maintained and secure databases. Data Integrity First Careful data election and pre-processing are pivotal. We ensure the model trains on high- quality information that adheres to ethical data governance principles, promoting loveliness and translucency [8].

Optimizing interpretation and Efficiency Resource Optimization Strategies enforcing resource optimization programs is crucial. ways like model pruning, quantization, or containerization will be considered grounded on the model’s special requirements and accessible coffers. This ensures our model operates efficiently and bring- effectively.

Ethical AI A gut reflection Building Fairness and translucency AI Ethics are intermediary throughout the evolution lifecycle[21]. We work loveliness criteria and explainability ways during model training. This helps us alleviate implicit impulses and guarantee adherence to data sequestration regulations, furthering responsible AI evolution [14].

Nonstop enhancement Through Feedback a Perpetual Loop A feedback circle will be established between model evolution and resource configuration. The model’s interpretation, involving delicacy and resource consumption, will be continuously covered. This perceptivity will be exercised to upgrade resource allocation strategies and, if necessary, detector model retraining. This nonstop enhancement process guarantees the model remains optimized and operates within ethical boundaries.

7.6) MODEL MONITORING AND UPDATE:

To ensure harmonious interpretation and address implicit effects, we will establish a complete model monitoring and update protocol, which includes:

Nonstop Monitoring: The model's interpretation will be regularly covered at predefined intervals, encompassing criteria similar as delicacy, perfection, recall, and resource application.

Benchmark evaluation: Evaluation interpretation data will be assimilated against established marks, deduced from literal data or assiduity norms workable to analogous models. In our pursuit of Ethical AI, we reflect deeply on fairness and transparency, recognizing their crucial role in the development lifecycle. Throughout our process, we prioritize fairness criteria and incorporate explain ability techniques during model training. This proactive approach not only helps mitigate implicit biases but also ensures compliance with data privacy regulations, thus fostering the responsible evolution of AI [4].

Visionary Anomaly Discovery: Sophisticated anomaly discovery ways will be stationed to identify significant diversions or trends in the model’s interpretation beyond anticipated parameters.

Root Cause Analytics and remedial conduct: In the event of detected diversions, a thorough root cause dissection will be conducted to identify underpinning effects, which may carry data drift, conception drift, or environmental changes. latterly, applicable remedial measures will be enforced, similar as retraining the model with streamlined data or conforming hyperparameters.

7.7) CONTINOUS IMPROVEMENT AND ADAPTATION:

Maintaining Model Efficacy: A gut principle is the nonstop improvement and conservation of the model’s interpretation, especially in the dynamic demesne of healthcare data dissection.

Regular Model Reassessment: We'll establish a program for regular modeller-evaluation. This may involve periodically retraining the model with fresh healthcare data to regard for evolving trends and patterns in the medical field.

Emphasis on Generalization: During retraining, we will prioritize ways that enhance the model's generalizability. This ensures the model can effectively acclimatize to new data points and scripts, pivotal for robust healthcare operations. Incorporating Domain Expertise: Collaboration with healthcare professionals will be vital throughout the process. Their sphere knowledge can be inestimable in relating applicable datasets, opting applicable evaluation criteria special to healthcare tasks, and interpreting the model’s labours within the medical environment [16].

7.8) QUALITY ASSURANCE AND OPTIMIZATION:

To guarantee the influence of the model in analysing healthcare dossier, we stress a strong approach that includes two together control of product quality and performance growth [15]. This contains:

Strict control of product quality: We implement strict control of product quality measures, containing inclusive unit experiment, unification testing and dossier confirmation contracts health facts.

Performance Optimization Techniques: Continuous conduct growth efforts are fault-finding. We will use methods in the way that hyperparameter tuning, model trimming and conceivably allied learning approaches to develop model adeptness and scalability. This ensures that the model can process increasing capacities of healthcare dossier and adapt to progressing flows in a up-to-the-minute manner.

REFERENCES

1. **Islam, M. R., Ahmed, M. U., Barua, S., & Begum, S**. (2022). A systematic review of explainable artificial intelligence in terms of different application domains and tasks. Applied Sciences, 12(3), 1353.

2. **Saraswat, D., Bhattacharya, P., Verma, A., Prasad, V. K., Tanwar, S., Sharma, G., & Sharma, R**. (2022). Explainable AI for healthcare 5.0: opportunities and challenges. IEEE Access, 10, 84486-84517.

3. **Razaque, A., Amsaad, F., Khan, M. J., Hariri, S., Chen, S., Siting, C., & Ji, X**. (2019). Survey: Cybersecurity vulnerabilities, attacks and solutions in the medical domain. IEEE Access, 7, 168774-168797.

4. **Islam, S. R., Kwak, D., Kabir, M. H., Hossain, M., & Kwak, K. S.** (2015). The internet of things for health care: a comprehensive survey. IEEE access, 3, 678-708.

5. **Kruse, C. S., Frederick, B., Jacobson, T., & Monticone, D. K**. (2017). Cybersecurity in healthcare: A systematic review of modern threats and trends. Technology and Health Care, 25(1), 1-10.

6. **Dehraj, P., & Sharma, A.** (2021). A review on architecture and models for autonomic software systems. The Journal of Supercomputing, 77(1), 388-417.

7. **Yaqoob, T., Abbas, H., & Atiquzzaman, M**. (2019). Security vulnerabilities, attacks, countermeasures, and regulations of networked medical devices—A review. IEEE Communications Surveys & Tutorials, 21(4), 3723-3768.

8. **Nasiri, S., Sadoughi, F., Tadayon, M. H., & Dehnad, A.** (2019). Security requirements of internet of things-based healthcare system: a survey study. Acta Informatica Medica, 27(4), 253.

9. **Dehraj, P., & Sharma, A**. (2020). An approach to design and develop generic integrated architecture for autonomic software system. International Journal of System Assurance Engineering and Management, 11(3), 690-703.

10. **Raul, A., Patil, A., Raheja, P., & Sawant, R.** (2016, October). Knowledge discovery, analytics and prediction in healthcare using data mining and analytics. In 2016 2nd International Conference on Next Generation Computing Technologies (NGCT) (pp. 475-478). IEEE.

11. **Ta, V. D., Liu, C. M., & Nkabinde, G. W.** (2016, July). Big data stream computing in healthcare real-time analytics. In 2016 IEEE international conference on cloud computing and big data analytics (ICCCBDA) (pp. 37-42). IEEE.

12. **Bansal, A., Deshpande, A., Ghare, P., Dhikale, S., & Bodkhe, B**. (2014). Healthcare data analytics using dynamic slot allocation in Hadoop. International Journal of Recent Technology and Engineering, 3(5), 15-18.

13. **Raghupathi, W., & Raghupathi, V.** (2014). Big data analytics in healthcare: promise and potential. Health information science and systems, 2, 1-10.

14. **Dehraj, P., & Sharma, A**. (2020). A new software development paradigm for intelligent information systems. International Journal of Intelligent Information and Database Systems, 13(2-4), 356-375.

15. **Horn, P**. (2001). Autonomic computing: IBM's perspective on the State of information technology. IBM Corp.

16. **Kephart, J. O., & Chess, D. M.** (2003). The vision of autonomic computing. Computer, 36(1), 41-50.

17. **Khalid, A., Haye, M. A., Khan, M. J., & Shamail, S**. (2009, April). Survey of frameworks, architectures and techniques in autonomic computing. In 2009 fifth international conference on autonomic and autonomous systems (pp. 220-225). IEEE.

18. **Pena, J., Hinchey, M. G., Sterritt, R., Ruiz-Cortes, A., & Resinas, M.** (2006, September). A model-driven architecture approach for modeling, specifying and deploying policies in autonomous and autonomic systems. In 2006 2nd IEEE International Symposium on Dependable,Autonomic and Secure Computing (pp. 19-30). IEEE.

19. **Gautam, P., & Dehraj, P.** (2021, April). A Review Paper on Machine Learning Techniques and Its Applications in Health Care Sector. In International Conference on Ubiquitous Computing and Intelligent Information Systems (pp. 177-197). Singapore: Springer Nature Singapore.

20. **Kumar, M., & Sharma, A**. (2017). An integrated framework for software vulnerability detection, analytics and mitigation: an autonomic system. Sādhanā, 42, 1481-1493.

21**. Vaupel, J. W., & Yashin, A. I**. (1985). Heterogeneity's ruses: some surprising effects of selection on population dynamics. The American Statistician, 39(3), 176-185.

22**. Polaraju, K., & Prasad, D. D**. (2017). Prediction of heart disease using multiple linear regression model. International Journal of Engineering Development and Research Development, 5(4), 1419-1425.