

Healthcare Using Machine Learning

Major project report submitted in partial fulfilment of the requirement for the degree of Bachelor of Technology

In

Computer Science and Engineering

By

Alok Pratap Singh(181248)-

UNDER THE SUPERVISION OF

Dr. Rakesh Kanji



Department of Computer Science & Engineering and Information
Technology

**Jaypee University of Information Technology, Wagnaghat
173234, Himachal Pradesh, INDIA**

TABLE OF CONTENT

Content	Page No.
Declaration.....	iii
Certificate.....	iv
Acknowledgement.....	v
Abstract.....	vi
Chapter 1: Introduction	
1.1 Overview.....	1
1.2 Problem Statement.....	2
1.3 Objective.....	2
1.4 Motivation.....	3
1.5 Scope.....	5
1.6 Language Used.....	6
1.7 Technical Requirements.....	6
1.7.1 Hardware Requirements.....	7
1.7.2 Software Requirements.....	7
1.8 Deliverables.....	7
Chapter 2:	
Literature/Survey Review.....	8
Chapter 3: Major Project SDLC	
3.1 Feasibility Study.....	9
3.2 Case Diagram.....	10
3.3 DFD Diagram.....	12
3.4 State Transition Diagram.....	13
Chapter 4: System Development	
4.1 Technologies Used.....	14
4.2 Design of Problem Statement.....	15
4.3 Algorithm/Pseudo Code.....	18
4.4 Flow Graph.....	20

Chapter 5: Performance Analysis

5.1 Data Set.....	21
5.2 Data Set Features.....	21
5.2.1 Description of Data Set.....	21
5.3 Screenshots of the various stages of the project.....	22
5.4 Libraries and Requirements.....	33

Chapter 6: Results

6.1 Discussion on the result achieved.....	41
6.2 Application.....	43
6.3 Limitations.....	43
6.4 Conclusions and Future Scopes.....	44
References.....	45

DECLARATION

I the undersigned solemnly declare that the project report “ **Healthcare Using Machine Learning** ” that's able to predict the probabilities of getting a disease in organs like liver, kidney, heart, etc. relies on my very own work administered during the course of our study under the supervision of **Dr. Rakesh Kanji**.

I assert the statements made and conclusions drawn are an outcome of my Project work. I further certify that

- I. The work contained within the report is original and has been done by me under the overall supervision of my supervisor.
- II. The work has not been submitted to the other Institution for the other degree/diploma/certificate during this university or the other University of India or abroad.
- III. We've followed the rules provided by the university in writing the report.
- IV. Whenever we've used materials (data, theoretical analysis, and text) from other sources, we've given due credit to them within the text of the report and given their details within the references.

Alok Pratap Singh
181248



CERTIFICATE

This is to certify that **Mr. ALOK PRATAP SINGH** of **Jaypee University of Information Technology** has successfully completed the project work titled “**Healthcare Using Machine Learning**” as a partial fulfilment of the requirement for the degree of Bachelor of Technology.

This Project report is the record of authentic work carried out by him during the educational period from 18th August 2021 to May 2022.

All help received by him from various sources are duly acknowledged. No part of this report has been submitted elsewhere. He has worked under my guidance.

(Dr. Rakesh Kanji)
Assistant professor
Supervisor

ACKNOWLEDGEMENT

The success and final outcome of this project required lots of guidance and assistance from many of us and that I am extremely privileged to own got this right along the completion of my project. All that i've got done is merely because of such supervision and assistance and that i wouldn't forget to thank them. I've made efforts during this project. However, it might not be possible without the sort support and help of the many individuals and organisations. I might prefer to extend my sincereness because of all of them.

I thank my god for providing me with everything that I required in completing this project. I am highly indebted to my supervisor Dr. Rakesh Kanji for his guidance and constant supervision yet as for providing necessary information regarding the project and also for his support in completing the project.

I would prefer to express my gratitude towards my parents for his or her kind co-operations and encouragement which helped me within the completion of this project.

Last but not the least my thanks and appreciation also attend to those who have willingly helped me out with their abilities in finalising this project within the stipulated time frame.

Alok Pratap Singh

ABSTRACT

The advancements of computer technologies have generated an out of this world amount of knowledge and knowledge from numerous sources. Nowadays, the way of implementing health care is being changed by utilising the advantages of advancements in computer technologies. It's believed that engineering this amount of information can assist in developing predictive tools that may help physicians to diagnose and predict some debilitating life-threatening illness like disease.

Breast cancer may be a group of diseases during which cells in breast tissue change and divide uncontrolled, typically leading to a lump or mass. Most breast cancers begin within the lobules (milk glands) or within the ducts that connect the lobules to the nipple. Carcinoma typically has no symptoms when the tumour is little and most easily treated, which is why screening is vital for early detection. The foremost common physical sign could be a painless lump. Sometimes carcinoma spreads to underarm lymph nodes and causes a lump or swelling, even before the initial breast tumour is large enough to be felt. Less common signs and symptoms include breast pain or heaviness; persistent changes, like swelling, thickening, or redness of the skin; and nipple changes, like spontaneous discharge, scaliness, or retraction. Any persistent change within the breast should be evaluated by a physician.

Diabetes could be a chronic disease that happens either when the pancreas doesn't produce enough insulin or when the body cannot effectively use the insulin it produces. Insulin could be a hormone that regulates glucose. Hyperglycaemia, or raised blood glucose, could be a common effect of uncontrolled diabetes and over time ends up in serious damage to several of the body's systems, especially the nerves and blood vessels.

Liver disease is any disturbance of liver function that causes illness. The liver is to blame for many critical functions within the body and may become diseased or injured, the loss of these functions can cause significant damage to the body. disease is additionally cited as hepatic disease. The liver is the largest solid organ within the body; and is additionally considered a gland because among its many functions, it makes and secretes bile. The liver is found within the upper right portion of the abdomen protected by the skeletal structure. It's two main lobes that are made of tiny lobules. The liver cells have two different sources of blood supply. The arteria hepatica supplies oxygen rich blood that's pumped from the guts, while the venous blood vessel supplies nutrients from the intestine and also the spleen.

Heart disease may be a kind of disease that affects the guts or blood vessels. The chance of certain heart diseases is also increased by smoking, high force per unit area, high cholesterol, unhealthy diet, lack of exercise, and obesity. The foremost common cardiovascular disease is arteria coronaria disease, which may result in hurting, heart attacks, or stroke. Other heart diseases include congestive heart condition, regular recurrence problems, congenital

cardiopathy (heart disease at birth), and endocarditis (inflamed inner layer of the heart). Also called disorder.

Kidney disease means our kidneys are damaged and can't filter blood the way they ought to. Patients are at greater risk for nephrosis if they need diabetes or high force per unit area. If you experience nephropathy, treatments include kidney transplant or dialysis. Other kidney problems include acute kidney injury, kidney cysts, kidney stones, and kidney infections. Nephrosis can cause dangerous levels of fluid, electrolytes and wastes to make up in your body. In the early stages of chronic nephrosis, you would possibly have few signs or symptoms. you may not realize that you just have uropathy until the condition is advanced. Treatment for chronic nephropathy focuses on slowing the progression of kidney damage, usually by controlling the cause. But, even controlling the cause won't keep kidney damage from progressing. Chronic renal disorder can attain end-stage renal disorder, which is fatal without artificial filtering (dialysis) or a kidney transplant.

From the various machine learning techniques, compared widely used three algorithms namely logistic regression, decision trees and k-nearest neighbour (kNN) algorithms to predict and evaluate their performance in terms of accuracy and Support Vector Machine (SVM) for multiclass classification of dataset.

This study has represented the intuition of the way to predict disease and highlighted a way to apply the logistic regression, decision trees and kNN as a tool for the classification. For this, a carcinoma Wisconsin data set, Pima Indians Diabetes dataset, Indian Liver Patients collected from North East of state, cardiopathy UCI Dataset, Chronic nephropathy Dataset of machine learning repository has been used from Kaggle.

CHAPTER 1

INTRODUCTION

1.1 Overview

The colossal headway of information innovation, framework combination strategies and programming improvement have delivered a substitution age of convoluted PC frameworks. Thus, these frameworks have introduced large difficulties to designing science. A fair illustration of those complicated frameworks is the medical services framework. As of late, there has been an expanded interest to utilise the headway of correspondence, information handling and AI advancements in medical services frameworks. Subsequently, numerous nations are changing the method of getting sorted out medical services frameworks towards a world medical services framework by standardising medical services in correspondence and structuring electronic medical records across the country.

The Electronic Health Record (EHR) is a centralised collection of electronic health information on a single patient or a group of patients. It's approved for use by medical care providers in an extremely secure country. General clinical records, anamnesis, patient assessments, research centre outcomes, patient medicines, radiology pictures, hypersensitivities, inoculation status, and a few other useful data for assessment are all common components of health records. As a result, this valuable information is likely to aid analysts in analysing and detecting ailments using PC methodologies. The adoption of EHRs may aid in improving the standard of care, lowering the cost of legacy frameworks, and improving record mobility.

Along these lines, our point inside the momentum work is to explore the parts of using the storehouse of wellbeing information for the beneficial thing about people by utilising AI. Our thought is to propose a programmed technique for diagnosing illnesses upheld recently and put away information and information. Notwithstanding, there are numerous issues related with successfully utilising this recently gotten patient information, which could make any electronic device less effective. Some of these issues are: the trouble of enormous elements or ascribes and the method for picking the chief helpful ones, the question of missing qualities and the method for handling that, the issue of extricating precise demonstrative markers that might anticipate the initial beginning of the infection. This venture will attempt to research some of these issues and propose a prescient apparatus for infection finding, upheld the potential and furthermore the force of computerised innovations and accordingly the past patients or information. Be that as it may, the extent of the proposition is restrictive to the issues laid out above, and does exclude other similarly significant issues like protection and security.

In this examination, Kaggle dataset is utilised as our information hotspots for creating programmed AI devices, to give a helpful prescient technique to diagnosing illness. The review used important datasets that are publically accessible for research purposes in order to keep

track of the examinations for this work. It was decided that the instrument developed supported decision tree calculating techniques, and that approving on this dataset is regularly applied to truly clinical problems.

1.2 Problem Statement

To fabricate a grouping technique to anticipate whether or not an individual is covered with that singular illness upheld the given preparing informational index. Our point during this task work is to examine the parts of using the storehouse of wellbeing information for the beneficial thing about people by applying distinctive AI Algorithms. to search out a pristine gratitude to use patient's chronicles, wellbeing data, and information bases for distinguishing and diagnosing illnesses, likewise give prescient instruments as clinical experts. This exploration is expected to decide an application which will help doctors in diagnosing illnesses and characterising patients in valuable examples upheld various properties.

1.3 Objective

The points of this task work are:

- A. To observe a substitution on account of use patient's chronicles, wellbeing data, and data sets for identifying and diagnosing illnesses, likewise give prescient apparatuses as clinical experts. This examination is anticipated to discover an application that might help doctors in diagnosing illnesses and arranging patients in valuable examples upheld various properties, and the way AI procedures will be successful to spot such examples. This could help in finding the beginning stage of the sickness, therapy plans and recognizable proof of infection stages.
- B. To adapt to a larger than usual number of elements and traits inside the dataset, and distinguish the significance of certain elements over others. In any case, a larger than usual number of highlights can begin the scourge of dimensionality, likewise could deliver an AI calculation or method restricted as far as exactness, explicitness and accuracy.
- C. To resolve an essential issue related with making up a GUI joined with an AI calculation, that might assume a vital part in deciding the adequacy and basic utilisation accomplished by planning advances and AI calculations.

Consequently, this undertaking proposes new strategies for exploring AI procedures and creates AI instruments for giving prescient apparatuses to conclusion. This work imagines that the results of this exploration as far as a shrewd forecast application , with simple use plan of GUI , getting together with choice tree algorithmic strategies, extricating huge 'highlight

choice' and utilising AI based grouping can build up the precision with which sort of infection an individual is blasted by or a standard state of the brokenness might be recognized. Figure 1 shows our work process Architecture.

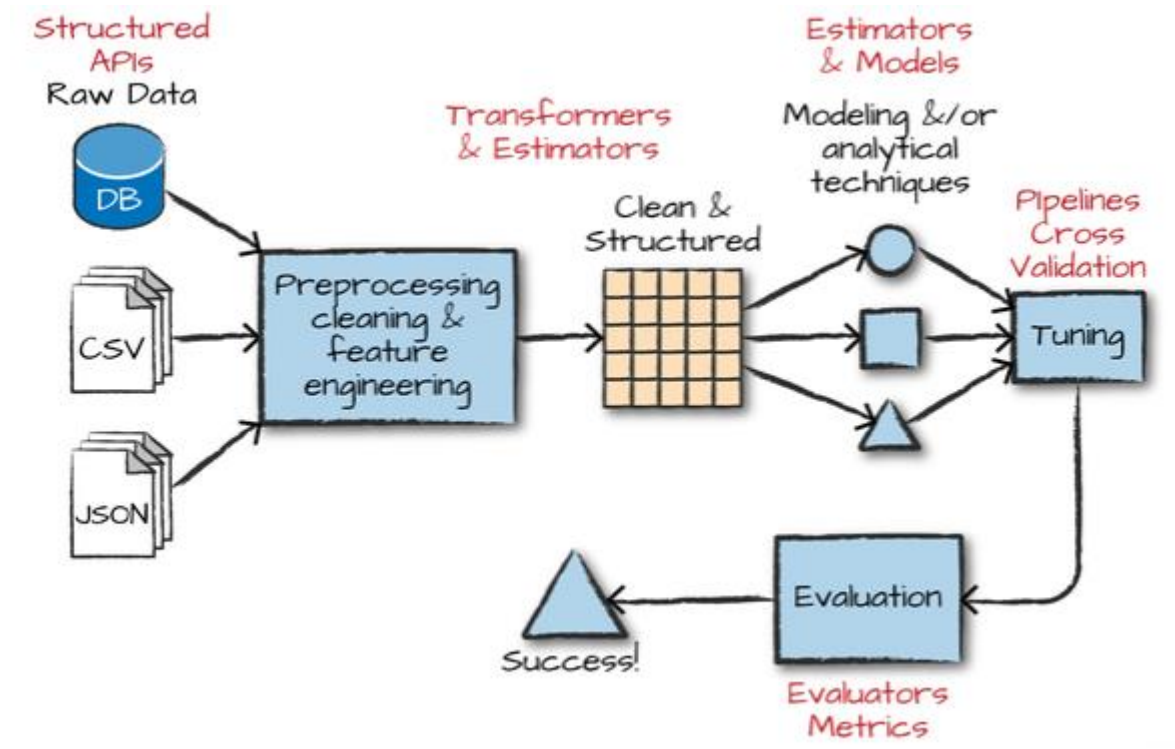


Figure 1: Workflow Architecture.

To close, the exploration destinations of this work is to use patient's accounts, wellbeing data, and information bases from assortments of data sets at stores for finding finishes paperwork for early heart, kidney, liver, carcinoma, diabetes illness determination with a coordinated canny forecast application comprising of incredible element choice, and learning based grouping. The exploration is anticipated to decide an instrument or an application that might help doctors in diagnosing infections. The point is to style a ML application that blends human aptitude in with the innovation knowledge to accomplish a determination that is more precise. This application might help doctors in higher intellectual cycles, for better and powerful treatment plans.

1.4 Motivation

Bosom malignant growth could be a significant issue for American ladies. The passing rate from this illness in this nation has been among the awesome inside the world for a considerable length of time. Carcinoma is the most run of the mill malignant growth in ladies, representing at least 26% of female disease. Today, one out of 11 ladies gets the sickness. There has been a fixed passing rate from carcinoma in this country for the past 40 years. Information from Great Britain, where carcinoma frequency is practically equivalent to thereto during this nation, show

an expansion in death rate inside the beyond couple of years. Carcinoma found in an extremely restricted stage has a 85% five-year endurance contrasted with 53% when organs are involved. Since most carcinoma today isn't restricted when originally seen by the clinician, a technique should be found to forestall ladies introducing themselves for assessment with their illness at a previous stage than is generally the situation. This implies, in a really common sense way, discovery of preclinical malignant growth in evidently "well" ladies, when the sickness is unsuspected by quiet or doctor just like that the case in mass screening.

Diabetes is a common chronic illness that poses a significant threat to human health. The characteristic of diabetes is that the glucose level is higher than normal, which is caused by insufficient insulin production or its hampered organic effects, or both. Diabetes can injure and break down changed tissues over time, particularly the eyes, kidneys, heart, veins, and nerves. Type 1 diabetes (T1D) and type 2 diabetes (T2D) are the two types of diabetes (T2D). Patients with type 1 diabetes are typically younger, as young as 30 years old. Expanded thirst and subsequent urination, as well as elevated glucose levels, are common clinical symptoms. Oral medicines alone will not suffice to treat this type of diabetes, and patients will also require insulin.

As per the WHO in Egypt constant illnesses are to be blamed for 78% of absolute deaths and within the following 10 years twenty lakhs of people will kick the bucket in light of ongoing illnesses. Liver illnesses likewise are accessible in the classification of ongoing illnesses. an outsized number of contaminations affect the liver which prompted the changed liver illnesses. The passings on account of liver illnesses have reached 208185 or 7.34% of complete demise in Egypt. One among the most common reasons for expanded liver illnesses in Egypt is corpulence, inward breath of unsafe gases, admission of tainted food, exorbitant utilisation of pickles and drugs, and liquor. The objective of this paper is to propose a Machine learning strategies upheld Classification of Liver issues for decreasing weight on specialists.

AI procedures are around us and are thought about and utilised for investigation for quite some time in information science applications. The main inspiration driving this exploration based task was to investigate the component choice strategies, information arrangement and handling behind the preparation models in AI. Subsequently this task is distributed with the inspiration to investigate behind the models, and further carry out Logistic Regression. During this venture we've fostered a model which arranges in the event that a patient will have cardiomyopathy in ten years or not and upholds different elements utilising calculated relapse. Thus, the main hypothesis of cardiovascular infections can support settling on choices on way of life changes in high danger patients and progressively decrease the difficulties, which may be a decent achievement inside the field of drug.

The pervasiveness of persistent nephrosis (CKD) as a huge public weakness is developing inside the older. This review is expected to survey CKD commonness and its connected danger factors in the old populace of Fars area. Strategies. During this cross sectional review a total 1190 older individuals were enlisted, and segment and clinical information were gotten. Information was investigated by SPSS, and P of yet 0.05 was thought of as measurably huge. Results. Commonness of CKD stages III–V was 27.5% inside the 60–69 years individuals,

36.5% inside the 70–79 years individuals, and 40% inside the ≥ 80 years associate. The pervasiveness of CKD expanded with maturing in the two men and young ladies. Female sexual orientation was the most grounded hazard factor for CKD. Ends. Commonness of CKD in old age is high in Southern Iran, which has turned into an essential neurotic state while it is forestalled or postponed in movement.

Fixing of infection might be a customary worry for the medical services specialists, and furthermore the errorless conclusion at the appropriate time for a patient is critical. As of late, by some high level determination strategies, the normal composed report might be produced with a further report supporting side effects. The different inquiries like "what are the reasons for influencing the disease?", "Which companion of people are impacted because of that individual disease?", "what is that the applicable treatment for a disease?", and so on may observe replies on carrying out AI techniques. Medical services information will be handled and after carried out with specific philosophies; it can give data which will be used in determination and therapy of illnesses all the more productively and precisely with better choosing and limiting the passing danger.

AI approaches could be used to deal with the massive amount of data. For the layout and qualification of data classes, grouping models are comparable. The arrangement techniques may end the treatment of both mathematical and downright features. Characterization could be a two-part process, with the first stage supporting some preparatory information and the second step giving the model an obscure tuple to aggregate into a class name.

In human existence, the characterization includes an extraordinary impact. The examination of different grouping strategies is non-minor and remembers an incredible reliance for the data set properties. Inside the insights local area, strategic relapse, choice tree and k-closest neighbour have gotten a regarded position for grouping issues.

The techniques for order utilised are the notable strategies. To focus on the above-examined issues, this report clarifies the use of three order AI calculations: strategic relapse characterization, choice tree grouping and closest neighbours arrangement to arrange individuals pruned by infection utilising the illness data set. The report clarifies personally about the readiness, preparing and testing of the information, bit by bit depiction of everything about procedures utilised, and a correlation of the precision of the techniques utilised in the forecast.

1.5 Scope

The extent of this undertaking is to make a characterization procedure to anticipate the kind of sickness upheld by the given preparing dataset. Either a person suffering from that disease or not, it can be a positive case or negative case ; if it is a positive case then what type of case it's?. What types of disease a person is suffering from?. So, at the first level of specialisation my model will predict that a person is suffering from disease or not, what type of case it's. If

the result is found to be positive then a treatment of that person will be a bit fast-track, a doctor will come and give him attention and they will treat their treatment on priority.

But let's say if the result is coming negative then what will happened is report of that particular patients it will go to the junior doctor, then that junior doctor with their own expertise they will identify okay weather the model has done correct prediction or not; mostly the model does correct prediction , but still we can't directly go ahead and move the human intervention.

If the model has made a correct prediction and the junior doctors agree that the person is not suffering from any disease or particular symptoms, they will release the patient. But again based on the reading and different tests, the doctor can conclude okay, the person might be suffering from the disease. Again the report of that patient will again be sent to the senior doctor and they will treat the patient, so this is how the mechanism is going to work.

1.6 Language Used

The venture is built on PyCharm which is an incorporated advancement climate (IDE) used in making by mental demonstrations, explicitly for the Python language and Anaconda which could be an appropriation of the Python and R programming dialects for logical registering (information science, AI applications, enormous scope handling, prescient investigation, and so on), that intends to work on bundle the board and sending.

This project only supported Python Programming Languages. Different python libraries like pandas, numpy, matplotlib, pyplot, seaborn, resample, KNNImputer, LabelEncoder and a number of other other python libraries are used.

1.7 Technical Requirements

1.7.1 Hardware Requirements

Device Specifications

Processor	Intel(R) Celeron(R) J4005 CPU @ 2.00GHz 2.00 GHz
Installed RAM	4.00 GB (3.85 GB usable)
Device ID	5D876BDF-30FA-45D6-9F96-2D96BF4830B5
System Type	64-bit working framework, x64-based processor

1.7.2 Software Requirements

- Internet Browser: Microsoft Internet Explorer, Mozilla, Google Chrome or later
- Operating System: Windows XP / Windows7/ Windows Vista/ Windows 10
- PyCharm Version 2021.1

- Anaconda Version 2020.11
- astropy 4.0.2
- beautifulsoup4 4.9.3
- ipython 7.19.0
- jupyterlab 2.2.6
- matplotlib 3.3.2
- notebook 6.1.4
- numpy 1.19.2
- pandas 1.1.3
- python 3.8.5
- scikit-learn 0.23.2
- scipy 1.5.2
- sqlalchemy 1.3.20
- statsmodels 0.12.0

1.8 Deliverables

Medical care utilising Machine Learning could be a venture idea that focuses on an astute and exact on account of anticipating the kind of infection. We've utilised calculated relapse calculations to mentor our dataset and to foresee sickness with more exactness. Here the machine is prepared to distinguish whether or not the individual has disappeared with that definite sickness, what style of case it upholds the client's feedback. So when a client enters information during a web application the data will be handled inside the backend (model) and furthermore the outcomes are shown on the screen. Our goal was to give society a proficient and exact method of AI which might be utilised in applications intending to perform sickness recognition. Further advancement will be finished by utilising picture handling of ultrasonic checking of pictures to foresee the infection with better exactness, which can not be perceived in biopsy reports. By joining both the outcomes, Our model will anticipate the infection with better precision.

CHAPTER 2

Literature/Survey Review

There has been loads of work done to analyse the discrete illnesses. The creators demonstrated a sufficient methodology and sureness to search out the illnesses undifferentiated by the work that has different datasets and calculations connected with the work that will be tiring since quite a while ago from the run viewpoint to achieve viable and better outcomes. The purpose of the report deciphers different methods of AI and furthermore the measurable qualities that has been advocated inside the last a long time for translation of thyroid illnesses with the arrangement to accomplish different possibilities and for different approaches. There are different calculations of AI irregular woodland, choice tree, gullible Bayes, SVM . The creators examined and analysed the four order models, specifically Naive Bayes, Decision Tree, Multilayer Perceptron and Radial Basis Function Network. The determination shows a groundbreaking exactness for all the characterization models. The decision Tree model surpasses the contrary grouping models. In this work 29 ascribes of dataset is recruited and implemented as a Feature Selection method for example Chi-Square, The datasets are being sifted by leading the unaided covered channels on the properties for transformation inside the persistent qualities into ostensible and henceforth decrease the 29 ascribes to 10 credits.

AI (ML) might be a division of computerised reasoning and is invaded inside the components of examination projects at developing advances. AI works with calculations to audit as a matter of fact without outstandingly being focused on . AI has been instigated by the info explosion that is associated with a growing computational ability, and traditionally the study of disease transmission is a convoluted mixed ongoing information science way to deal with lash the capacities of the tasteful information.

The indications dataset was imported from the UCI ML storehouse, where it contained side effects of numerous normal sicknesses. The framework utilised CNN and KNN as classification strategies to understand various illnesses forecast. In addition, the proposed arrangement was enhanced with more data that concerned the living propensities for the tried patient, which ended up being useful in understanding the measure of hazard connected to the expected illness. Dahiwade looked at the outcomes among KNN and CNN calculation as far as span and exactness. The exactness and timespan were 84.5% and 11.1 seconds, separately. \ The creators presumed that the proposed model scored higher as far as exactness, which is clarified by the fitness of the model to identify complex nonlinear connections inside the element space. In addition, CNN recognizes highlights with high significance that render better portrayals of the sickness, which empowers it to precisely foresee illnesses with high intricacy. This end is all around upheld and supported with exact perceptions and factual contentions. Regardless, the introduced models needed subtleties, for instance, Neural Networks boundaries like organisation size, design type, learning rate and back proliferation calculation, and so on moreover, the examination of the exhibitions is scarcely assessed as far as precision, which exposes the legitimacy of the introduced findings. Besides, the creators didn't think about the predisposition issue that was looked by the tried calculations. In representation, the fuse of more element factors could tremendously enhance the presentation measurements of failed to meet expectations calculations.

CHAPTER 3

Major Project Software Development Life Cycle

3.1 Feasibility Study

A Feasibility Study is the best gratitude to seeing the wellsprings of the patient stream to the Hospital. The Feasibility Study involves both a Market and Financial Analysis of the Project and is a vital viewpoint to be thought about one or the other while building up a substitution clinic or a substitution office in a current emergency clinic or growing the common emergency clinic.

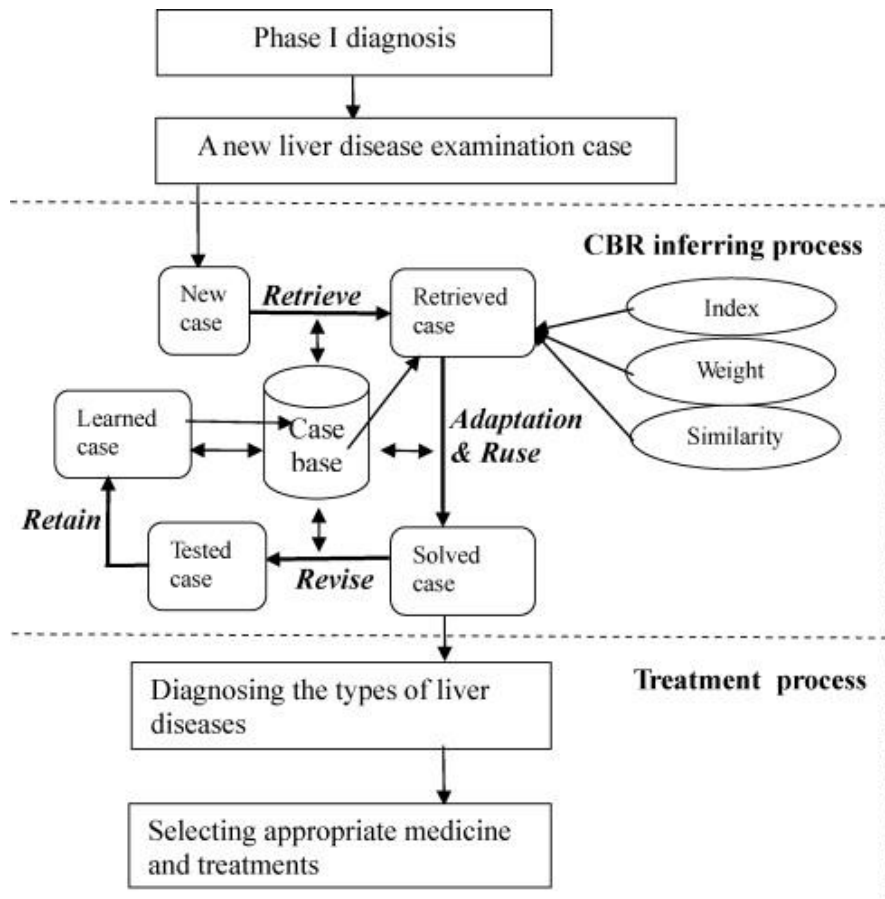
It is an overall practice to explore the extent of the task or the idea prior to making any more extended strides and here is the place where possibility to concentrate on comes into the picture. The review incorporates assessing the degree of involvement required, subjective and quantitative evaluation of the potential results, the range of abilities that is required, ID of the urgent variables, plausible danger and functional gaps.

The Benefits Of Conducting A Hospital Feasibility Study

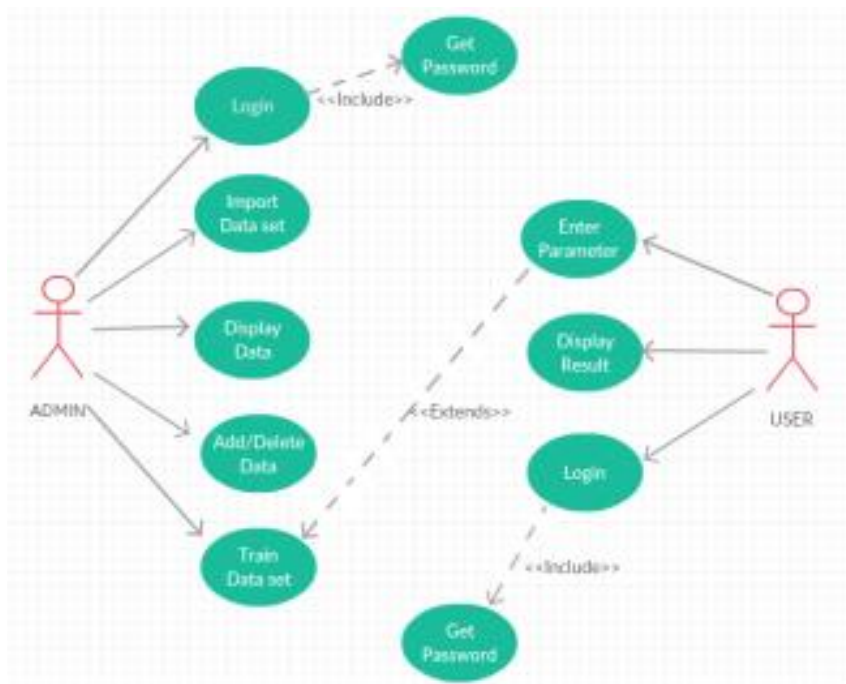
- Expanding the extension in order to detect the new chances.
- Upgrades the assets abilities while expecting the difficulties.
- Helps in reducing the likely dangers and tracking down arrangements.
- In particular, recognizes the consequence of the venture with the most extreme precision.

Each medical services project includes time, venture and assets and achievability study guarantees that the administration contains an unmistakable thought to take a position right inside the perfect opportunity. The high level investigation and definite situation of the more drawn out term market will consistently help in making the right strides, staying away from any minutes that are unsure for the objective of the training.

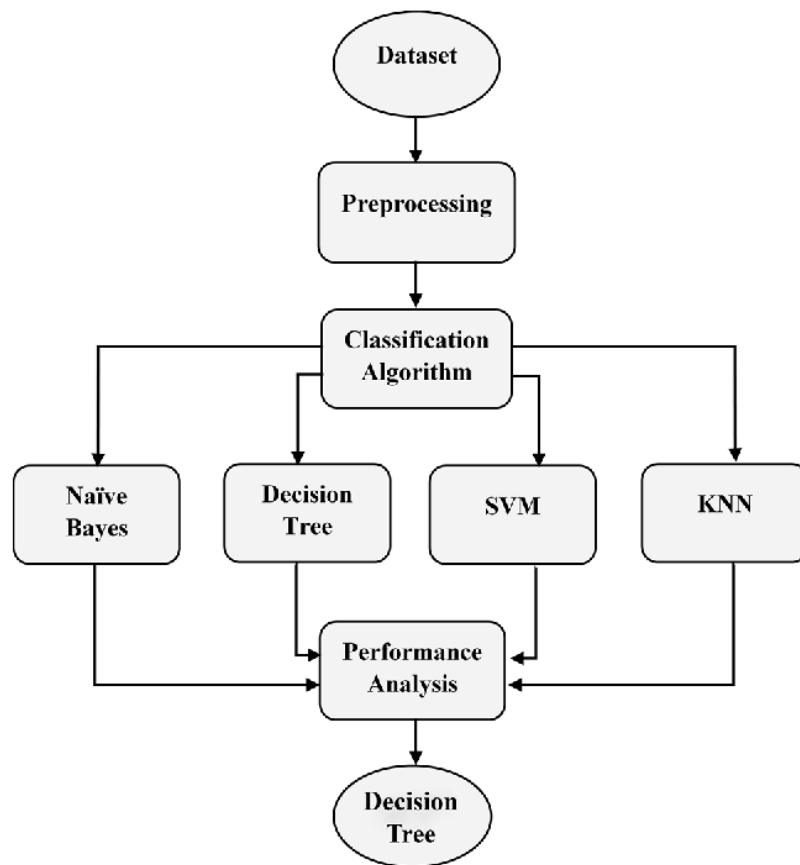
3.2 Case Diagram



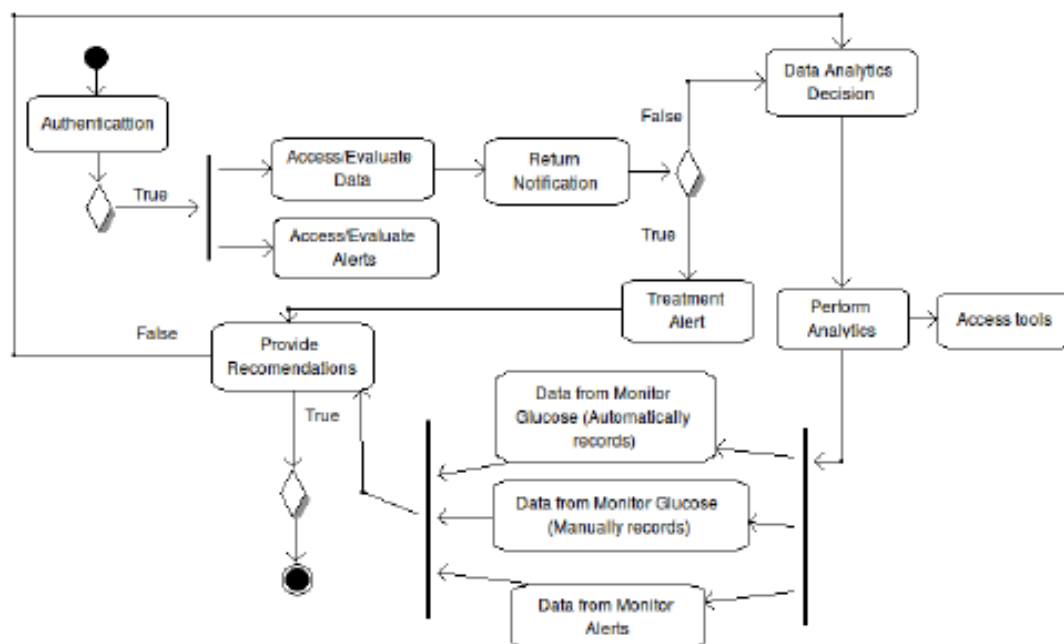
Liver Diagram



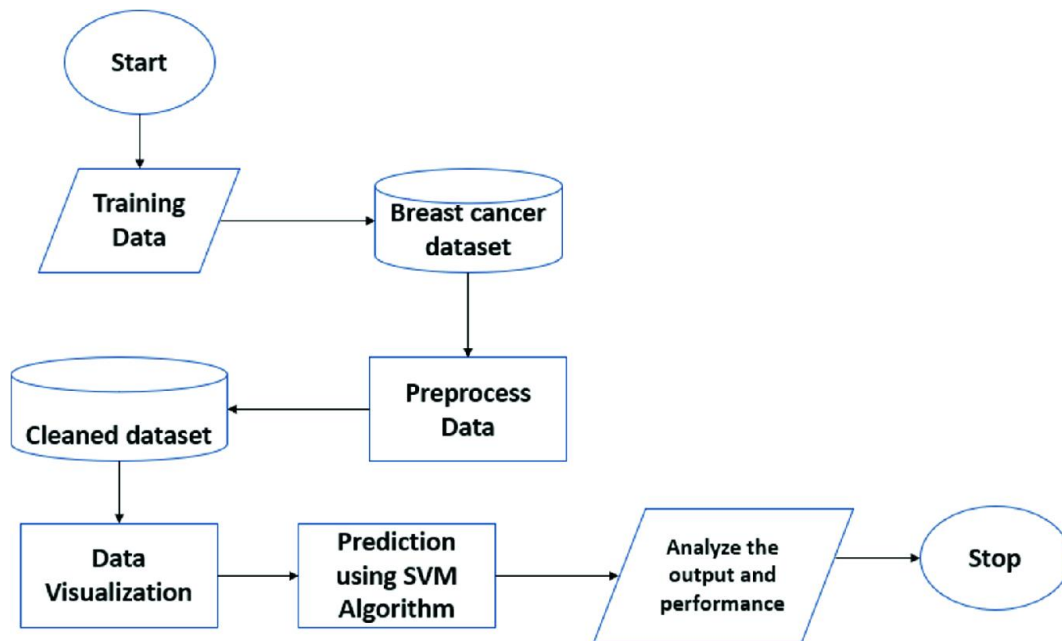
Heart Disease



Kidney Chronic Disease

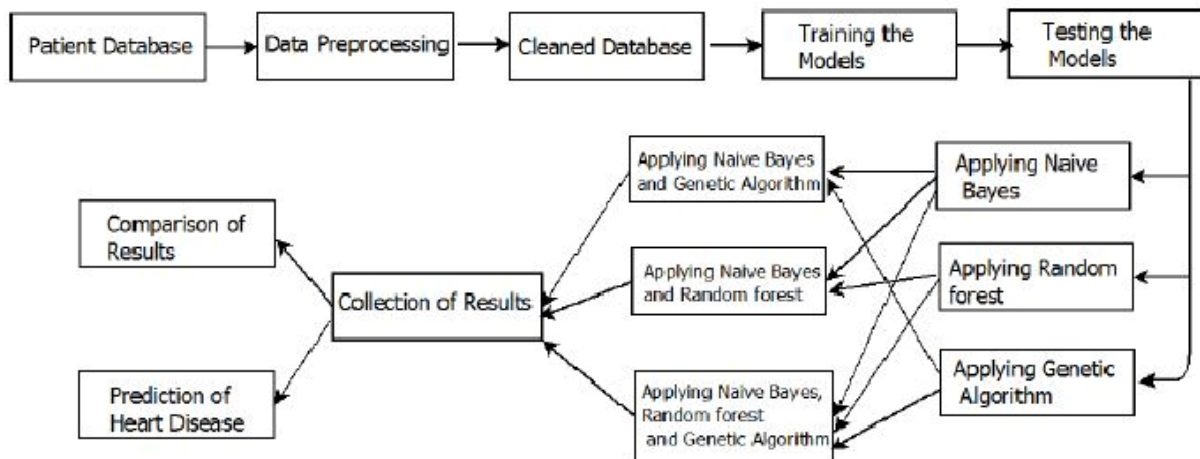


Diabetes Disease

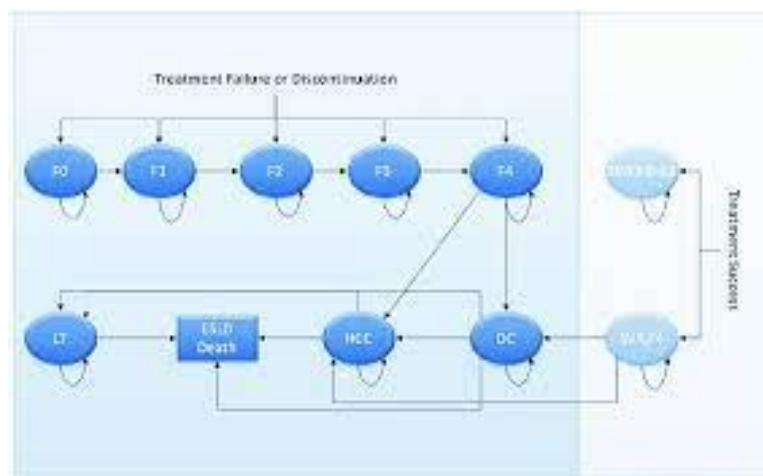
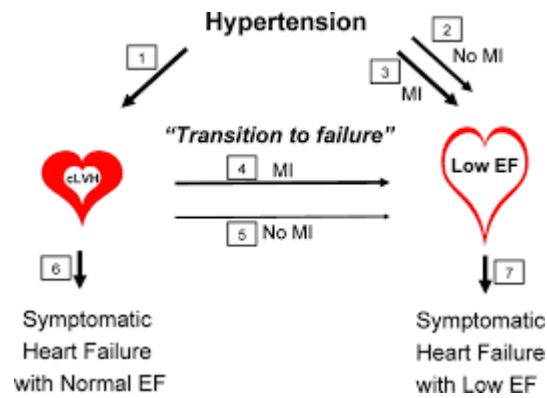
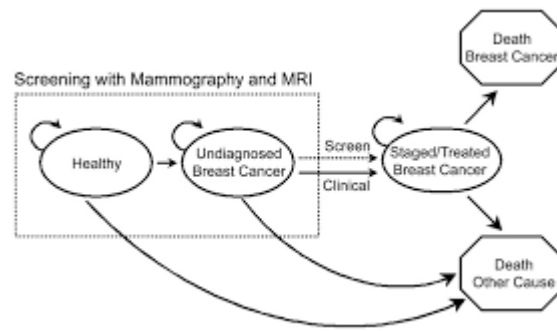


Breast Cancer

3.3 DFD Diagram



3.4 State Transition Diagram



CHAPTER 4

SYSTEM DEVELOPMENT

4.1 Technologies Used

4.1.1 Keras

Keras is a Python library that was first delivered in 2015, regularly utilised for perusing precisely. Keras contains many utilised opening capacities, designs, layers, and so forth. Along these lines, it empowers you to make neural organisations effectively and rapidly. Keras was created and kept up with by François Chollet, and viable with Python 2.7-3.6 (Keras.io, n.d.).

4.1.2 NLTK

The natural language Toolkit (NLTK) might be a broadly utilised text handling library Indigenous Language Processing (NLP). Upholds elite execution tokens, division, partition, and so on. The NLTK group started delivering it in 2001.

4.1.3 Scikit-learn

Scikit-learn is an AI library in Python. It performs simple-to-utilise dimensional decrease techniques like Principal Component Analysis (PCA), grouping strategies like k-10 methods, relapse calculations like calculated relapse, and characterization calculations like arbitrary timberlands.

4.1.4 Pandas

Pandas gives an adaptable stage to information the board inside the information structure. It contains many Open source information investigation apparatuses written in Python, similar to strategies for testing missing information, consolidate information outlines, and resize information structures, and so on.

4.1.5 Tensorflow

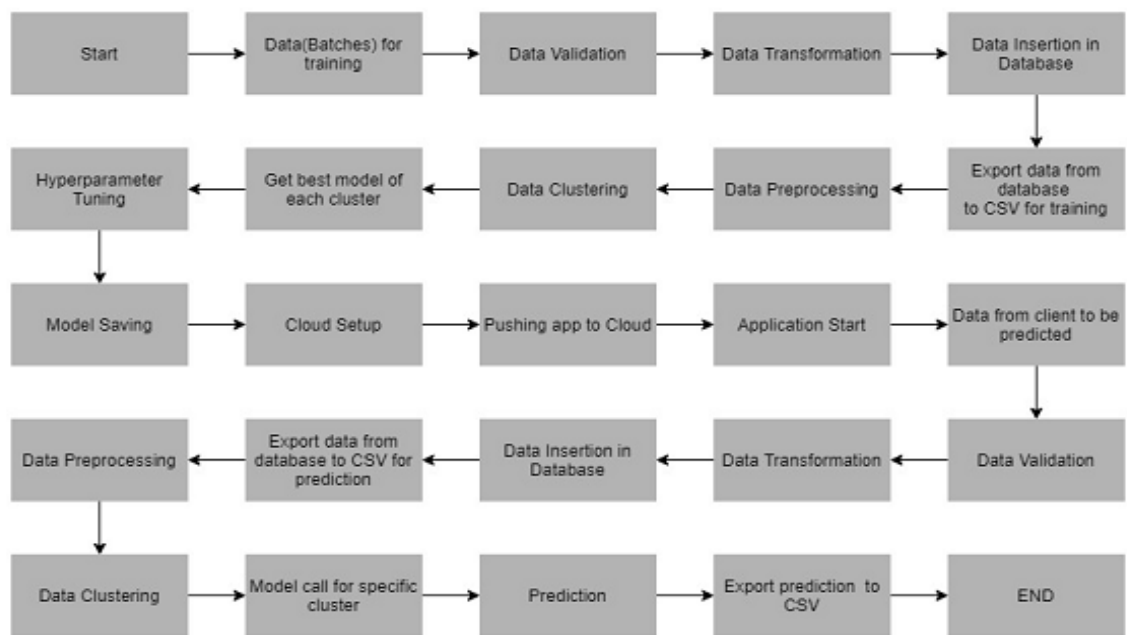
TensorFlow is a free, open-source AI framework that takes you from start to end. It's a full, customizable ecosystem of devices, libraries, and local resources that enable experts to advance the existing situation with machine learning and designers to simply create and utilise great machine learning apps.

4.1.6 Numpy

NumPy is a Python counterfeit language library, which adds support for goliath, multi-sided clusters and lattices, additionally as an outsized assortment of cutting edge numerical capacities to figure on these parts.

4.2 Design of Problem Statement

The workflow Diagram of my project “Healthcare Using Machine Learning” is shown below. Our bigger problem is to create a model or to train a model which is going to predict whether a certain wafer is faulty or not. So we are breaking this bigger problem into a smaller problem e.g; How to read Data, then how to validate the data, then how to do data pre-processing steps, how to train a mode and so on... By combining these small steps we are going to give bigger solutions.



Data Validation

We execute numerous approval arrangements on the specified arrangement of preparing papers throughout this evolution.

1. **Name Validation**-The document names are approved based on the specified name in the pattern record. To utilise for approval, we created a regex design based on the name specified in the composition document. We check for the length of date in the document name, much like the timeframe in the record name, once we approve the example in the

name. Assuming every one of the qualities are according to necessity, we move such documents to "Good_Data_Folder" else we move such records to "Bad_Data_Folder."

2. **Number of Columns** -We approve the number of segments included in the records, and if it differs from the value specified in the blueprint document, the record is moved to the next level "Bad_Data_Folder."
3. **Name of Columns** - The section names have been authorised and should be as old as the construction records. If this is not the case, the record is transferred to "Bad_Data_Folder".
4. **The datatype of columns** - The diagram record specifies the datatype of segments. When we embed the records into the database, this is approved. If the datatype is incorrect, the record is relocated to a new location "Bad_Data_Folder".
5. **Null values in columns** - If all of the characteristics in a part of a document are NULL or missing, we discard the record and move it to the trash "Bad_Data_Folder".

Data Insertion in Database

1. **Database Creation and connection** - Make a data set with the given name passed. Assuming the data set is as of now made, open the association with the information base.
2. **Table creation in the database** - Table with name - "Good_Data", is made in the data set for embedding the records in the "Good_Data_Folder" in view of given section names and information type in the pattern document. Assuming that the table is now present, then, at that point, the new table isn't made and new records are embedded in the all around present table as we need preparing to be done on new just as old documents.
3. **Insertion of files in the table** - Every one of the records in the "Good_Data_Folder" are embedded in the above-made table. In case any record has invalid information type in any of the sections, the document isn't stacked in the table and is moved to "Bad_Data_Folder".

Model Training

1. **Data Export from Db** - The information in a put away data set is traded as a CSV document to be utilised for model preparation.

2. Data Preprocessing

- a. Drop columns are not useful for training the model. Such columns were selected while doing the EDA.
 - b. Replace the invalid values with numpy “nan” so we can use the imputer on such values.
 - c. Encode the categorical values.
 - d. Check for null values in the columns. If present, impute the null values using the KNN imputer.
 - e. After imputing, handle the imbalanced dataset by using RandomOverSampler.
3. **Clustering** - KMeans calculation is utilised to make groups in the preprocessed information. The ideal number of groups is chosen by plotting the elbow plot, and for the unique choice of the quantity of bunches, we are utilising the "KneeLocator" work. The idea behind clustering is to implement different algorithms. To train data in different clusters. The K Means model is trained over preprocessed data and the model is saved for further use in prediction.
4. **Model Selection** - After groups are made, we track down the best model for each bunch. We are utilising two calculations, "Arbitrary Forest" and "KNN". For each group, both the calculations are passed with the best boundaries obtained from GridSearch. We ascertain the AUC scores for the two models and select the model with the best score. Also, the model is chosen for each group. Every one of the models for each bunch are put something aside for use in expectation.

Prediction Data Description

1. Customers will send the data in different arrangements of documents in bunches at a given area. Information will contain Wafer names and 590 segments of different sensor esteems for each wafer.
2. Aside from forecast documents, we additionally require a "pattern" record from customer which contains all the applicable data about the preparation records, for example, Name of the documents, Length of Date esteem in FileName, Length of time esteem in FileName, Number of Columns, Name of the Columns and their datatype.

Prediction

1. **Data Export from Db** - The data in the stored database is exported as a CSV file to be used for prediction.
2. **Data Preprocessing**
 - a) Drop columns are not useful for training the model. Such columns were selected while doing the EDA.
 - b) Replace the invalid values with numpy “nan” so we can use the imputer on such values.
 - c) Encode the categorical values.
 - d) Check for null values in the columns. If present, impute the null values using the KNN imputer.
3. **Clustering** - KMeans model created during training is loaded, and clusters for the preprocessed prediction data is predicted.
4. **Prediction** - Based on the cluster number, the respective model is loaded and is used to predict the data for that cluster.
5. Once the prediction is made for all the clusters, the predictions along with the original names before label encoder are saved in a CSV file at a given location and the location is returned to the client.

4.3 Algorithm / Pseudo Code

In this piece of the report, we talk about the CART choice tree (Classification and Regression Tree). This calculation is one among the preeminent normal AI techniques with the objective of building Prediction models that predicts the value of a variable, which is named target per the upsides of free factors (input information).

Truck arrangement calculation which depends on choice trees which might be a taking in of choice trees from class mark preparing tuples. The Classification and Regression (CART) tree technique utilises recursive apportioning to isolate the preparation records information into fragments with similar to yield field esteems. The CART tree hub starts by analysing the information fields to distinguish the best split, scaled by the decrease in a contamination record that results from the split.

Calculation for the development of the choice tree.

- Make a root hub assume(N);
- In the event that tuples in (D) are the entirety of the indistinguishable class, expect (C);
- Return (N) as a leaf hub named with the classification (C);
- Assuming that the quality rundown is unfilled, thenReturn (N) as a leaf hub marked with the mass class in (D);
- Apply highlight determination technique to look out the least complex parting rule;
- Mark hub (N) with parting standard;
- Characteristic rundown skilled quality rundown – parting attribute;Then for each result expect 'j' of parting model
- Leave 'Dj' alone the arrangement of data tuples in (D) fulfilling result j;

Assuming 'Dj' is vacant, Attach a leaf hub marked with the mass class in (D) to root hub (N);

Else,

join the hub returned by creating choice tree to root hub (N);

Decision Tree calculation has been executed and 92.3% precision score is gotten.

```
[ ] from sklearn.metrics import confusion_matrix, accuracy_score
cm = confusion_matrix(y_test, y_pred)
print(cm)
accuracy_score(y_test, y_pred)

[[46  1  2]
 [ 1 10  0]
 [ 1  0  4]]
0.9230769230769231
```

SVM calculation has been executed and 95.38% precision score is obtained.



```
from sklearn.metrics import confusion_matrix, accuracy_score
cm = confusion_matrix(y_test, y_pred)
print(cm)
accuracy_score(y_test, y_pred)

[[49  0  0]
 [ 1  0  0]
 [ 5  0  0]]
0.9538061538061538
```

ANN calculation has been executed and a 75.38% precision score is gotten.

```
[ ] y_pred = ann.predict(X_test)

[ ] from sklearn.metrics import confusion_matrix, accuracy_score
cm = confusion_matrix(y_test, y_pred)
print(cm)
accuracy_score(y_test, y_pred)

[[49  0  0]
 [ 1  0  0]
 [ 5  0  0]]
0.7538061538061538
```

Logistic Regression has been carried out and 96.92% precision score is acquired.

```
[ ] from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression(random_state = 0, max_iter=500)
classifier.fit(X_train, y_train.ravel())

LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
intercept_scaling=1, l1_ratio=None, max_iter=500,
multi_class='auto', n_jobs=None, penalty='l2',
random_state=0, solver='lbfgs', tol=0.0001, verbose=0,
warm_start=False)

[ ] y_pred = classifier.predict(X_test)

[ ] classifier.score(X_test, y_test)

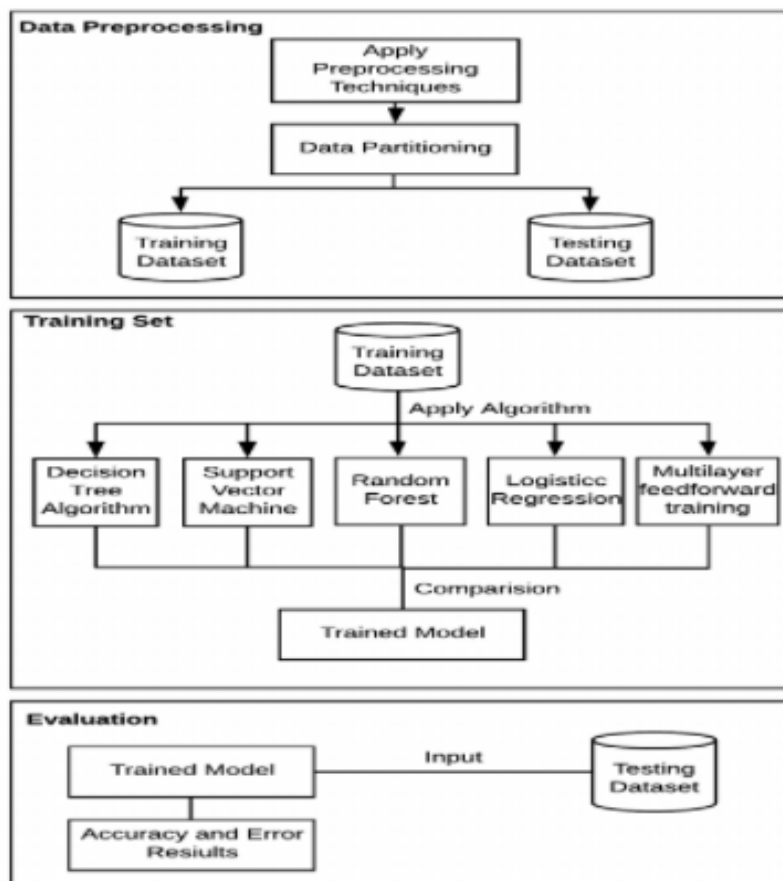
0.9692307692307692
```

KNN calculation has been executed and 93.84% precision score is obtained.

```
[ ] from sklearn.metrics import confusion_matrix, accuracy_score
cm = confusion_matrix(y_test, y_pred)
print(cm)
accuracy_score(y_test, y_pred)

[[49  0  0]
 [ 2  9  0]
 [ 2  0  3]]
0.9384615384615385
```

4.4 Flow Graph



CHAPTER 5

PERFORMANCE ANALYSIS

5.1 Data Set

Breast Cancer Data Set (<https://www.kaggle.com/uciml/breast-cancer-wisconsin-data>)

Data Set Characteristics : Multivariate, Domain-Theory

Number Of Instances : 7200

Region : Life

Quality Characteristics : Categorical, Real

Number Of Attributes : 21

Date Donates : 1987-01-01

Related Tasks : Classification

Missing Values? : N/A

Number Of Web Hits: 268078

5.2 Data Set Features

A digital image of a fine needle suction of a boson mass is used to register highlights. They represent the characteristics of the cell cores in the image. This informative index includes 416 liver patient records and 167 non-liver patient records from India's North East area. The "Dataset" segment may or may not be a class name used to divide bunches into liver patients (liver disease) (no infection). There are 441 male patient records and 142 female patient records in this informative index. The National Institute of Diabetes and Digestive and Kidney Diseases provided the data for this analysis. The dataset's goal is to symptomatically predict whether a patient has diabetes, which is supported by precise analytic estimates included inside the dataset. A few requirements were imposed on the decision-making process in those circumstances based on a larger data set. All of the patients at this facility are Pima Indian women who are at least 21 years old. There are 76 features in this data collection, however only 14 of them are used in completely dispersed trials. To date, ML scientists have mostly employed the Cleveland information base. The "objective" field indicates that the patient has cardiovascular disease. Its value ranges between 0 and 4.

5.2.1 Description of Data Set

The informational index comprises of Age of the patient

- Sex of the patient
- Absolute Bilirubin
- Direct Bilirubin
- Soluble Phosphatase
- Alanine Aminotransferase

- Aspartate Aminotransferase
- Absolute Proteins
- Albumin
- Albumin and Globulin Ratio
- Dataset: field used to divide the information into two sets (patient with liver sickness, or no infection)

5.3 Screenshots of the various stages of the project

Before training our model first of all we must have to balance our Data set.
 Import the required Python libraries.

```
[ ] import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt

import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)

[ ] df = pd.read_csv("cancer.csv")

[ ] df.head()
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dimension_mean	radius_se	texture_se	perimeter_se	area_se
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	0.07871	1.0950	0.9053	8.589	153.40
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667	0.5435	0.7339	3.398	74.08
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	0.05999	0.7456	0.7869	4.585	94.03
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	0.09744	0.4956	1.1560	3.445	27.23
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	0.05883	0.7572	0.7813	5.438	94.44

Diagnosis is the class label. Unnamed 32 consists NaN values. ID has no role.

```
[ ] cols = df.columns
print(cols)

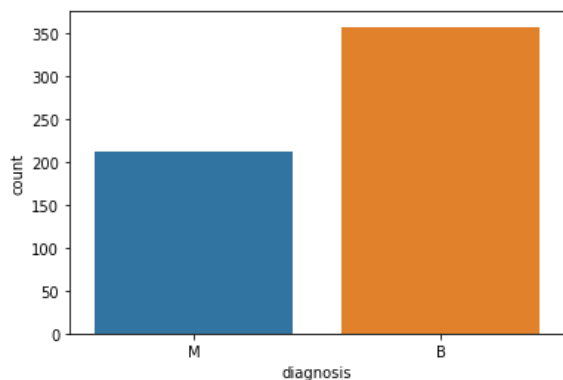
Index(['id', 'diagnosis', 'radius_mean', 'texture_mean', 'perimeter_mean',
       'area_mean', 'smoothness_mean', 'compactness_mean', 'concavity_mean',
       'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
       'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
       'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
       'fractal_dimension_se', 'radius_worst', 'texture_worst',
       'perimeter_worst', 'area_worst', 'smoothness_worst',
       'compactness_worst', 'concavity_worst', 'concave points_worst'],
      dtype='object', name='columns')

[ ] y = df.diagnosis
list = ['Unnamed: 32','id','diagnosis']
x = df.drop(list,axis = 1)
x.head()
```

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean	fractal_dimension_mean	radius_se	texture_se	perimeter_se	area_se	smoothness_se
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	0.07871	1.0950	0.9053	8.589	153.40	0.006399
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667	0.5435	0.7339	3.398	74.08	0.005225
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	0.05999	0.7456	0.7869	4.585	94.03	0.006150
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	0.09744	0.4956	1.1560	3.445	27.23	0.009110
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	0.05883	0.7572	0.7813	5.438	94.44	0.011490

```
[ ] ax = sns.countplot(y, label="Count")
B,M = y.value_counts()
print("Number of Benign: {}".format(B))
print("Number of Malignant: {}".format(M))

Number of Benign: 357
Number of Malignant: 212
```



We can see from the data description that there are no missing values. But if you check the dataset the missing values are replaced with invalid values like '?'. Let's replace such values with 'nan' and check for missing values again.

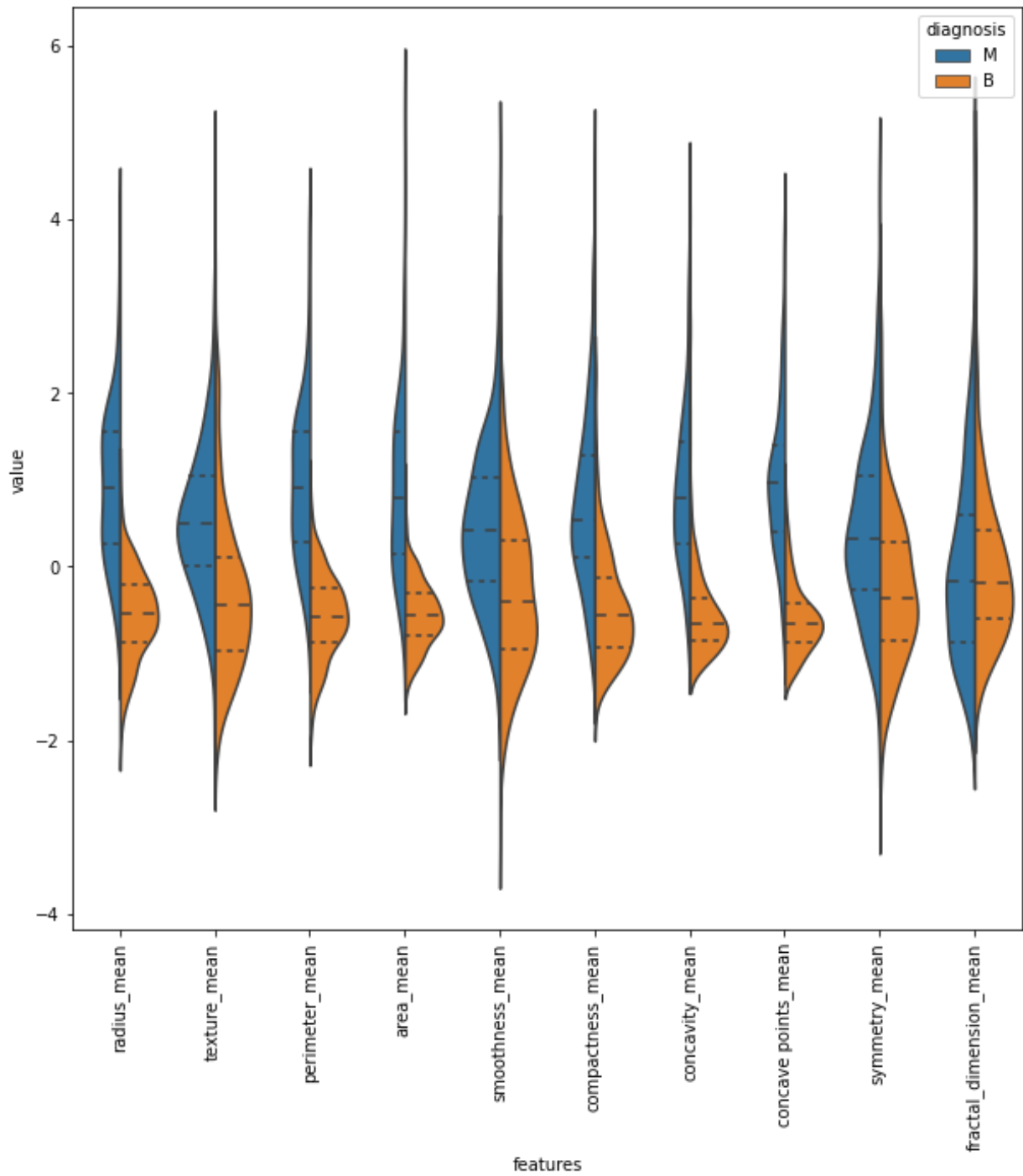
```
x.describe().T
```

	count	mean	std	min	25%	50%	75%	max
radius_mean	569.0	14.127292	3.524049	6.981000	11.700000	13.370000	15.780000	28.11000
texture_mean	569.0	19.289649	4.301036	9.710000	16.170000	18.840000	21.800000	39.28000
perimeter_mean	569.0	91.969033	24.298981	43.790000	75.170000	86.240000	104.100000	188.50000
area_mean	569.0	654.889104	351.914129	143.500000	420.300000	551.100000	782.700000	2501.00000
smoothness_mean	569.0	0.096360	0.014064	0.052630	0.086370	0.095870	0.105300	0.16340
compactness_mean	569.0	0.104341	0.052813	0.019380	0.064920	0.092630	0.130400	0.34540
concavity_mean	569.0	0.088799	0.079720	0.000000	0.029560	0.061540	0.130700	0.42680
concave points_mean	569.0	0.048919	0.038803	0.000000	0.020310	0.033500	0.074000	0.20120
symmetry_mean	569.0	0.181162	0.027414	0.106000	0.161900	0.179200	0.195700	0.30400
fractal_dimension_mean	569.0	0.062798	0.007060	0.049960	0.057700	0.061540	0.066120	0.09744
radius_se	569.0	0.405172	0.277313	0.111500	0.232400	0.324200	0.478900	2.87300
texture_se	569.0	1.216853	0.551648	0.360200	0.833900	1.108000	1.474000	4.88500
perimeter_se	569.0	2.866059	2.021855	0.757000	1.606000	2.287000	3.357000	21.98000
area_se	569.0	40.337079	45.491006	6.802000	17.850000	24.530000	45.190000	542.20000
smoothness_se	569.0	0.007041	0.003003	0.001713	0.005169	0.006380	0.008146	0.03113
compactness_se	569.0	0.025478	0.017908	0.002252	0.013080	0.020450	0.032450	0.13540
concavity_se	569.0	0.031894	0.030186	0.000000	0.015090	0.025890	0.042050	0.39600
concave points_se	569.0	0.011796	0.006170	0.000000	0.007638	0.010930	0.014710	0.05279
symmetry_se	569.0	0.020542	0.008266	0.007882	0.015160	0.018730	0.023480	0.07895
fractal_dimension_se	569.0	0.003795	0.002646	0.000895	0.002248	0.003187	0.004558	0.02984
radius_worst	569.0	16.269190	4.833242	7.930000	13.010000	14.970000	18.790000	36.04000

So these are the columns which have missing values but missing values are replaced with '?'. We will replace these values with 'nan' and then do an imputation of these missing values. Also, we can see that for column 'TBG' all the values are missing. So we will drop this column as it is of no use to us.

```
[ ] data_dia = y
data = x
data_n_2 = (data - data.mean()) / (data.std())
data = pd.concat([y,data_n_2.iloc[:,0:10]],axis=1)
data = pd.melt(data,id_vars="diagnosis",
              var_name="features",
              value_name='value')
plt.figure(figsize=(10,10))
sns.violinplot(x="features", y="value", hue="diagnosis", data=data,split=True, inner="quart")
plt.xticks(rotation=90)

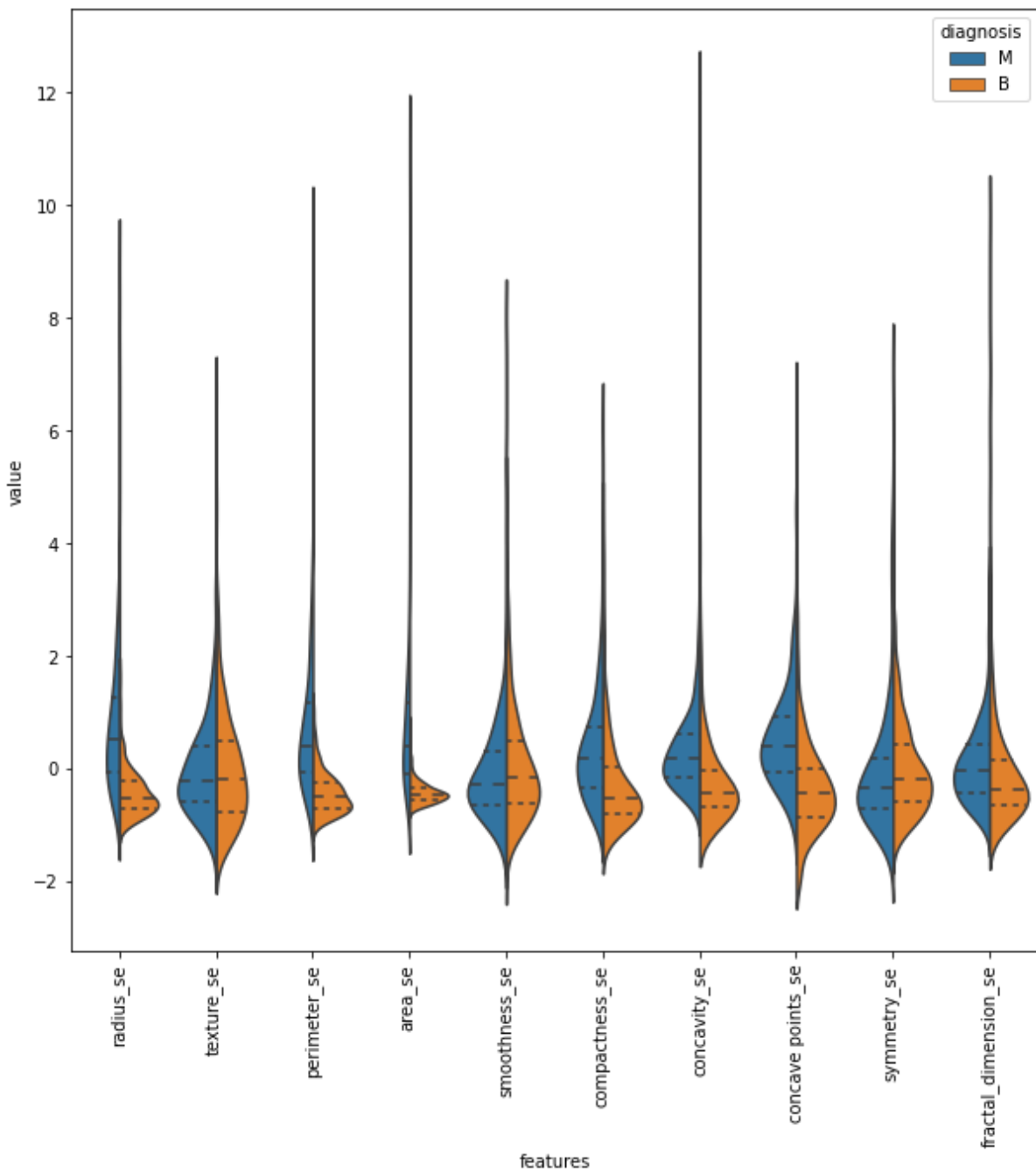
(array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9]),
 <a list of 10 Text major ticklabel objects>)
```



Also, looking at the dataset, we can see that some columns with true and false values are just the indication of whether the next column has values or not.

```
[ ] data = pd.concat([y,data_n_2.iloc[:,10:20]],axis=1)
data = pd.melt(data,id_vars="diagnosis",
               var_name="features",
               value_name='value')
plt.figure(figsize=(10,10))
sns.violinplot(x="features", y="value", hue="diagnosis", data=data,split=True, inner="quart")
plt.xticks(rotation=90)

(array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9]),
 <a list of 10 Text major ticklabel objects>)
```



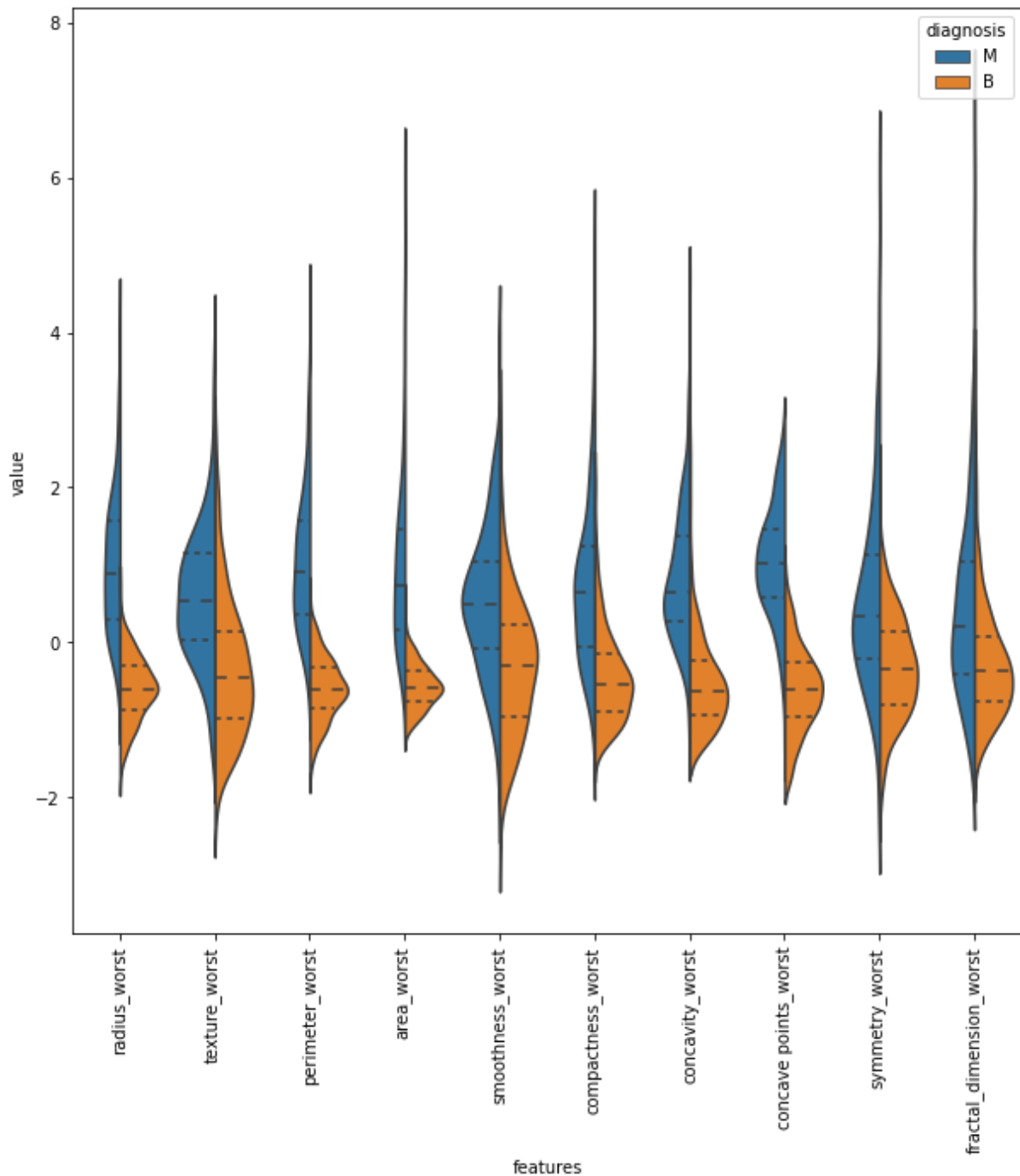
Since, we are anyway going to handle the missing values, there is no point of having such columns in our dataset.

Let's drop such columns as well.

```
[ ] # Third ten features
data = pd.concat([y,data_n_2.iloc[:,20:31]],axis=1)
data = pd.melt(data,id_vars="diagnosis",
               var_name="features",
               value_name='value')

plt.figure(figsize=(10,10))
sns.violinplot(x="features", y="value", hue="diagnosis", data=data,split=True, inner="quart")
plt.xticks(rotation=90)

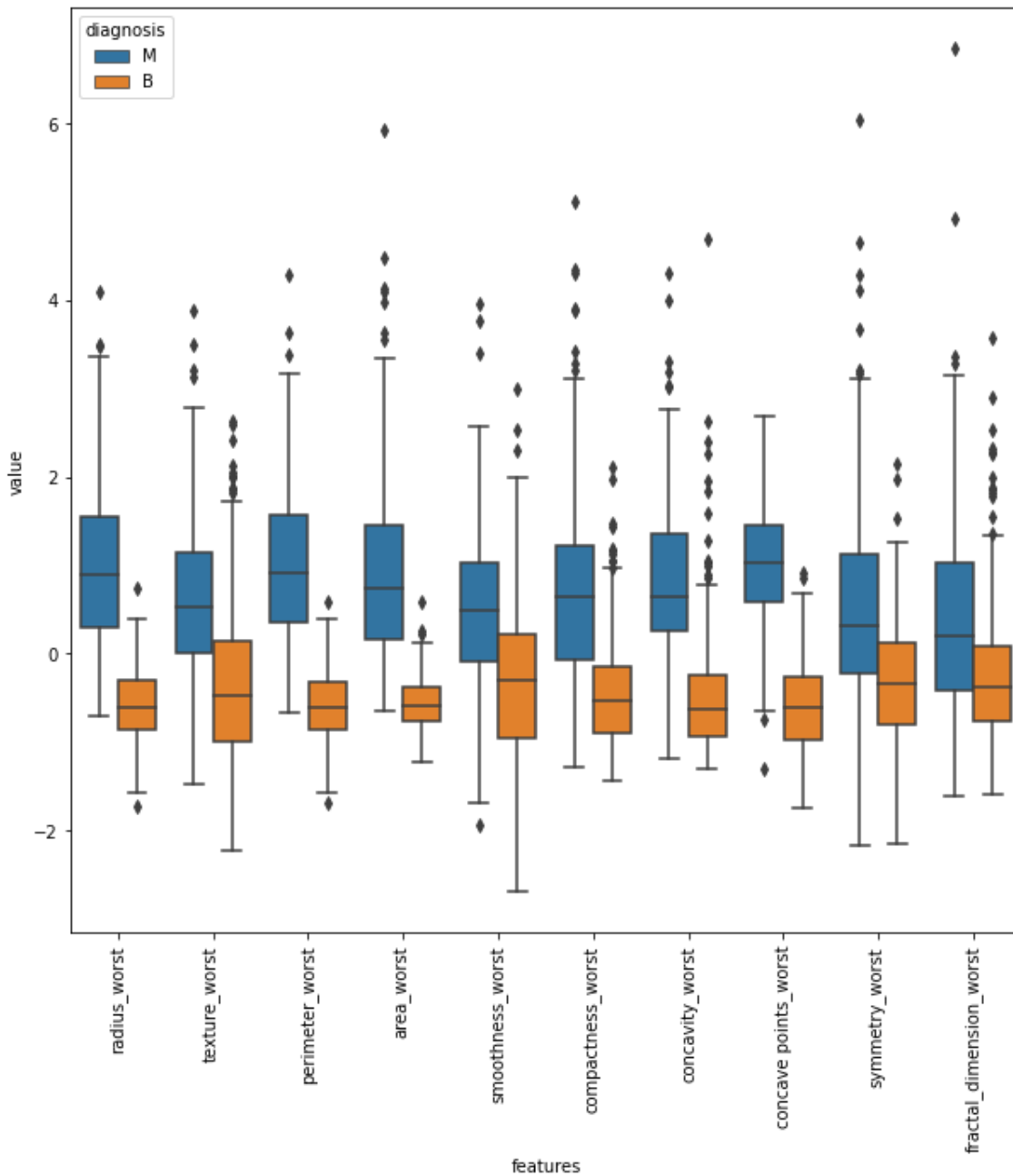
(array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9]),
 <a list of 10 Text major ticklabel objects>)
```



Now that we have replaced all such values with 'nan'. Let's deal with these missing values now.

```
[ ] # As an alternative of violin plot, box plot can be used
# box plots are also useful in terms of seeing outliers
# I do not visualize all features with box plot
# In order to show you lets have an example of box plot
# If you want, you can visualize other features as well.
plt.figure(figsize=(10,10))
sns.boxplot(x="features", y="value", hue="diagnosis", data=data)
plt.xticks(rotation=90)
```

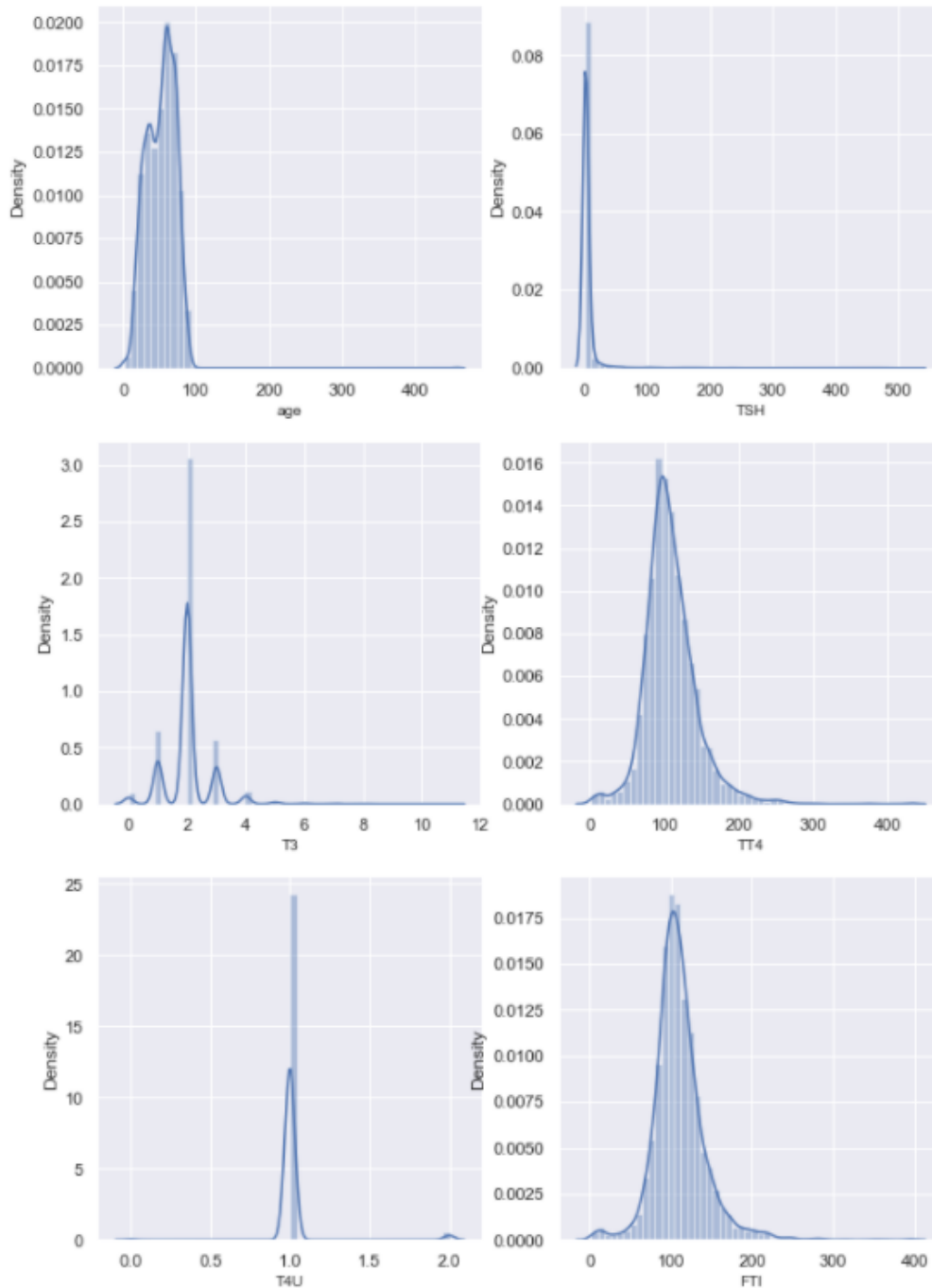
```
(array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9]),
 <a list of 10 Text major ticklabel objects>)
```



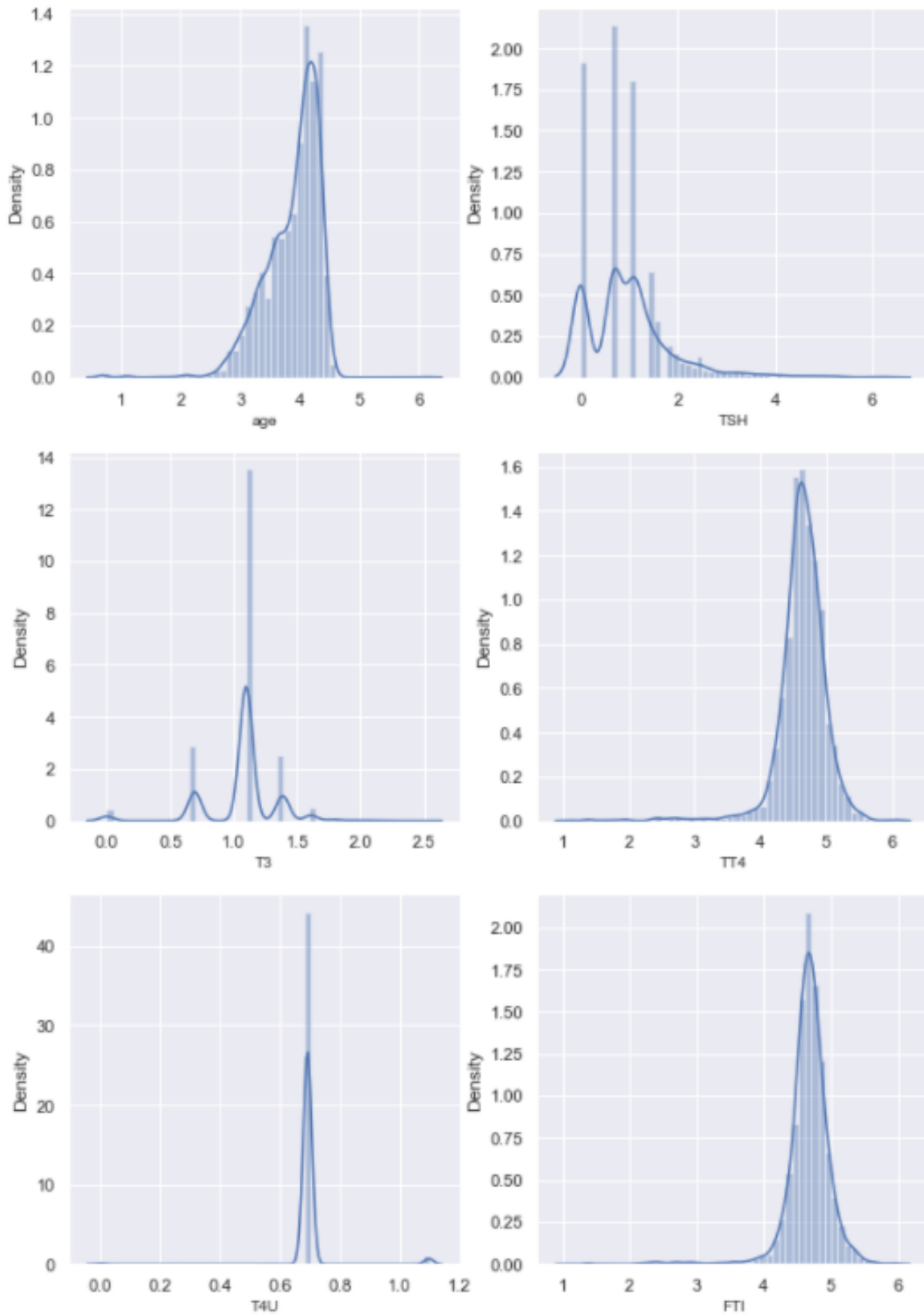
Since the values are categorical, we have to change them to numerical before we use any imputation techniques.

We can use get dummies but since most of the columns have only two distinct categories we will use mapping for them. Why? Because since there are only two categories then the two columns formed after get_dummies will both have very high correlation since they both explain the same thing. So in any case we will have to drop one of the columns. That's why let's use mapping for such columns. For columns with more than two categories we will use get dummies.

Let's check the distribution for our continuous data in the dataset:

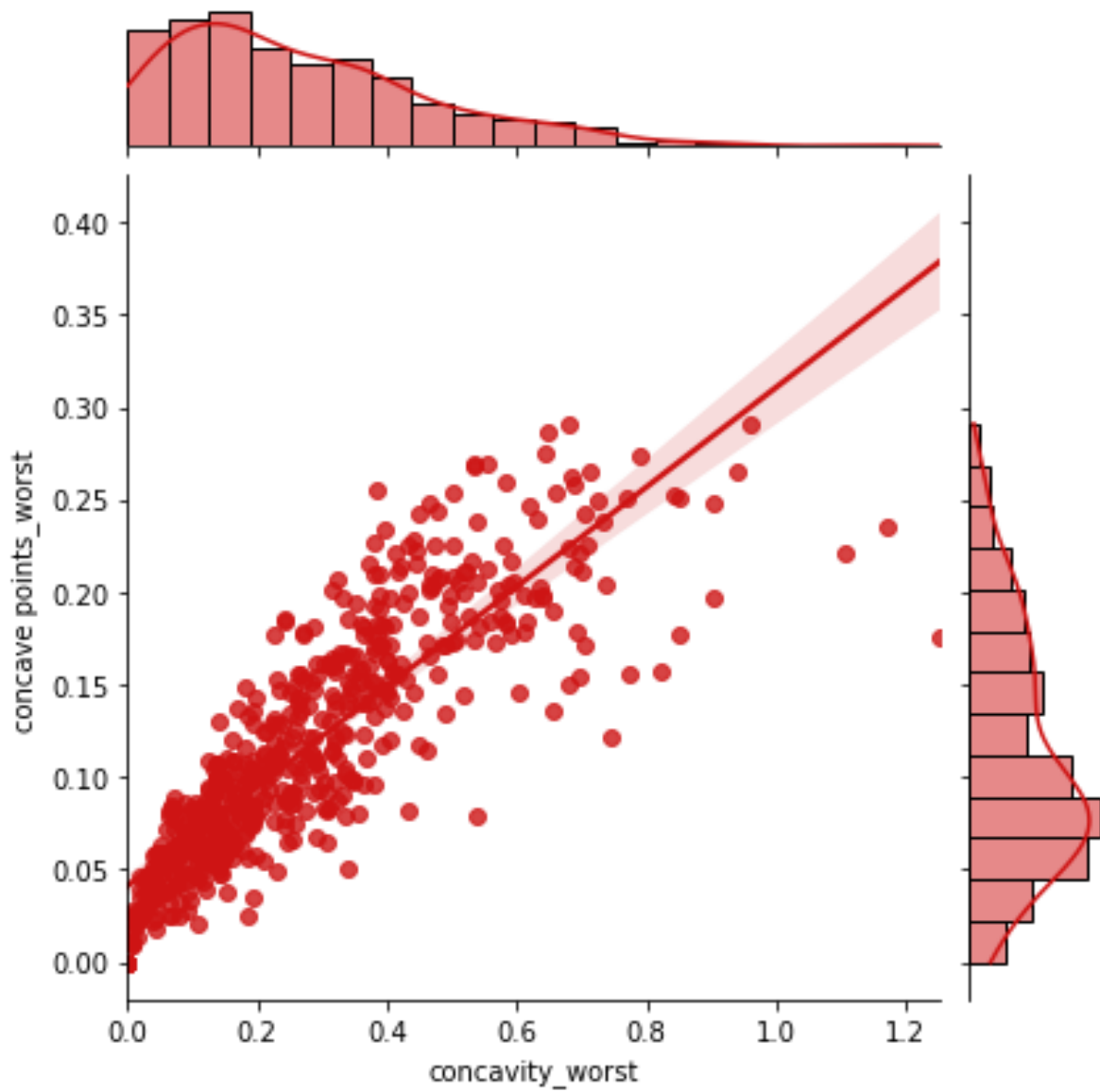


Before doing log transformation , let's add 1 to each value in the column to handle exceptions when we try to find a log of '0'.



```
[ ] sns.jointplot(x.loc[:, 'concavity_worst'], x.loc[:, 'concave points_worst'], kind="reg", color="#ce1414")
```

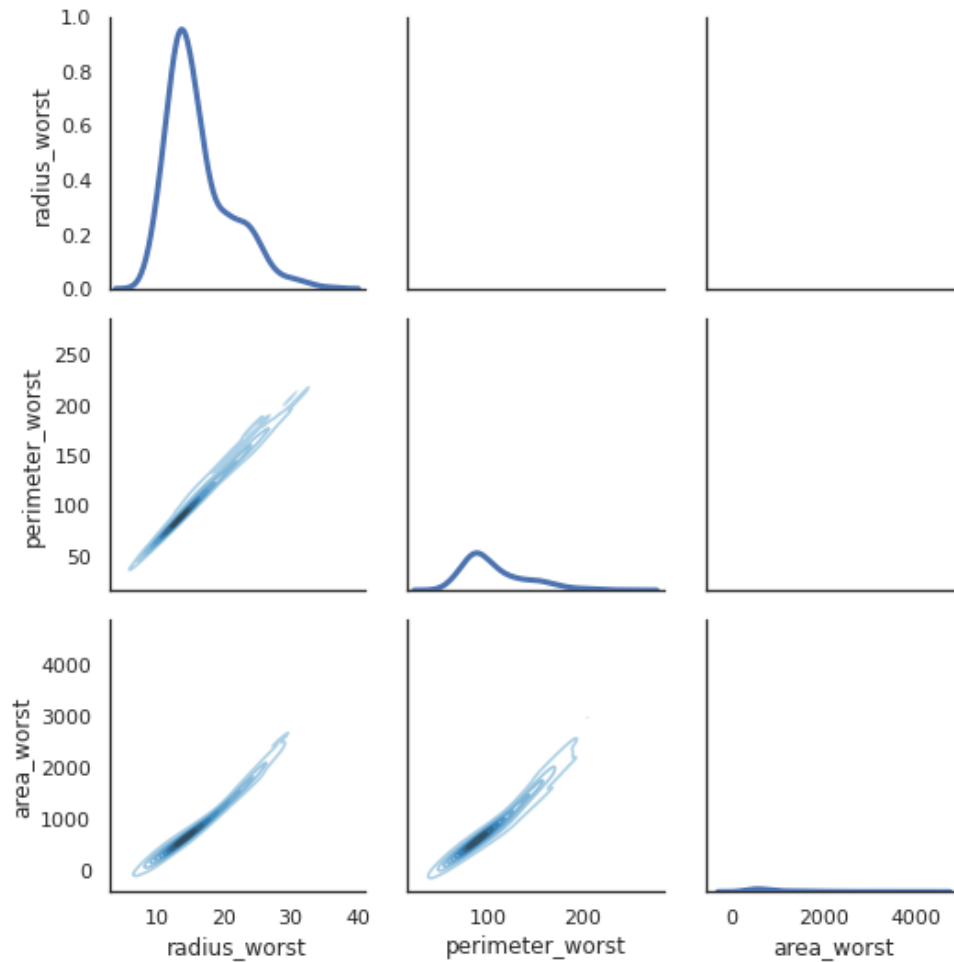
```
<seaborn.axisgrid.JointGrid at 0x7fbb0245470>
```



These features are highly correlated.

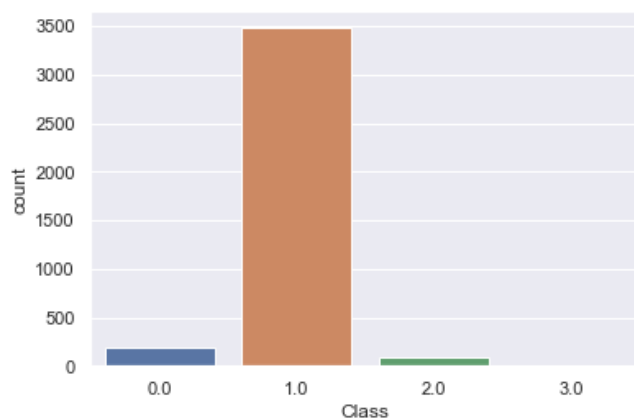
```
sns.set(style="white")
df = x.loc[:,['radius_worst', 'perimeter_worst', 'area_worst']]
g = sns.PairGrid(df, diag_sharey=True)
g.map_lower(sns.kdeplot, cmap="Blues_d")
g.map_upper(plt.scatter, cmap="Blues_d")
g.map_diag(sns.kdeplot, lw=3)
```

<seaborn.axisgrid.PairGrid at 0x7fbbccbe6198>



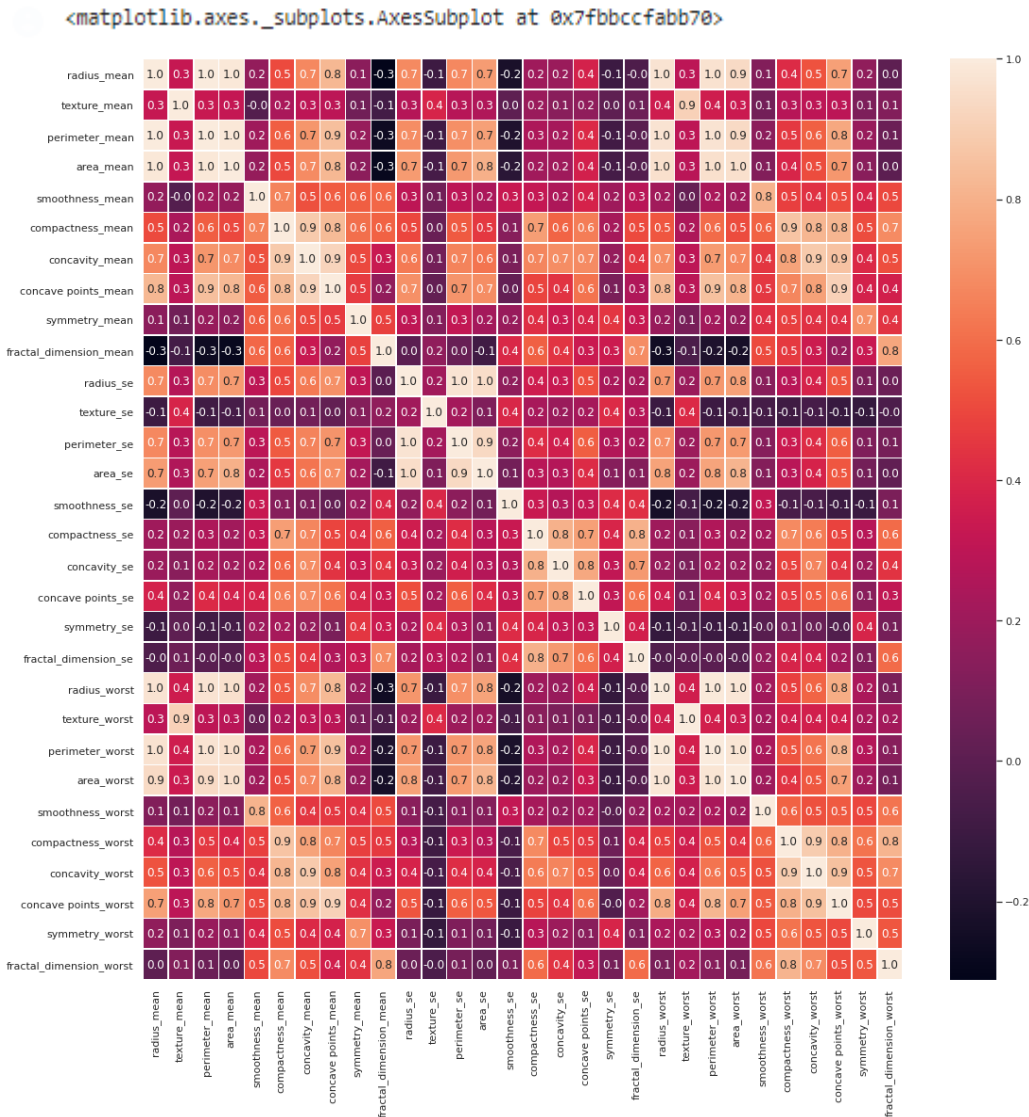
let's see how balanced our dataset in terms of given target classes:

```
sns.countplot(new_data['Class'])
```

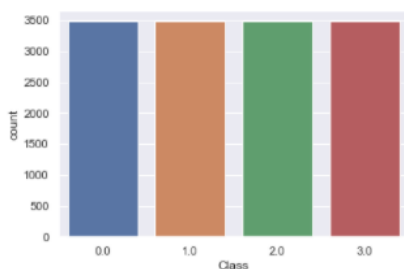


We can clearly see that the dataset is highly imbalanced. We will use a python library known as imbalanced-learn to deal with imbalanced data. Imbalanced learning has an algorithm called RandomOverSampler.

```
[ ] #correlation map
f,ax = plt.subplots(figsize=(18, 18))
sns.heatmap(x.corr(), annot=True, linewidths=.5, fmt= '.1f',ax=ax)
```



Out[37]: <AxesSubplot:xlabel='Class', ylabel='count'>



Our dataset looks balanced now. We can go ahead with training our model on this data.

5.4 Libraries and Requirements

```
APScheduler==3.6.3
attrs==19.3.0
certifi==2019.11.28
Click==7.0
colorhash==1.0.2
configparser==4.0.2
cycler==0.10.0
Flask==1.1.1
Flask-Cors==3.0.8
Flask-MonitoringDashboard==3.0.6
imbalanced-learn==0.6.1
imblearn==0.0
importlib-metadata==1.4.0
itsdangerous==1.1.0
Jinja2==2.11.0
joblib==0.14.1
jsonschema==3.2.0
kiwisolver==1.1.0
kneed==0.5.1
MarkupSafe==1.1.1
matplotlib==3.1.2
more-itertools==8.1.0
numpy==1.18.1
pandas==0.25.3
psutil==5.6.7
pyparsing==2.4.6
pysistent==0.15.7
python-dateutil==2.8.1
pytz==2019.3
PyYAML==5.3
regex==0.1
scikit-learn==0.22.1
scipy==1.4.1
six==1.14.0
sklearn==0.0
SQLAlchemy==1.3.13
tzlocal==2.0.0
Werkzeug==0.16.1
wincertstore==0.2
xgboost==0.90
zipp==2.0.1
```

Main.py:-

```
from wsgiref import simple_server
from flask import Flask, request
from flask import Response
import os
from flask_cors import CORS, cross_origin
from prediction_Validation_Insertion import pred_validation
from trainingModel import trainModel
from training_Validation_Insertion import train_validation
import flask_monitoringdashboard as dashboard
from predictFromModel import prediction

os.putenv('LANG', 'en_US.UTF-8')
os.putenv('LC_ALL', 'en_US.UTF-8')

app = Flask(__name__)
dashboard.bind(app)
CORS(app)

@app.route("/predict", methods=['POST'])
@cross_origin()
def predictRouteClient():
    try:
        if request.json['folderPath'] is not None:
            path = request.json['folderPath']
            pred_val = pred_validation(path)
            pred_val.prediction_validation()
            pred = prediction(path)
            path = pred.predictionFromModel()
            return Response("Prediction File created at %s!!!" % pa

    except ValueError:
        return Response("Error Occurred! %s" %ValueError)
    except KeyError:
        return Response("Error Occurred! %s" %KeyError)
    except Exception as e:
        return Response("Error Occurred! %s" %e)

@app.route("/train", methods=['POST'])
@cross_origin()
def trainRouteClient():
    try:
        if request.json['folderPath'] is not None:
            path = request.json['folderPath']
            train_valObj = train_validation(path)
            train_valObj.train_validation()
            trainModelObj = trainModel()
```

Application Logging

Logger.py:-

```
from datetime import datetime

class App_Logger:
    def __init__(self):
        pass

    def log(self, file_object, log_message):
        self.now = datetime.now()
        self.date = self.now.date()
        self.current_time = self.now.strftime("%H:%M:%S")
        file_object.write(
            str(self.date) + "/" + str(self.current_time) + "\t\t" + log_message + "\n")
```

Best Model Finder

Breast_Cancer.py:-

```
from flask import Flask, render_template, url_for, flash, redirect
import joblib
from flask import request
import numpy as np

app = Flask(__name__, template_folder='templates')

@app.route("/")

@app.route("/cancer")
def cancer():
    return render_template(r"C:\Users\Alok Singh\Desktop\HealthApp\Indivisual_Deployment\Breast_Cancer_API\cancer_model.pkl")

def ValuePredictor(to_predict_list, size):
    to_predict = np.array(to_predict_list).reshape(1,size)
    if(size==5):
        loaded_model = joblib.load('cancer_model.pkl')
        result = loaded_model.predict(to_predict)
    return result[0]

@app.route('/predict', methods = ["POST"])
def predict():
    if request.method == "POST":
        to_predict_list = request.form.to_dict()
        to_predict_list = list(to_predict_list.values())
        to_predict_list = list(map(float, to_predict_list))
        #cancer
        if(len(to_predict_list)==5):
            result = ValuePredictor(to_predict_list,5)

            if(int(result)==1):
                prediction = "Sorry you chances of getting the disease. Please consult the doctor immediately"
            else:
                prediction = "No need to fear. You have no dangerous symptoms of the disease"
            return(render_template("result.html", prediction_text=prediction))

if __name__ == "__main__":
    app.run(debug=True)
```

Diabetes_Disease.py:-

```
from flask import Flask, render_template, url_for, flash, redirect
import joblib
from flask import request
import numpy as np

app = Flask(__name__, template_folder='templates')

@app.route("/")

@app.route("/Diabetes")
def cancer():
    return render_template("diabetes.html")

def ValuePredictor(to_predict_list, size):
    to_predict = np.array(to_predict_list).reshape(1,size)
    if(size==6):
        loaded_model = joblib.load(r"C:\Users\Alok Singh\Desktop\HealthApp\Indivisual_Deployment\Diabetes_API\diabetes_model.pkl")
        result = loaded_model.predict(to_predict)
    return result[0]

@app.route('/predict', methods = ["POST"])
def predict():
    if request.method == "POST":
        to_predict_list = request.form.to_dict()
        to_predict_list = list(to_predict_list.values())
        to_predict_list = list(map(float, to_predict_list))
        #diabetes
        if(len(to_predict_list)==6):
            result = ValuePredictor(to_predict_list,6)

            if(int(result)==1):
                prediction = "Sorry you chances of getting the disease. Please consult the doctor immediately"
            else:
                prediction = "No need to fear. You have no dangerous symptoms of the disease"
            return(render_template("result.html", prediction_text=prediction))

if __name__ == "__main__":
    app.run(debug=True)
```

Heart_Disease.py:-

```

from flask import Flask, render_template, url_for, flash, redirect
import joblib
from flask import request
import numpy as np

app = Flask(__name__, template_folder='templates')

@app.route("/")

@app.route("/Heart")
def cancer():
    return render_template("heart.html")

def ValuePredictor(to_predict_list, size):
    to_predict = np.array(to_predict_list).reshape(1,size)
    if(size==7):
        loaded_model = joblib.load(r'C:\Users\Alok Singh\Desktop\HealthApp\Indivisual_Deployment\Heart_API\heart_model.pkl')
        result = loaded_model.predict(to_predict)
    return result[0]

@app.route('/predict', methods = ["POST"])
def predict():
    if request.method == "POST":
        to_predict_list = request.form.to_dict()
        to_predict_list = list(to_predict_list.values())
        to_predict_list = list(map(float, to_predict_list))
        #diabetes
        if(len(to_predict_list)==7):
            result = ValuePredictor(to_predict_list,7)

            if(int(result)==1):
                prediction = "Sorry you chances of getting the disease. Please consult the doctor immediately"
            else:
                prediction = "No need to fear. You have no dangerous symptoms of the disease"
            return(render_template("result.html", prediction_text=prediction))

if __name__ == "__main__":
    app.run(debug=True)

```

Kidney_Disease.py

```

from flask import Flask, render_template, url_for, flash, redirect
import joblib
from flask import request
import numpy as np

app = Flask(__name__, template_folder='templates')

@app.route("/")

@app.route("/kidney")
def cancer():
    return render_template("kidney.html")

def ValuePredictor(to_predict_list, size):
    to_predict = np.array(to_predict_list).reshape(1,size)
    if(size==7):
        loaded_model = joblib.load(r'C:\Users\Alok Singh\Desktop\HealthApp\Indivisual_Deployment\Kidney_API\kidney_model.pkl')
        result = loaded_model.predict(to_predict)
    return result[0]

@app.route('/predict', methods = ["POST"])
def predict():
    if request.method == "POST":
        to_predict_list = request.form.to_dict()
        to_predict_list = list(to_predict_list.values())
        to_predict_list = list(map(float, to_predict_list))
        #diabetes
        if(len(to_predict_list)==7):
            result = ValuePredictor(to_predict_list,7)

            if(int(result)==1):
                prediction = "Sorry you chances of getting the disease. Please consult the doctor immediately"
            else:
                prediction = "No need to fear. You have no dangerous symptoms of the disease"
            return(render_template("result.html", prediction_text=prediction))

if __name__ == "__main__":
    app.run(debug=True)

```

Liver_Disease.py

```
from flask import Flask, render_template, url_for, flash, redirect
import joblib
from flask import request
import numpy as np

app = Flask(__name__, template_folder='templates')

@app.route("/")
@app.route("/liver")
def cancer():
    return render_template("liver.html")

def ValuePredictor(to_predict_list, size):
    to_predict = np.array(to_predict_list).reshape(1,size)
    if(size==7):
        loaded_model = joblib.load(r'C:\Users\Alok Singh\Desktop\HealthApp\Indivisual_Deployment\Liver_API\liver_model.pkl')
        result = loaded_model.predict(to_predict)
    return result[0]

@app.route("/predict", methods = ["POST"])
def predict():
    if request.method == "POST":
        to_predict_list = request.form.to_dict()
        to_predict_list = list(to_predict_list.values())
        to_predict_list = list(map(float, to_predict_list))
        #liver
        if(len(to_predict_list)==7):
            result = ValuePredictor(to_predict_list,7)

            if(int(result)==1):
                prediction = "Sorry you chances of getting the disease. Please consult the doctor immediately"
            else:
                prediction = "No need to fear. You have no dangerous symptoms of the disease"
            return(render_template("result.html", prediction_text=prediction))

if __name__ == "__main__":
    app.run(debug=True)
```

Data Type Validation Insertion Prediction

DataTypeValidationPrediction.py

```
import shutil
import sqlite3
from datetime import datetime
from os import listdir
import os
import csv
from application_logging.logger import App_Logger

class dbOperation:
    def __init__(self):
        self.path = "Prediction Database/"
        self.badFilePath = "Prediction_Raw_Files_Validation/Bad_Raw"
        self.goodFilePath = "Prediction_Raw_Files_Validation/Good_Raw"
        self.logger = App_Logger()

    def DataBaseConnection(self, DatabaseName):
        try:
            conn = sqlite3.connect(self.path+DatabaseName+'.db')
            file = open("Prediction_Logs/DataBaseConnectionLog.txt", 'a+')
            self.logger.log(file, "Opened %s database successfully" % DatabaseName)
            file.close()
        except ConnectionError:
            file = open("Prediction_Logs/DataBaseConnectionLog.txt", 'a+')
            self.logger.log(file, "Error while connecting to database: %s" % ConnectionError)
            file.close()
            raise ConnectionError
        return conn

    def CreateTableDb(self, DatabaseName, column_names):
        try:
            conn = self.DataBaseConnection(DatabaseName)
            conn.execute("DROP TABLE IF EXISTS Good_Raw_Data;")
            for key in column_names.keys():
                type = column_names[key]
                try:
                    conn.execute("ALTER TABLE Good_Raw_Data ADD COLUMN \"{column_name}\" {dataType}".format(column_name=key, dataType=type))
                except:
                    conn.execute("CREATE TABLE Good_Raw_Data ({column_name} {dataType})".format(column_name=key, dataType=type))
            conn.close()
            file = open("Prediction_Logs/DbTableCreateLog.txt", 'a+')
            self.logger.log(file, "Tables created successfully!!")
```

Data Type Validation Insertion Training

DataTypeValidation.py

```
import shutil
import sqlite3
from datetime import datetime
from os import listdir
import os
import csv
from application_logging.logger import App_Logger

class DBOperation:
    def __init__(self):
        self.path = 'Training_Database/'
        self.badFilePath = "Training_Raw_files_validated/Bad_Raw"
        self.goodFilePath = "Training_Raw_files_validated/Good_Raw"
        self.logger = App_Logger()

    def databaseConnection(self, DatabaseName):
        try:
            conn = sqlite3.connect(self.path+DatabaseName+'.db')

            file = open("Training_Logs/DataBaseConnectionLog.txt", 'a+')
            self.logger.log(file, "Opened %s database successfully" % DatabaseName)
            file.close()
        except ConnectionError:
            file = open("Training_Logs/DataBaseConnectionLog.txt", 'a+')
            self.logger.log(file, "Error while connecting to database: %s" % ConnectionError)
            file.close()
            raise ConnectionError
        return conn

    def createTableDb(self, DatabaseName, column_names):
        try:
            conn = self.databaseConnection(DatabaseName)
            c=conn.cursor()
            c.execute("SELECT count(name) FROM sqlite_master WHERE type = 'table'AND name = 'Good_Raw_Data'")
            if c.fetchone()[0] ==1:
                conn.close()
                file = open("Training_Logs/DbTableCreateLog.txt", 'a+')
                self.logger.log(file, "Tables created successfully!!")
                file.close()

                file = open("Training_Logs/DataBaseConnectionLog.txt", 'a+')
                self.logger.log(file, "Closed %s database successfully" % DatabaseName)
                file.close()
            else:
                for key in column_names.keys():
```

Prediction Raw Data Validation

PredictionDataValidation.py

```
import sqlite3
from datetime import datetime
from os import listdir
import os
import re
import json
import shutil
import pandas as pd
from application_logging.logger import App_Logger

class Prediction_Data_validation:
    def __init__(self,path):
        self.Batch_Directory = path
        self.schema_path = 'schema_prediction.json'
        self.logger = App_Logger()

    def valuesFromSchema(self):
        try:
            with open(self.schema_path, 'r') as f:
                dic = json.load(f)
                f.close()
                pattern = dic['SampleFileName']
                LengthOfDateStampInFile = dic['LengthOfDateStampInFile']
                LengthOfTimeStampInFile = dic['LengthOfTimeStampInFile']
                column_names = dic['ColumnName']
                NumberOfColumns = dic['NumberOfColumns']

                file = open("Training_Logs/valuesfromSchemaValidationLog.txt", 'a+')
                message = "LengthOfDateStampInFile:: %s \ LengthOfDateStampInFile + '\t' + "LengthOfTimeStampInFile:: %s" % LengthOfTimeStampInFile + "\t" + "NumberOfColumns:: %s" % NumberOfColumns + "\n"
                self.logger.log(file,message)
                file.close()

        except ValueError:
            file = open("Prediction_Logs/valuesfromSchemaValidationLog.txt", 'a+')
            self.logger.log(file, "ValueError: Value not found inside schema_training.json")
            file.close()
            raise ValueError

        except KeyError:
            file = open("Prediction_Logs/valuesfromSchemaValidationLog.txt", 'a+')
```

Training Raw Data Validation

RawValidation.py:-

```
import sqlite3
from datetime import datetime
from os import listdir
import os
import re
import json
import shutil
import pandas as pd
from application_logging.logger import App_Logger

class Raw_Data_validation:

    def __init__(self,path):
        self.Batch_Directory = path
        self.schema_path = 'schema_training.json'
        self.logger = App_Logger()

    def valuesFromSchema(self):
        try:
            with open(self.schema_path, 'r') as f:
                dic = json.load(f)
                f.close()
            pattern = dic['SampleFileName']
            LengthOfDateStampInFile = dic['LengthOfDateStampInFile']
            LengthOfTimeStampInFile = dic['LengthOfTimeStampInFile']
            column_names = dic['ColName']
            NumberofColumns = dic['NumberofColumns']

            file = open("Training_Logs/valuesfromSchemaValidationLog.txt", 'a+')
            self.logger.log(file,message)

            file.close()

        except ValueError:
            file = open("Training_Logs/valuesfromSchemaValidationLog.txt", 'a+')
            self.logger.log(file,"ValueError:Value not found inside schema_training.json")
            file.close()
            raise ValueError

        except KeyError:
            file = open("Training_Logs/valuesfromSchemaValidationLog.txt", 'a+')
            self.logger.log(file, "KeyError:Key value error incorrect key passed")
            file.close()
            raise KeyError

        except Exception as e:
            file = open("Training_Logs/valuesfromSchemaValidationLog.txt", 'a+')
            self.logger.log(file, str(e))
```

5.4 Output





Atypical Angina

Resting Blood Pressure (in mm Hg)

455

Serum Cholesterol in mg/dl

25

Fasting Blood Sugar

Fasting Blood Sugar < 120 mg/dl

Resting Electro-cardiographic Result

Normal

Maximum Heart Rate Achieved

221

Exercise Induced Angina

Yes

Predict

Sorry you chances of getting the disease. Please consult the doctor immediately

CHAPTER 6

Results

6.1 Discussion on the result achieved

As indicated by the examination on AI calculations, highlights choices procedures, and induction strategies this work proposed an expectation instrument for diagnosing illness upheld the best performed AI calculation and furthermore the best performed include choices strategies with the intelligent deduction methods. In this manner, we proposed a substitution apparatus that upheld the decision tree calculation. The prescient apparatus showed an order exactness of 99.82 %, which underlined the device ability on Dataset.

The report presented the term of characterization on notable AI classifiers on our five distinctive illness datasets. Arrangement turns into a hotly debated issue inside the field of AI in view of the capability of diminishing complex data and giving a manual for treatment and anticipation, helping with smoothing out information assortment, and helping clinicians to actually look at viability and productivity of medicines. This review affirms that order calculations can further develop the forecast precision. Furthermore, there's no best blend of classifiers that suits all datasets. In any case, no particular component choice technique suits all AI devices.

In view of the trials on various AI calculations and Dataset, the review might infer that the CART calculation performed better precision contrasted and two notable calculations. Table 1 shows that by utilising support vector machine calculation (SVM) the best precision has been acquired as follows. Expectation exactness for preparing dataset 92.45% Prediction precision for testing dataset 93.06%. Utilising k-closest neighbour calculation (KNN) the best precision has been acquired as follows. Expectation precision for preparing dataset 94 % though Prediction exactness for testing dataset 94.82% .With 8 closest neighbours which has been chosen utilising GridSearchCV work that does all the work and returns the least complex k boundary .

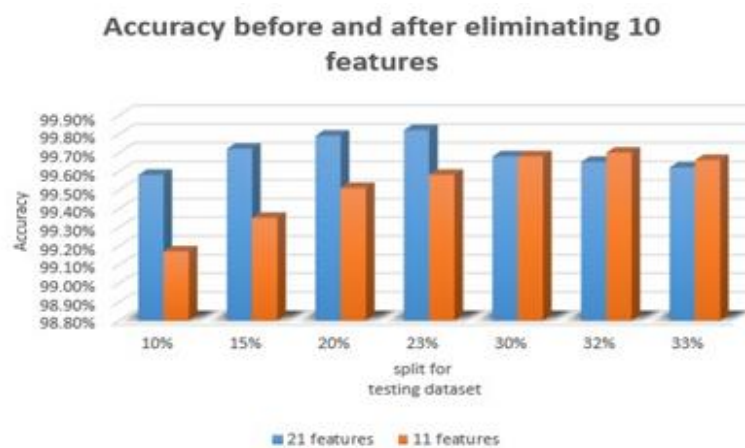
Algorithm	Accuracy (%) Training dataset	Accuracy (%) Testing dataset
Decision tree CART	99.62%	99.82%
Support vector machine SVM	92.45%	93.45%
k-nearest neighbor algorithm KNN	94%	94.82%

Table 1: Accuracy comparison for three algorithms

Among different preprocessing strategies, during this review we applied initially guideline part examination (PCA) for include choice yet that precision diminished from 99.644 to 97.533. For the point of diminishing the amount of highlights to be prepared to plan an appropriate GUI, since the dataset has 21 elements data gain quality assessment has been applied as an element determination technique. a good positioning has been acquired which assists with killing the ten least significant qualities and keep the contrary 11 credits. With 11 highlights, we got a 99.70 % precision, which isn't a lot diminishing, since the best exactness accomplished with each of the 21 elements is 99.82% with a 23% testing dataset . In this manner, we chose to wipe out those 10 highlights to make our device interface more adequate and simpler to utilise. In Table 2, there are diverse precision esteems with various parting of dataset contrasting prior and then afterward applying the element choice strategy and killing the more modest elements, the given beneath chart shows this examination besides.

Percentage split for testing dataset	Accuracy (%) With 21 features	Accuracy (%) With 11 features
10 %	99.58 %	99.17 %
15 %	99.72 %	99.35 %
20 %	99.79 %	99.51 %
23 %	99.82 %	99.58 %
30 %	99.68 %	99.68 %
32 %	99.65 %	99.70 %
33 %	99.62 %	99.66 %

Table 2: Specifies the values of Accuracy before and after eliminating 10 features.



Accuracy comparison graph for the splitting percentage

6.2 Application

As a quickly developing field, there's a huge scope of expected utilizations of AI inside the medical services field which can envelop assistant parts of the area like work force the board, protection approaches, administrative undertakings, and undeniably more. Accordingly, the points covered during this part are restricted directly down to three normal uses of AI.

The first is the utilisation of AI in clinical pictures like reverberation imaging (MRIs), automated hub tomography (CAT) checks, ultrasound (US) imaging, and positron outflow tomography (PET) examines. The aftereffects of these imaging modalities could be a set or series of pictures which by and large requires a radiologist to decipher and make a finding. ML methods have quickly been progressing to foresee and find pictures which can show a sickness state or major issue.

The second is tongue handling of clinical reports. With the push towards electronic clinical records (EMR) in numerous nations, the agreement from numerous medical services experts has been that the strategy is slow, dreary, and, by and large, totally messed up. This may at times bring about more unfortunate general medical services for patients. One among the preeminent difficulties is the measure of actual clinical records and documentation that as of now exists in numerous clinics and facilities. Diverse arranging, manually written notes, and a plenty of fragmented or non-brought together data has made the change to embracing electronic clinical records yet proficient.

The third AI application incorporates the use of human hereditary qualities to foresee illness and track down reasons for sickness. With the appearance of cutting edge sequencing (NGS) procedures and along these lines the blast of hereditary information including enormous data sets of populace wide hereditary data, the attempt to perceive significant data of what hereditary qualities might mean for human wellbeing is presently at the front line of the many exploration tries. By seeing how complex illnesses might show and the manner in which hereditary qualities might increment or diminishing an individual's danger can support deterrent medical services. This may give doctors more data en route to tailor a specific patients' consideration to decrease the risk of securing more intricate illnesses.

6.3 Limitations

In spite of the fact that AI has progressed significantly in the clinical world, human reconnaissance is as yet fundamental. For instance, medical procedure robots work coherently, instead of sympathetically. Wellbeing specialists might see fundamental social perceptions that can help analyse or forestall unexpected problems.

Patient requirements regularly reach out past prompt states of being. Social, financial and chronicled elements can play into proper proposals for specific patients. For example, an AI framework might have the option to dispense a patient to a specific consideration place dependent on a particular analysis. Notwithstanding, this framework may not represent patient financial limitations or other customised inclinations.

Protection likewise turns into an issue while joining an AI framework. Brands like Amazon have free rules with regards to gathering and utilising information. Medical clinics, then again, may confront a few mishaps when endeavouring to channel information from Apple cell

phones, for example. These administrative and social limitations might restrict AI's capacity to work with clinical practices.

As AI utilises information to make frameworks more astute and more exact, cyberattacks will join AI to become more intelligent with every achievement and disappointment, making them more hard to foresee and forestall. When harming dangers out-move security protections, the assaults will be substantially more testing to address.

6.4 Future Work

Future work of my undertaking is depicted as follows. The current undertaking lived fundamentally on arrangement precision on the grounds that the primary rules for estimating the exhibition of proposed apparatuses. Notwithstanding, future work will target different standards like characterization speed and computational expense.

As of late, an endless advancement in wellbeing labourer frameworks has been seen, and furthermore the frameworks now accessible are developed enough for designated reception practically speaking. Notwithstanding, master frameworks might be logically incorporated in medical clinic data frameworks to convey medical services significantly more adequately.

In general illnesses are frequently precarious to analyse on the grounds that side effects are effortlessly mistaken for other disease conditions. At the point when illness is gotten early, treatment can handle the dysfunctionality. During this review, diagnosing illness is pointed out with an AI device called MLTDD (AI apparatus for infection finding). MLTD might be a device, which is introduced to endocrinologists or understudies, to foresee determination with 99.7% precision for infections. With regards to the examination, utilising this apparatus as far as the determination cycle gives a more beautiful and variegated climate for the clients than enormous and hard-covered materials. Be that as it may, understudies reading up endocrinology for illnesses can utilise this device for testing their insight by contrasting their forecasts and MLTDD.

REFERENCES

<https://www.kaggle.com/uciml/breast-cancer-wisconsin-data>
<https://www.kaggle.com/uciml/pima-indians-diabetes-database>
<https://www.kaggle.com/uciml/indian-liver-patient-records>
<https://www.kaggle.com/ronitf/heart-disease-uci>
<https://www.kaggle.com/mansoordaku/ckdisease>
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5321289/>
<https://link.springer.com/article/10.1007/s40009-020-00979->
<https://www.hindawi.com/journals/jhe/2018/8087624/>
<https://www.frontiersin.org/articles/10.3389/fonc.2020.591846/full>
<https://www.intechopen.com/chapters/72044>
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7233077/>
<https://www.ajronline.org/doi/full/10.2214/AJR.15.15813>

[1] Gunter-TD, and, Terry-NP, the Emergence of National Electronic Health Record Architectures in the United States and Australia: Models, Costs, and Questions. J Med Internet Res. 2005.

[2] Michael B. Zimmermann, Iodine deficiency. Endocr Rev 2009. Available: <http://press.endocrine.org/doi/abs/10.1210/er.2009-0011>

[3] Mark P. J. Vanderpump and W. Michael and G. Tunbridge, The epidemiology of thyroid diseases. In: Braverman LE, Utiger RD, editors. Werner and Ingbar's The Thyroid: A Fundamental and Clinical Text. Ninth edn. Philadelphia: JB Lippincott-Raven; 2005.

[4] Tunbridge WMG, Evered DC, Hall R, et al. The spectrum of thyroid disease in the community, 1977.

[5] Vander JB, Gaston EA, Dawber TR. The significance of nontoxic thyroid nodules: final report of a 15-year study of the incidence of malignancy. Ann Intern Med 1968.

[6] Reiners C, Wegscheider K, Schicha H, et al. Prevalence of thyroid disorders in the working population of Germany: ultrasonography screening in 96,278 unselected employees.

[7] National Cancer Institute. Physician Data Query (PDQ). Breast Cancer Treatment – Patient Version. 2019. Accessed at <https://www.cancer.gov/types/breast/patient/breasttreatment-pdq> on July 31, 2019.

[8] Henry NL, Shah PD, Haider I, Freer PE, Jagsi R, Sabel MS. Chapter 88: Cancer of the Breast. In: Niederhuber JE, Armitage JO, Doroshow JH, Kastan MB, Tepper JE, eds. Abeloff's Clinical Oncology. 6th ed. Philadelphia, Pa: Elsevier; 2020.

[9] International agency for research on cancer, world health organization GLOBOCAN 2012 estimated cancer [incidence , mortality and prevalence worldwide in 2012. http://globocan.iarc.fr/Pages/fact_sheets_population.aspx](http://globocan.iarc.fr/Pages/fact_sheets_population.aspx)

- [10] Grütters A, Krude H, on the management of congenital hypothyroidism. 2007.
- [11] Right diagnosis for health grades. Web site: <http://www.rightdiagnosis.com/t/thyroid/stats-country.htm#extrapwarning> .
- [12] Rokach, L. and O. Maimon, eds. Data Mining and Knowledge Discovery, 2010. [13] Tarca, A.L., et al., Machine Learning and Its Applications to Biology. PLoS Comput Biol, 2007.
- [14] Machine Learning Stanford University, <https://www.coursera.org/learn/machine-learning>, Instructors, Andrew Ng, Associate Professor, Stanford University; Chief Scientist, Baidu; Chairman and Co-founder, Coursera
- [15] Jagsi R, King TA, Lehman C, Morrow M, Harris JR, Burstein HJ. Chapter 79: Malignant Tumors of the Breast. In: DeVita VT, Lawrence TS, Lawrence TS, Rosenberg SA, eds. DeVita, Hellman, and Rosenberg's Cancer: Principles and Practice of Oncology. 11th ed. Philadelphia, Pa: Lippincott Williams & Wilkins; 2019.
- [16] Venture radar, Discover the Innovators and Disruptors, VentureRadar discovers and ranks companies, making them visible to potential partners, customers, and investors. Website: <http://blog.ventureradar.com/2015/09/30/top-5-companies-revolutionizing-healthcare-with-machine-learning/>
- [17] The editor-in-chief at MachineLearningMastery.com. Jason is a husband, father, researcher, author, professional programmer and a machine learning practitioner. Website: <http://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/>
- [18] Rokach, L., Data mining with decision trees: theory and applications. Vol. 69. 2007: World scientific.
- [19] IDF Diabetes Atlas [article online], 2009. 4th ed. Brussels, Belgium, International Diabetes Federation. Available from www.diabetesatlas.org. Accessed 1 April 2010 [Google Scholar](#)
- [20] Boyle P, Bernard L World Cancer Report 2008 [article online], 2008. Boyle P, Bernard L Eds. Cedex, France, World Health Organisation, International Agency for Research on Cancer. Available from <http://www.iarc.fr/en/publications/pdfs-online/wcr/index.php>. Accessed 1 April 2010 [Google Scholar](#)