A

MINI PROJECT REPORT

on

ONE YEAR LIFE EXPECTANCY POST THORACIC SURGERY USING IBM WATSON STUDIO IN MACHINE LEARNING

Submitted to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ,HYDERABAD

In partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

(DATA SCIENCE)

(BATCH : B-07)

NIMMARAJULA SUPRIYA	:227Y1A67C0
DOMBALE UMADEVI	:227Y1A67C1
GADUDASU VAISHNAVI	:227Y1A67C2

Under the Guidance Of

Ms. B. Madhavi (Assistant Professor)

DEPARTMENT OF INFORMATION TECHNOLOGY

MARRI LAXMAN REDDY

INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(AUTONOMOUS)

JUNE 2025

(Affiliated to JNTU-H, Approved by AICTE New Delhi and Accredited by NBA & NAAC With ‘A’ Grade)

MARRI LAXMAN REDDY
INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(AN AUTONOMOUS INSTITUTION)
(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)
Accredited by NBA and NAAC with 'A' Grade & Recognized Under Section2(f) & 12(B)of the UGC act,1956

CERTIFICATE **Date:**

This is to certify that the project report titled "**ONE YEAR LIFE EXPECTANCY POST THORACIC SURGERY USING IBM WATSON STUDIO**" is being submitted by **NIMMARAJULA SUPRIYA (227Y1A67C0), DOMBALE UMADEVI (227Y1A67C1) AND GADUDASU VAISHNAVI (227Y1A67C2)** students of Department of Computer Science Engineer (Data Science), is a record of bonafide work carried out by the members during a period from January, 2025 to June, 2025 under the supervision of **Ms. B.Madhavi, Assistant Professor, Department of Data Science**. This project is done as a fulfilment of obtaining Bachelor of Technology Degree to be awarded by Jawaharlal Nehru Technological University Hyderabad, Hyderabad.

The matter embodied in this project report has not been submitted by us to any other university for the award of any other degree.

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Date: **(Ms. B.Madhavi)**

The Viva-Voce Examination of above students, has been held on………………………

Head of the Department **External Examiner**

Principal/Director

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our guide Ms. B. Madhavi, Assistant Professor, Department of Computer Science and Engineering(Data Science), for her excellent guidance and invaluable support, which helped us accomplish the B.Tech (CSD) degree and prepared us to achieve more life goals in the future. Her total support of our dissertation and countless contributions to our technical and professional development made for a truly enjoyable and fruitful experience. Special thanks are dedicated for the discussions we had on almost every working day during our project period and for reviewing our dissertation.

We are very much grateful to our Project Coordinator, Dr. M. Sunita, Associate Professor, Department of Computer Science and Engineering(Data Science), MLRITM, Dundigal, Hyderabad, who has not only shown utmost patience, but was fertile in suggestions, vigilant in directions of error and has been infinitely helpful.

We are extremely grateful to Dr. A. Arun Kumar, Head of the Department of Computer Science and Engineering (Data Science), MLRITM, Dundigal, Hyderabad, for the moral support and encouragement given in completing our project work.

We wish to express deepest gratitude and thanks to Dr. R. Murali Prasad, Principal, and Dr.P.Sridhar, Director for their constant support and encouragement in providing all the facilities in the college to do the project work.

We would also like to thank all our faculties, administrative staff and management of MLRITM, who helped us to completing the mini project.

On a more personal note, we thank our beloved parents and friends for their moral support during the course of our project.

TABLE OF FIGURE

MARRI LAXMAN REDDY
INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(AN AUTONOMOUS INSTITUTION)
(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)
Accredited by NBA and NAAC with 'A' Grade & Recognized Under Section2(f) & 12(B)of the UGC act,1956

ABSTRACT PROFORMA

MINI PROJECT

Year & Branch: 2025 & CSD	**Section:** B	**Batch No.: 07**
Academic Year: 2024 – 2025		**Regulation:** R22
Student Registration Details	**Name**	**Roll Number**
	1. NIMMARJULA SUPRIYA	227Y1A67C0
	2. DOMBALE UMADEVI	227Y1A67C1
	3. GADUDASU VAISHNAVI	227Y1A67C2
Name of the Guide & Designation	Ms. Madhavi, Assistant Professor	
Area (Domain) of the Project	MACHINE LEARNING	
Title of the Project	ONE YEAR LIFE EXPECTANCY POST THORACIC SURGERY USING IBM WATSON STUDIO IN MACHINE LEARNING	
Tools Required	Python Programming Language, Pandas and Numpy, Scikit-learn, Matplotlib and Seaborn, Streamlit, CSV File for Dataset	

Abstract:

This project focuses on predicting one-year survival after thoracic surgery using machine learning models. A dataset of 470 patient records from the Wroclaw Thoracic Surgery Centre was used, analyzing clinical features like pain, cough, haemoptysis, smoking, and asthma. Several algorithms, including Decision Tree, Random Forest, SVM, and Naïve Bayes, were evaluated. A Deep Neural Network model delivered the highest accuracy. The solution aids in identifying high-risk patients and enhances post-surgical planning, offering significant value to healthcare decision-making.

Guide **Project Coordinator** **HOD**

ABSTRACT

The scope of this paper is to propose a life expectancy rate and examine the mortality after thoracic surgery which takes into account the different importance of various features which can have an effect in the end result. The data of the patients collected after diagnosis have been used as the dataset. Various metrics which affect the result have been analyzed with the help of random forest and decision tree algorithms to better understand the consequences of post-Surgery.

Particular metrics have been selected according to their weightage on the main outcome for prediction. Thus, it enables us to have better comprehension of various algorithms and also some important parameters are selected for the construction of a better model. In addition, we have several classification features such as the presence of pain before surgery, hemoptysis before surgery, cough before surgery, whether the patient is a smoker, whether the patient has asthma, and a few others.

This classification model predicts whether the patient will survive for a year-long period or not with better selection of the data features. Many machine learning models like Multi-layer perceptron (MLP), SVM, Naïve Bayes, Decision Tree, Random forest, and Logistic regression have been applied for post thoracic surgery life expectancy prediction based on datasets from UCI. Also, work has been carried out towards attribute ranking and selection in performing better in improving prediction accuracy with machine learning algorithms. So accordingly, we here have developed a Deep Neural Network based approach in the prediction of post thoracic Life expectancy which is the most advanced form of Neural Networks. This is based on a dataset obtained from Wroclaw Thoracic Surgery Centre machine learning repository which contained 470 instances. On comparing the accuracy, the results indicate that the deep neural network can be efficiently used for predicting life expectancy.

CHAPTER 1

INTRODUCTION

1.1 . MAIN IDEA OF THE PROJECT

The introduction of computer applications into the medical industry has had a direct impact on doctors' productivity and accuracy in recent years. One of these applications is the study of health outcomes. In most nations, cancer is now one of the leading causes of mortality. Thoracic surgery is the most common operation performed on lung cancer patients. Massive datasets of cancer have been collected and made available to medical professionals as a result of the advancement of new tools in the field of medicine. Many machine learning techniques such as KNN, Logistic regression, random forest etc…are used to predict life expectancy for post thoracic surgery.

About the scientific and biological things that happen inside a human body. This way it becomes far easier to work the technical advancements. Thoracic surgery is done when lungs stop working properly. In an elaborate way, lungs stop exchanging gasses which is obviously a death deal. Alveoli are the minute organs in lungs which are critical for exchange of gasses. When alveoli fades or dies the septal cells also become dead which in turn form a dead tissue what we generally call a Tumor. What makes alveoli die? Many things, especially tobacco. Tobacco contains Nicotine carcinoma which is a deadly component. Tumor that is responsible for lung cancer can be detected in CT scans which is a common way for detecting any kind of abnormality in humans.

That is why one of our 17 attributes is smoking criteria. Even though people are aware of how deadly cancer can be somehow they are always reckless and careless about taking care. Lung cancer became very obvious that today we are doing a project related to it as a development in technology.

Lung cancer cannot be cured but certainly can be prevented and avoided. The most prevalent cause of death after any sort of thoracic surgery is postoperative respiratory problems. The predictive models that are provided are based on various supervised machine learning techniques including logistic regression and Random forest as an aim to model cancer risk or patient outcomes.

1.2 SCOPE

Their results indicated that simple logistic regression technique is better or other machine learning techniques with 81% prediction accuracy. Notwithstanding, with the low post-surgery survival rate of lung cancer patients whether it is SCLC or NSCLC (e.g., Timmerman et al, 2016; Adam et al, 2019), many critical factors such as age, experience of the surgeon and patient medical condition, among other things, must be considered in determining the risk of operating on these patients.

Hence, a thorough diagnosis and analysis must be performed based on past historic patient data and current patient medical condition prior to recommending surgery. With respect to the prediction of post-thoracic life expectancy, there has been emerging work implementing ML techniques. Predictions from the use of these techniques are often good enough to assist the patient (and surgeon) in deciding to undergo surgery or not.

Given the limited cancer patient survival rate post-thoracic surgery, research has emerged in applying data mining techniques for its medical diagnosis and prediction (e.g., Nachev & Reapy, 2015). Models used included decision trees, Naïve Bayes (NB), artificial neural network (ANN) and support vector machines (SVM). More recently, Desuky & El Bakrawy (2016) have applied MLP, Naïve Bayes, J48, logistic regression (LR) for post-thoracic surgery life expectancy prediction on the UCI thoracic surgery dataset. Their work also involved attribute ranking and selection to achieve more accurate prediction. While only satisfactory accuracy has been achieved with traditional ML algorithms as well as with more recent ones such as Multi-Layer Perceptron (MLP), a type of Artificial Neural Network (ANN), and Bayesian model, no work on deep learning (DL) for post-thoracic life expectancy has yet been found.

Deep Neural Network (DNN) which is a part of Deep Learning (DL) is an advancement of ANN; if applied, it stands to achieve better accuracy with a reduced percentage of error vis-à-vis other ML algorithms (Geron, 2016). As it is touted to be superior in performance vis-à-vis other traditional ML algorithms such as LR, SVM, ANN, Decision Tree (DT), and Random Forest (RF), we hereby proposed to develop a DNN-based approach, the most advanced form of NNs in ML, to predict post-thoracic life expectancy. In this work, the thoracic surgery dataset was drawn from the Wroclaw Thoracic Surgery Centre ML repository, which contained 470 instances.

CHAPTER 2

LITERATURE SURVEY

REVIEW:

Thoracic surgery is considered the consummating operation being performed on carcinoma patients. Survival rate (kokulu, et al., 2015) is a key factor for surgeons to determine on which patient surgery would be beneficially performed.

Patient selection is one of the challenging factors in thoracic surgery decision, taking into account parameters to determine risk-benefit considerations for the patient both in the short-term (e.g. post-operative complications, including death-rate within the ðrst month) and long-term perspective (e.g. survival for 1-5 years).

In the last decades, different ML algorithms have been studied as well as evaluating attribute ranking and selection methods towards disease prognosis and prediction. Zieba, et al. (2014), for example, used “boosted SVM" to predict the postoperative life expectancy. These authors have solved the imbalanced data problem towards extracting the decision rules from boosted SVM by applying an “oracle-based” approach. Danjuma (2015) analyzed the performance of MLP vis-à-vis J48 and the NB algorithm on the UCI ML repository dataset for thoracic surgery.

From the analysis, MLP was found to perform the best with a classification accuracy of 82.3% vis-à-vis J48 and NB. Kourou, et al. (2015) evaluated predictive models based on various supervised ML techniques such as SVM, ANN, Bayesian networks, and DT with the aim to model cancer risk or patient outcomes. Notably, with the Bayesian model, an accuracy of 91.28% was achieved on the same UCI repository dataset from Wroclaw Thoracic Surgery Centre, Poland. To improve on ML techniques when the datasets have a large number of features or attributes, Desuky & El Bakrawy (2016) employed attribute ranking and selection to identify the most relevant attributes while removing those redundant and irrelevant attributes from the dataset.

All four (4) of their applied ML algorithms (SVM, LR, MLP, and J48) have also been compared with their boosted versions. Their results showed that boosting is not always the better choice. In another body of work, Sindhu, et al. (2014) used six (6) classification approaches, including NB, J48, Partial Decision Tree (PART), One R, Decision Stump (DS), and RF to analyze thoracic surgery data.

Nachev & Reapy (2015) studied the chance of patient survival after undergoing post thoracic surgery by applying data mining techniques for medical diagnosis. Models used included DT, NB, and SVM. Results showed that SVM is the most suited one vis-à-vis other models in term of accuracy.

More recently, with the mushrooming of ML algorithms, the call for better accuracy gained further attention. Kittipat, et al. (2018) have employed the Bayesian network model towards predicting post-thoracic surgery life expectancy as performed on the UCI thoracic surgery dataset.

Their experimental results unveiled an accuracy of 91.28% for the Bayesian model with discretization and learning scheme. Zhangheng, et al. (2020) have developed an artificial intelligence (AI) model for International Journal of Healthcare Information Systems and Informatics Volume 16 • Issue 4 4 predicting the life expectancy of post-thoracic surgery within a year period. This was done for NSCLC patients with bone metastases by employing the Extreme Gradient Boosting (XGBOOST) algorithm. XGBoost was further compared with SVM, RF, LR towards generating predictive models. XGBoost outperformed other models in terms of accuracy for training and validation with an accuracy of 78.6% being achieved for XGBOOST during validation vis-à-vis other models.

Altogether, ML algorithms clearly have prevailed for predicting post-thoracic surgery life expectancy. Many have even applied a boosted ML version and explored with various decision rules to handle data imbalance.

Importantly, where the dataset is high with different attributes and features, attribute ranking and selection are employed, and ML models such as SVM, LR, MLP, J48 and others with boosted versions are recommended to improve prediction accuracy (Desuky & El Bakrawy, 2016). To date, results have shown that ANN/MLP and Bayesian model appeared to have achieved the best classification accuracy of about 82.3% and 91.28% respectively (Danjuma, 2015; Kourou, et al., 2015; Kittipat, et al., 2018) Although the use of Bayesian models has often resulted in a higher accuracy, these models are based on directed acyclic graphs, which represent independent (and dependence) relationships between variables. The links in the model represent conditional relationships in the probabilistic sense.

Compared to DNN, these models are much simpler with no hidden layers, weights, biases, and activation function for producing the output. Also, there is no concept of back propagation to reduce the gradient loss.

With the current trend towards DNN, a part of deep learning, our focus here is on using DNN to predict the post-thoracic life expectancy of patients with superior accuracy and reduced error, thereby benefiting the healthcare industry.

A superior accuracy can be expected because DNN is an advanced form of NN with multiple hidden layers. Thus, results from this work would especially benefit patients and hospital management.

2.1 RELATED WORK

Even though people are aware of how deadly cancer can be somehow they are always reckless and careless about taking care. Lung cancer became very obvious that today we are doing a project related to it as a development in technology. Lung cancer cannot be cured but certainly can be prevented and avoided. The most prevalent cause of death after any sort of thoracic surgery is postoperative respiratory problems.

The predictive models that are provided are based on various supervised machine learning techniques including logistic regression and Random forest as an aim to model cancer risk or patient outcomes. Their results indicated that simple logistic regression technique is better or other machine learning techniques with 81% prediction accuracy.

Related works and previous works had made ways for a lot of advancements in lung cancer predictions which has led us to these great inventions. By this technological development we can estimate the life span of certain patients.

Machine learning is an advancement in coding which makes pretty much everything easier. Machine learning is basically written in Python. We use a jupyter notebook or jupyter lab for implementation. In machine learning there are three types of techniques named as Supervised, Unsupervised and semi supervised.

There are different algorithms in each of these categories.

To name a few Logistic Regression, SVM, decision trees, Random Forest, KNN, K-Means ,Naive Bayes theorem etc. As per our requirements we choose algorithms. We compare different algorithms and stick with the one that gives the highest accuracy.

2.2 DATA COLLECTION

The data set is from the UCI Machine Learning Repository. According to the main repository site, the data was collected retrospectively at Wroclaw Thoracic Surgery Center for primary lung cancer patients in 2007-2011. The biomedical data used consisted of 470 samples and were based on classification issues related to postoperative life expectancy in lung cancer patients. It consists of two classes: death or survival within one year after surgery.

There were 70 death class data and 400 survival class data for one year after surgery. In this work, the data is divided into training and testing. Based on clinical studies it was identified that there are many risks to patients after thoracic surgery, most common contributing factors are age, preoperative pulmonary function test, cardiovascular comorbidities, chronic obstructive pulmonary disease, and smoking status .

In this dataset, features are indicative of lung cancer patients.

Preprocessing Data

Preprocessing is a step used in data mining to transform raw data into a form that is easy to understand so that it can represent data effectively so that there is not much excessive information that is not related or noisy . Data preprocessing in this research consists of transforming the data from nominal and binary to numeric so that the algorithm can process it. Outliers were removed after data transformation. Outliers are data that are significantly different and do not correspond to the normal behavior of the data . There are 16 outliers in the data, and after removing these outliers the new data set contains 454 instances from the original 470. This data does not have such problems as missing values and duplicate data. The research uses split data with a proportion of 60:40, 70:30, 80:20, and 90:10.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

It has always been a difficult task to accurately predict the life expectancy post an operation. The prediction depends on several health factors of which have a much crucial role compared to the other factors.++

A popular method used in the past was to analyze the CT scan images of the lungs and predict based on the regular check-up.

In existing methods we made predictions about life expectancy but that was not accurate. Some cases were so out of prediction that they led to false hope for patients. But now that has been covered with new advancements.

Confusion matrix obtained: In attributes we take there are many dimensions in a single data set. For all the features we do feature scaling to avoid or omit outliers. After feature scaling, with the features we obtained we interpret graphs by comparing every feature with every other feature.

Then we do a confusion matrix based on these. This is a common method to assign numerical for attributes to nullify the blanks in a data set..

3.2 PROPOSED SYSTEM

In order to improve quality initiatives, healthcare administration, and consumer education, it is critical to track health outcomes. Since the data is huge and is complex for the model to handle we remove certain attributes which are irrelevant for the prediction. The most relevant attributes are identified using attribute ranking and selection, and the duplicated and unnecessary attributes are removed from the dataset. The goal of our model is to look at patient mortality over the course of a year after surgery.

More precisely, we're looking into the patients' underlying health factors, which could be a powerful predictor of surgical-related mortality. In existing methods we made a lot of progress in obtaining more accuracy about predictions of life expectancy.

We not only eliminate unwanted attributes but consider mean values of features. Mean values are more accurate than singular values as they omit outliers. This is what made this method different from the previous method.

Patients undergoing thoracic surgical medical conditions believe that the medication would improve their lifestyle so that they can lead a longer and peaceful life.

But it is highly challenging to monitor the survival rate of patients within a year’s time post-thoracic surgery.

If a pattern exists in the patient dataset pertaining to age, health condition, and other parameters, it would be beneficial in predicting the life expectancy of the patient within a year period.

This prediction would be really helpful for the surgeons and the patients in making a more informed decision on whether they should go ahead with performing the surgery or if they would like to pursue palliative care or some other alternative treatments.

This information could also be used by clinical researchers to consolidate any useful findings with other research findings to uncover new discoveries.

With the advent of DL, a gap exists as no work has yet been reported in predicting life expectancy of post thoracic surgery patients via DNN. DNN is an advanced form of neural network and a subset of DL.

A highly accurate predictive model based on the 17 attributes pertaining to thoracic surgery for life expectancy within a year period can thus be expected with the application of DNN vis-à-vis the more traditional ML techniques .

CHAPTER 4

SYSTEM REQUIREMENTS

4.1 HARDWARE REQUIREMNETS

The hardware requirements for the ONE YEAR LIFE EXPECTANCY POST THORACIC SURGERY USING IBM WATSON STUDIO project depend on the complexity of the models and the size of the dataset. Here are some general hardware requirements: CPU: A modern multicore processor (e.g., Intel Core i5 or higher) is sufficient for small-scale experimentation. GPU (optional): Deep learning models can benefit significantly from GPU acceleration, especially when training larger models or performing extensive experiments. NVIDIA GPUs, such as the GeForce RTX series, are commonly used for deep learning tasks. RAM: At least 8 GB of RAM is recommended, although more may be required for larger datasets and models. Storage: Sufficient disk space to store the dataset, models, and intermediate results. SSD storage is preferred for faster read/write operations.

These hardware requirements can vary based on the specific project scope and computational demands. It is advisable to assess the hardware requirements based on the scale of the project and available resources.

- **System : Intel CORE i5**

- **Hard Disk : 2 TB**

- **Screen : 15 VGA Color.**

- **Ram : 16GB.**

4.2 SOFTWARE REQUIREMENTS

The software requirement specification outlines the software tools and libraries needed to implement the project. For the ONE YEAR LIFE EXPECTANCY POST THORACIC SURGERY USING IBM
WATSON STUDIO project, the following software requirements may be considered:

Programming Language: Python Deep Learning Frameworks: TensorFlow, Keras Data Manipulation and Analysis: NumPy, Pandas Visualization: Matplotlib

- **Operating System - Windows 11**

- **Database - CSV File, TSV File.**

- **Programming Language - Python**

- **IDE - Jupiter, Python 3.7,Google colab**

CHAPTER 5

SYSTEM ARCHITECTURE

ALGORITHMS:

5.1 Popular ML Algorithms:

Using traditional ML algorithms such as LR, RF classifier, SVM, KNN, and NB, the thoracic surgery dataset comprises labeled training data, which are categorized as binary classification. Here, we implement a DNN for the life expectancy prediction problem for post thoracic surgery with the following considerations

5.1.1 Support Vector Machine (SVM) SVM is used predominantly for classification and regression (Gandhi, 2018; Geron, 2017). Figure 2 shows the SVM algorithm being plotted as points in space.

Figure 2.

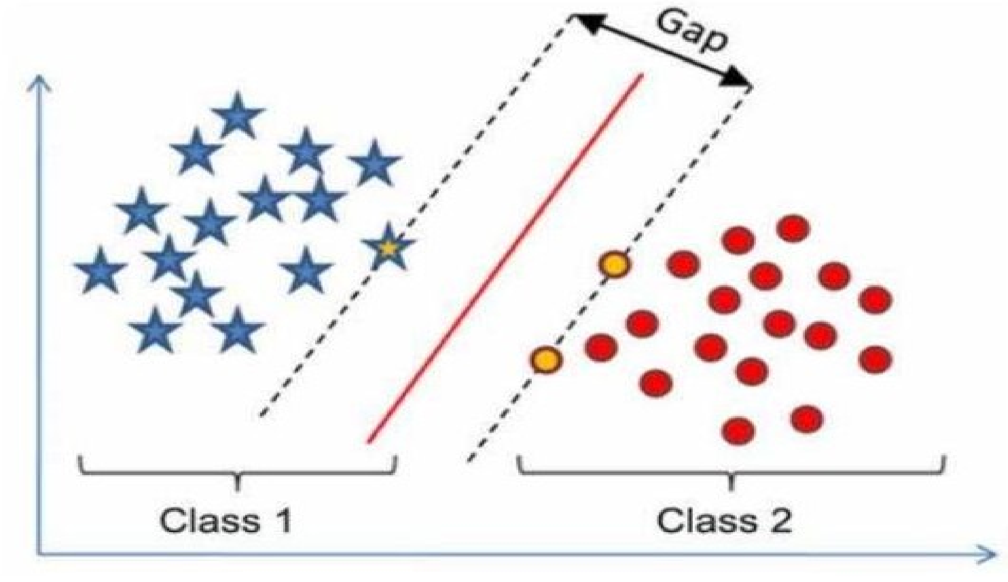
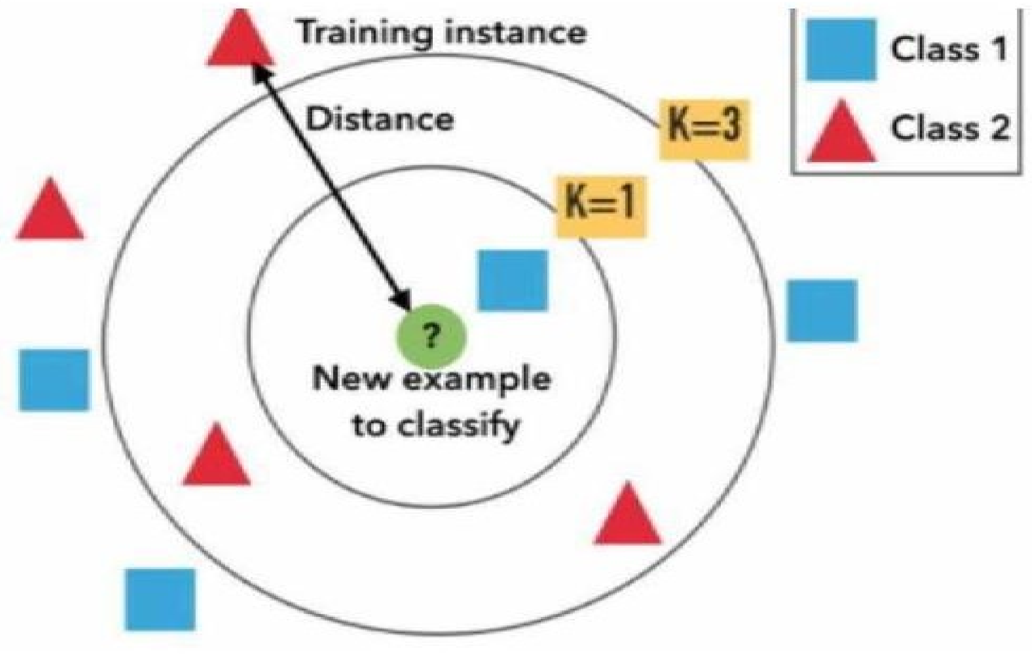


fig 1

SVM model is generally designated to one class or towards developing a binary or probabilistic linear classifier. In this method, each example belongs to one class or the other which are divided by visible space or gap. A line, known as a hyper plane, divides or demarcates the SVM algorithm during classification. A hyper plane is a line that splits the dataset into two halves where each stores the data from two previously established classes. The construction of the hyper plane is carried out in this algorithm where the classification of new values is constructed. After applying the SVM by constructing a hyper plane on a given dataset, data gets classified into different classes. Based on this classification, the prediction accuracy is then computed. 5.2K-Nearest Neighbors (KNN) As shown in Figure 3, KNN is one of the most non-parametric algorithms used for regression and classification (Geron, 2017; Subramanian, 2019).

In nearest neighbors, a particular number of samples that are closer in distance to new points are found, and from that basis, the labels are predicted. The number of neighbors or samples are user defined. In KNN, distance is calculated which can be of any metric and employs the most commonly used Euclidean distance as given in Equations 1 and 2 below

Figure 3.



$$d(p,q) = d(q,p)$$

fig 2

simply stores the instance of training data only. Classification in this algorithm is performed by computing votes of the nearest neighbor of each point. The output is the average values of its neighbors present around it.

The major problem in the KNN algorithm is choosing the "K" value. The main drawback is the difficulty in finding the number of nearest neighbors for each sample **3.1.3 Naïve Bayes (NB)** : Bayes' theorem is the basis of the NB technique, which assumes that the predictors are independent (Geron, 2017; Brownlee, 2019). It further assumes that there are no features that are related to one another; simply, the features are completely independent. The mathematical statement of the "Bayes' theorem" is as follows:

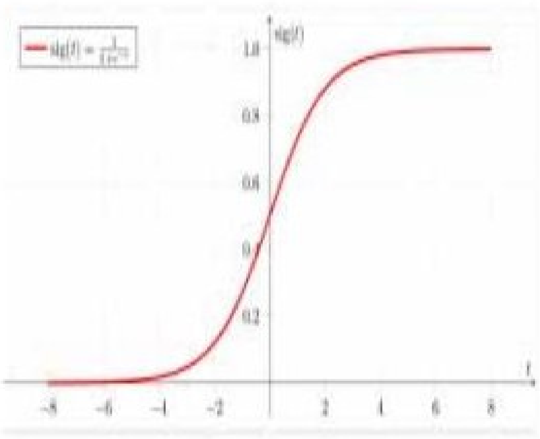
P(A│B) =P(B│A)*P(B)/P(A)

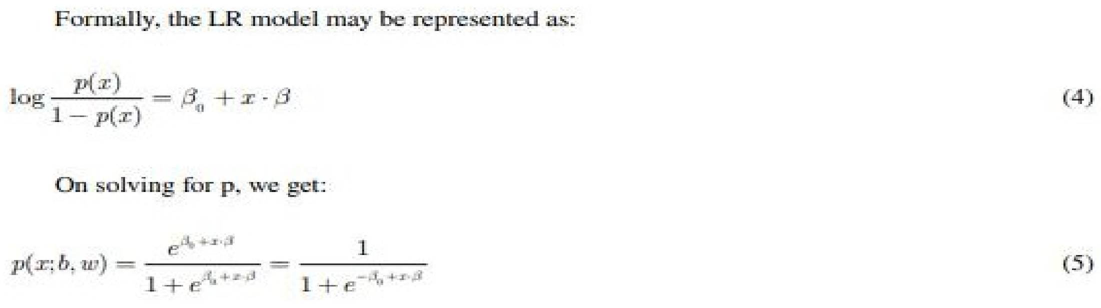
(3) Here, P(A) is the prior probability of event A and P(A|B) refers to event A’s probability after seeing the evidence.

This model is easy to build and may be used with large datasets. The first step in this algorithm is to convert the data set into a frequency table and after creating a likelihood or frequency table, the Naive Bayesian formula is applied to calculate the probability of each class. 3.1.4 Logistic Regression (LR) LR is a classification algorithm where observations are assigned to a discrete set of classes (Geron, 2017; Swaminathan, 2018).

As shown in Figure 4, the regression model is built to predict the probability where a given datum belongs to the category number as “1”. Only when a “decision threshold” is considered and brought into the picture, LR becomes a classification technique

Figure 4.





Formally, the LR model may be represented as:

$$\log \frac{p(x)}{1-p(x)} = \beta_0 + x \cdot \beta \qquad (4)$$

On solving for p, we get:

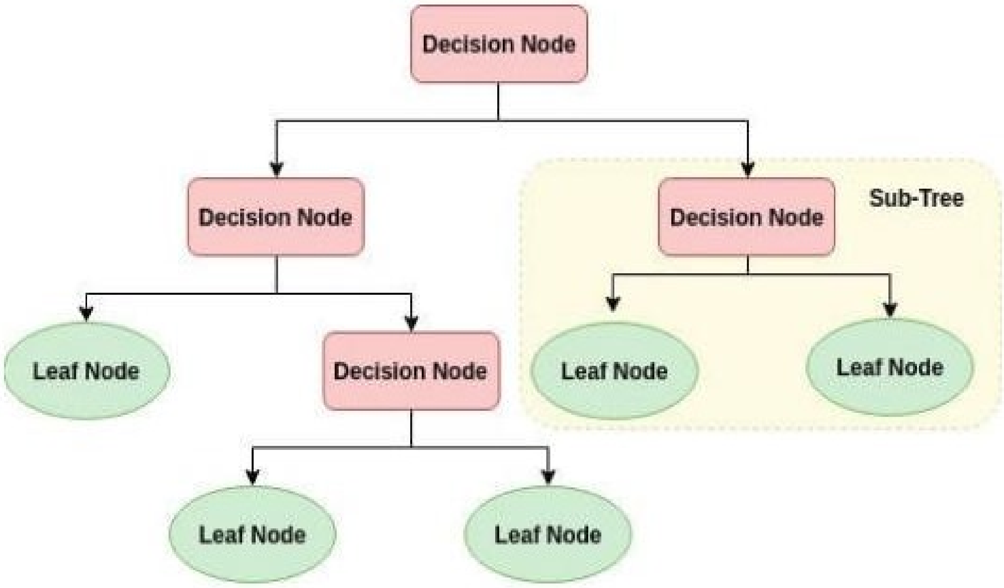
$$p(x; b, w) = \frac{e^{\beta_0 + x\cdot\beta}}{1 + e^{\beta_0 + x\cdot\beta}} = \frac{1}{1 + e^{-\beta_0 + x\cdot\beta}} \qquad (5)$$

Fixing the threshold value is the most important one in logistic regression and is dependent on the “classification” problem.

5.2 DECISION TREE (DT) :

A set of the axis-parallel hyperplane dividing the region into a hypercube, the DT is based on the nested if-else classifier (Geron, 2017; Gupta, 2017). It is a classification or regression model in the form of a tree structure being built via the decision trees. As shown in Figure 5, DT can handle both categorical as well as numerical data

Figure 5.



Given that the dataset is broken down into smaller subsets with increasing tree depth, the final result is a tree with decision nodes and leaf nodes. In a DT, the root node, which refers to the best predictor, is at the topmost. Two or more branches are created via a decision node whereas the classification (decision) is denoted by the leaf node.

- Construction of DT

Step 1: First, we calculate the entropy of the target

Step 2: Based on different attributes, the dataset is split; following that, entropy is assessed and added proportionally to compute the total entropy for the given split. The resulting entropy assessed is subtracted from the entropy before the split, resulting in information gain

Step 3: Then, we choose the decision node, which divides the dataset by its branches and repeats the same process on every branch.

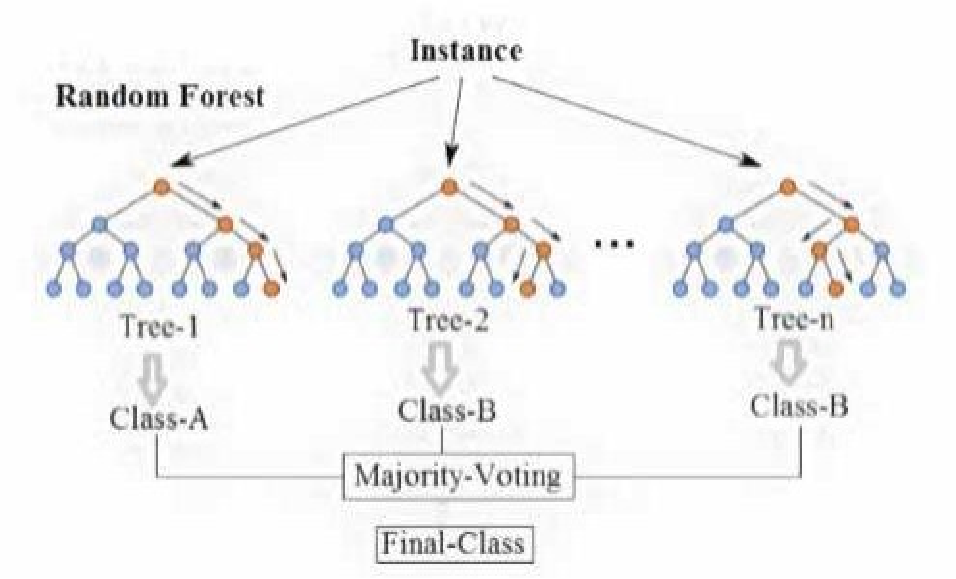
Step 4a: A branch with an entropy of 0 is noted as a leaf node

Step 4b: A branch with an entropy of more than 0 requires further splitting.

5.3 RANDOM FOREST (RF)

As shown in Figure 6, RF is a classification algorithm comprising multiple decision trees.

Figure 6.



RF uses bagging and features randomness while building every individual tree to create an uncorrelated forest of trees. The RF technique does both row sampling and column sampling with DT as a base. Owing to column sampling, model h1, h2, h3, h4 are more different than by doing only bagging.

With an increasing number of base learners, variance decreases; also, as the value of k decreases, there will be an increase in the variance.

For the entire process, bias remains constant. The value of "k" can be found using the cross-validation technique; in this method, low bias and high variance are needed for our base learner.

• Steps for implementing a RF classifier are as follows:

1. Consider a training data set of N observations and M features. A sample of data is taken from the training data set randomly with replacement;

2. Next, the subset of M features is chosen randomly; accordingly, the feature with the best split is used for splitting the node sequentially;

3. The tree grows as large as possible;

4. Repeat Steps 1 to 3; further, the prediction is performed in line with the aggregation of predictions from the multiple numbers of trees.

• Train and run-time complexity:

Training time = O (log(nd)*k) Run time = O (depth*k) Space = O (store each DT*K)

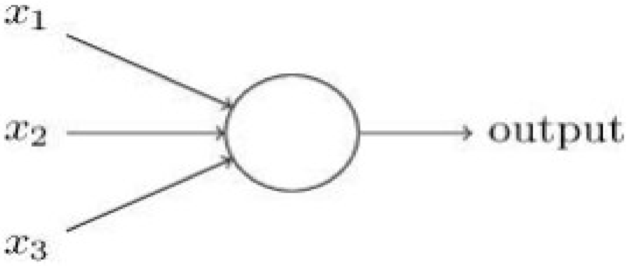
As the number of base models increases, training run time increases with an increasing number of the base model. Hence, the use of cross-validation to find the optimal hyperparameter is recommended.

5.4 DEEP NEURAL NETWORK (DNN)

A perceptron is also known as an artificial neuron forming the neural system (Geron, 2017; Allibhai, 2018). As shown in Figure 7, x1, x2, x3 are given as inputs to the perceptron, which produces a single binary output. Algebraically, that is everything as to how a perceptron

Functions.

4 **Figure 7.**

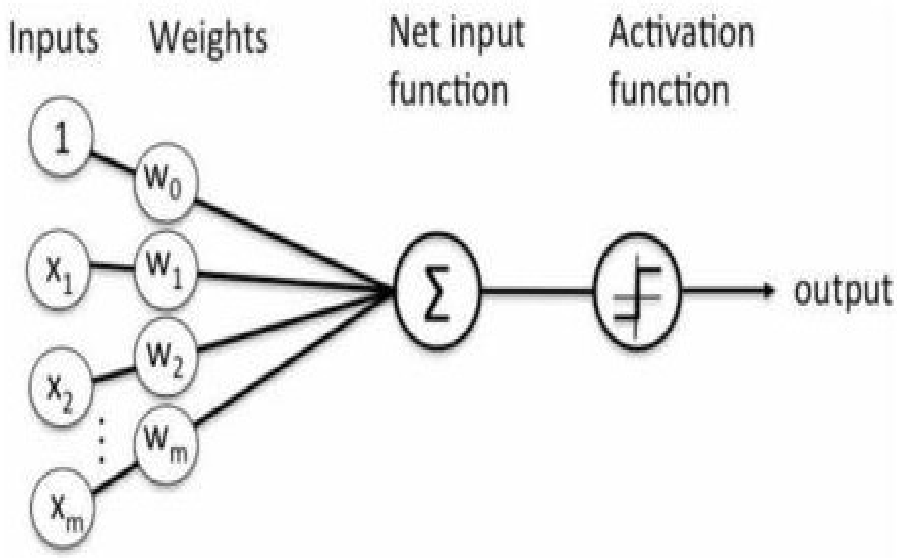


The functioning of the human brain is imitated by employing neural network (NN) technology to uncover pattern recognition rather than passing the input through the different layers of the simulated neural connection.

$$output = \begin{cases} 0 & if \Sigma_j w_j x_j \leq \quad threshold \\ 1 & if \Sigma_j w_j x_j > \quad threshold \end{cases}$$

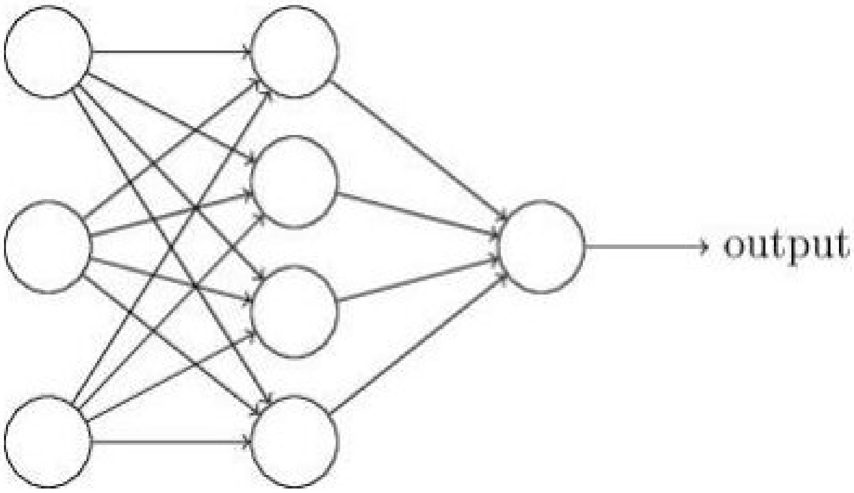
ANN have an input layer, at least one hidden layer in-between and an output layer. In feature hierarchy, specific sorting and ordering types are carried out in each layer. To deal with unlabelled or unstructured data is among the significant uses of these NNs. Figure 8 shows the Perceptron in ANN.

Figure 8.



Assuming we have the network as shown in Figure 9.

Figure 9.



Hierarchical composition of linear v. non-linear activation function is given by DNN (Brownlee, 2018). We use DNN, which is a subset of DL here, comprising an input layer, two hidden layers, and a final output layer.

The former layers will be activated via function, ReLu, while the output layer will be activated via function, Sigmoid. 3.2 System Architecture Figure 10 shows the dataflow of post-thoracic surgery life expectancy system and its interaction with the patient, surgeon, and the hospital system. It shows how the prediction model plays a very crucial role in making a decision towards surgery for patients.

The data flow diagram depicts a ML-enabled post-thoracic life expectancy prediction system being integrated into the hospital system.

Here, the patient makes an appointment for thoracic surgery with the patient's medical data being fed into the ML-based prediction system to forecast the life expectancy within a year after surgery.

Based on the analysis, the surgeon advises whether to perform the surgery or not. The information is then passed on to the patient for treatment, billing, and reports.

Notably, the surgeon's recommendation is also passed onto the hospital and stored in the hospital cloud as part of a dataset for further research.

This is the key role played by the ML-enabled prediction system in interacting between the patient and surgeon prior to a surgery decision.

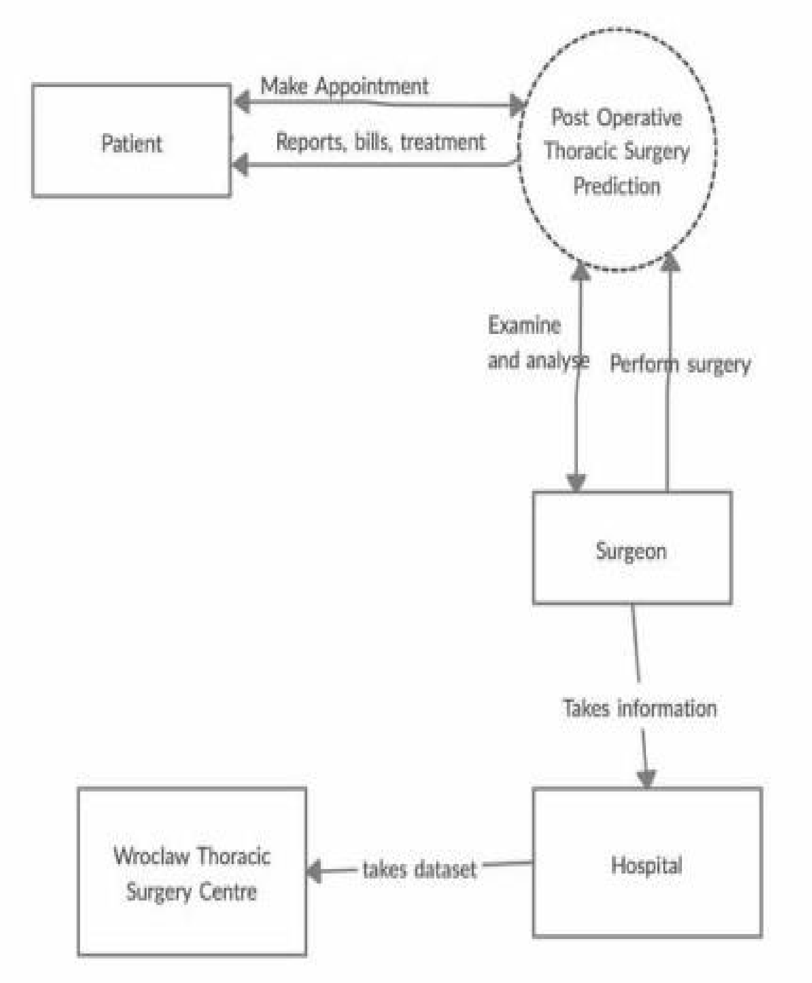
Also, data collected are stored in the cloud for continuously training and updating the model for better prediction.

Use case diagrams of the system developed are now shown in Figure 11. Four (4) key actors: the nurse, the surgeon, the patient, and the lab technician are involved in this use case. First, the patient interacts with the nurse for a thoracic surgery appointment.

Accordingly, the nurse makes the appointment with the surgeon and advises the patient for the respective tests and referral to the assigned surgeon for the consultation.

The patient here gives the test samples which are collected by the lab technician. Next, the surgeon examines the patient, analyses the symptoms, together with the test results and feeds the relevant information into the prediction system. Based on the output from the prediction system, surgery is ultimately decided, followed by having to prepare the patient for surgery.

Figure 10.

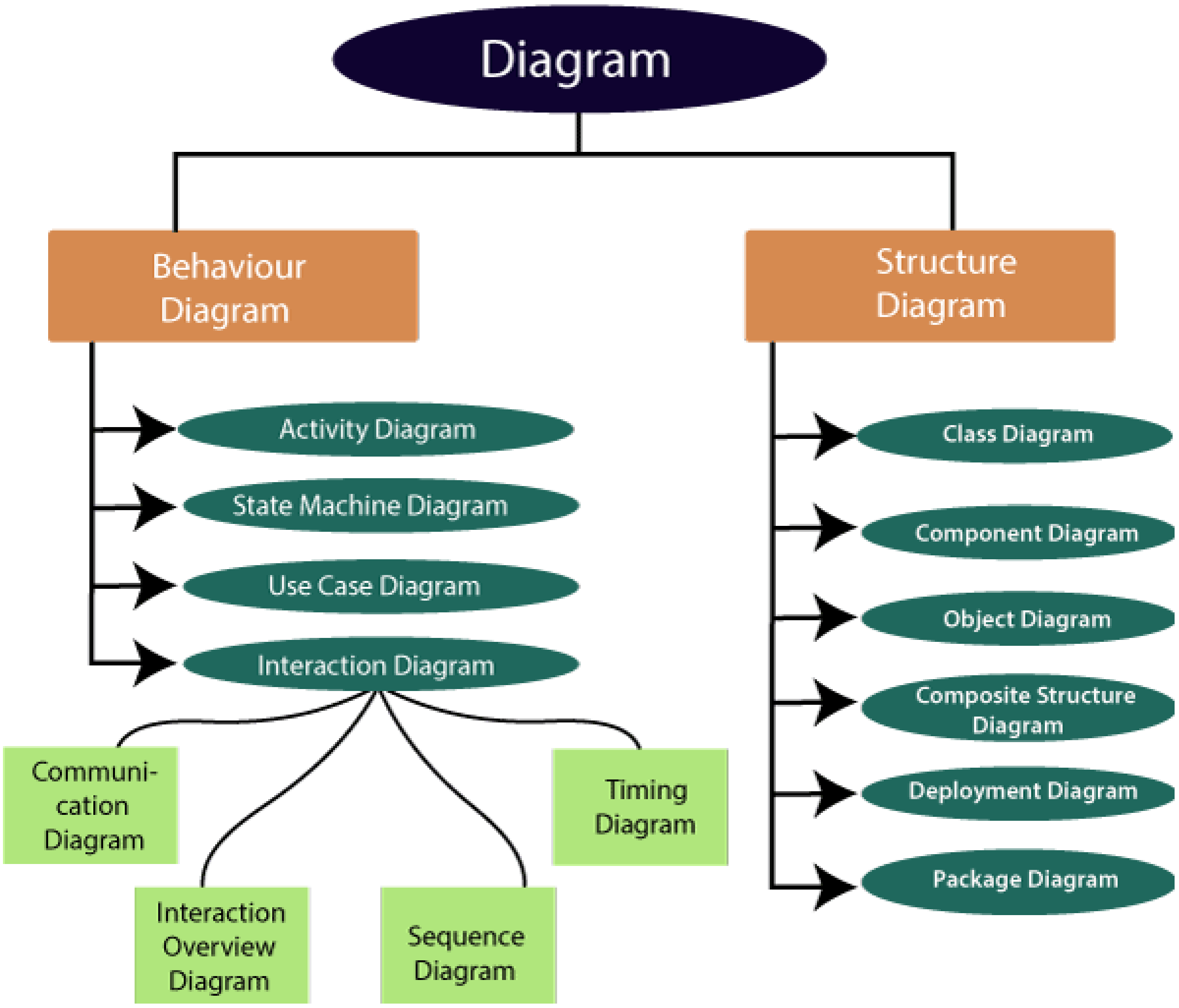


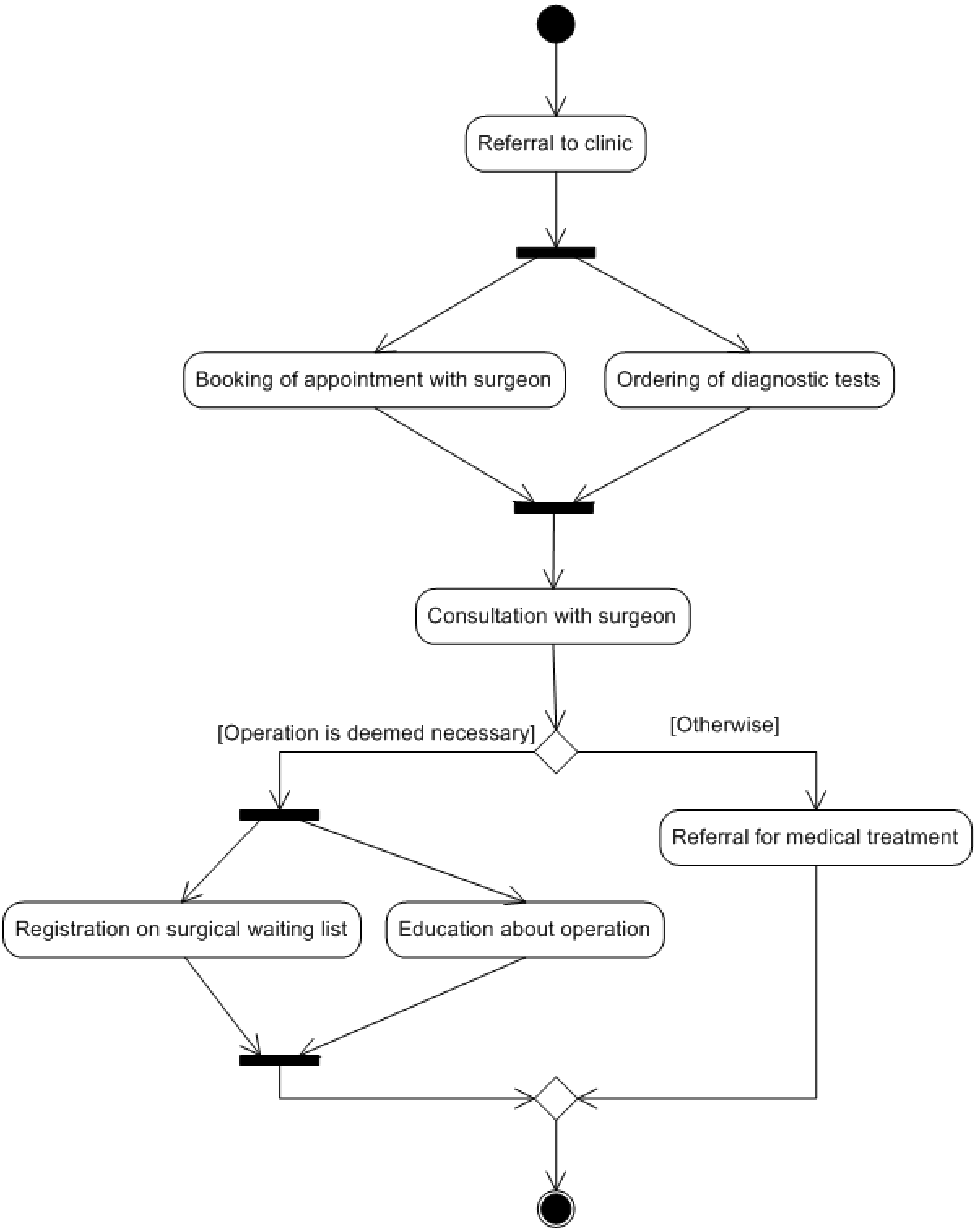
In summary, the prediction system becomes an implicit part of the use case diagram between the surgeon and patient in deciding on the surgery based on post-thoracic surgery life expectancy output vis-à-vis the test results and symptoms.

CHAPTER 6
UML DIAGRAMS

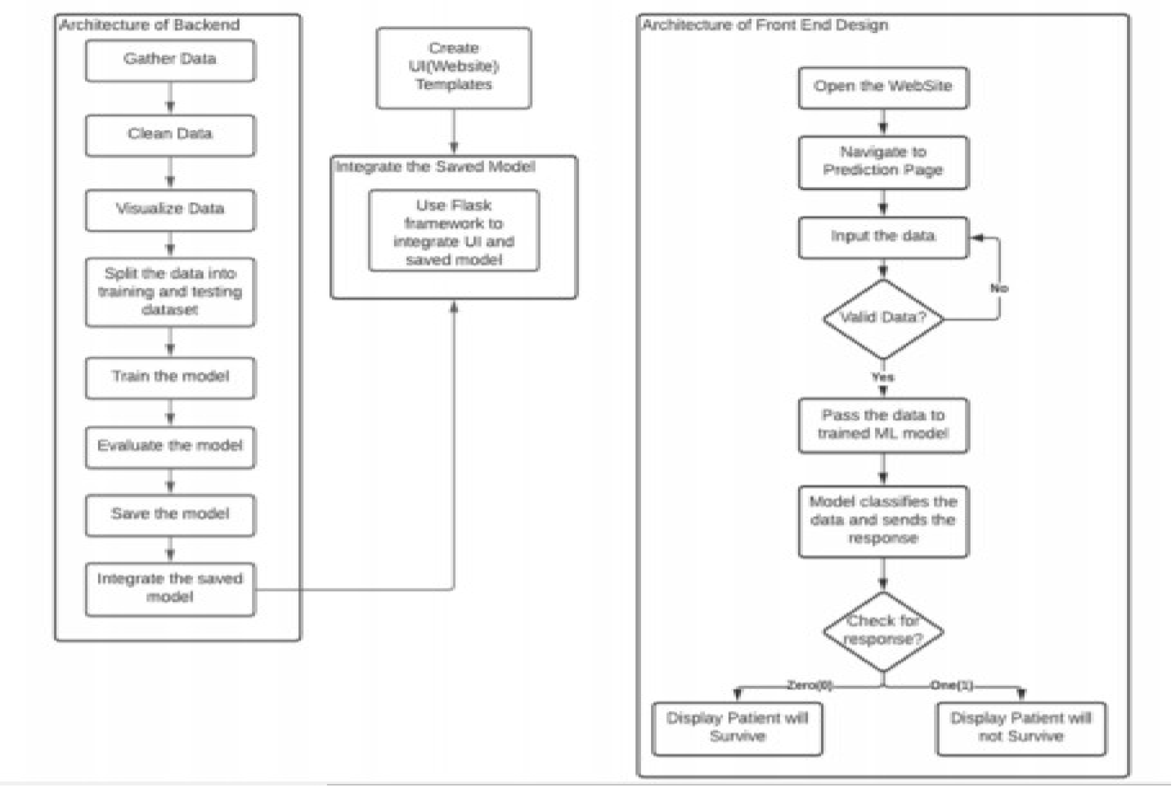
6.1 STRUCTURAL AND BEHAVIORAL

The UML diagrams are categorized into structural diagrams, behavioral diagrams, and also interaction overview diagrams. The diagrams are hierarchically classified in the following figure:




Referral to clinic
Booking of appointment with surgeon
Ordering of diagnostic tests
Consultation with surgeon
[Operation is deemed necessary]
[Otherwise]
Referral for medical treatment
Registration on surgical waiting list
Education about operation

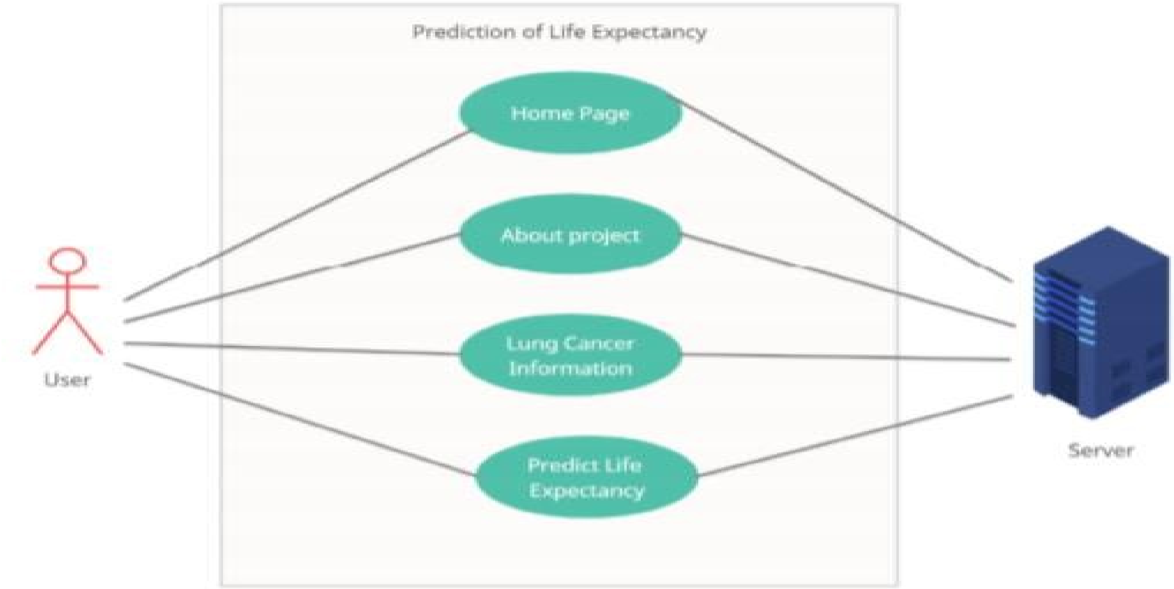
6.2 SYSTEM DESIGN:


Architecture of Backend
Gather Data
Clean Data
Visualize Data
Split the data into training and testing dataset
Train the model
Evaluate the model
Save the model
Integrate the saved model
Create UI(Website) Templates
Integrate the Saved Model
Use Flask framework to integrate UI and saved model
Architecture of Front End Design
Open the WebSite
Navigate to Prediction Page
Input the data
Valid Data?
No
Yes
Pass the data to trained ML model
Model classifies the data and sends the response
Check for response?
Zero(0)
One(1)
Display Patient will Survive
Display Patient will not Survive

System Architecture In the above diagram specifies System Architecture of the project where this diagram contains two actors, one is front end and the other is backend. First we should gather the data from the data sets and we should clean the null values from the data set then we should visualize the data. After visualizing the data we should split the data into two parts for testing and for training the dataset after the should train the model again should evaluate the model again should save model the saved model should be integrate by using the flask framework to integrate UI.In the backend at the
end of the program if the code is correct we will get one website link after open we come to the frontend in the we will be having the introduction and also it navigates to the prediction page in the we should insert the data if the data is valid it pass the data to the trained model if not again we should enter the data .If the data is valid the model classifies the data and sends the response from that response (zero , one)zero means patient will survive , one means patient will not survive All these controls using the
Local server without using the Internet. The control speed is very fast as everything is happening locally.

6.3 USE CASE DIAGRAM

The graphical results of a possible user interact with a system can be a case diagram. Different usage cases and different types of users are shown in the use case scheme. The various types of diagram are also often accompanied. Either circles or ellipses can be seen in the case of use. The performer and often as stick figures. Simplest user
interaction of object representation where the system shows difference between the user and the different use cases where the user and system was involved.



The above figure specifies the Use Case diagram of the system where the diagram mainly contains two actors, one is User and the other is the Server. Both the actors can access all the functionalities provided by the system. We have user at the left and
server at the right, In between we have four vowels that is Home page, About project, Lung Cancer Information, Predict Life Expectancy. Here each and every vowels are connected to both User and Server.

CHAPTER 7

DESIGNING PHASE

1. **Project Objective and Scope:** Define the specific objectives of your thoracic surgery project. What do you aim to achieve? Is it a research project, a clinical study, or a quality improvement initiative? Determine the scope of your project and its significance in the field of thoracic surgery.
2. **Team Formation:** Assemble a multidisciplinary team with expertise in thoracic surgery, anesthesia, nursing, and any other relevant areas. Clearly define roles and responsibilities for each team member.
3. **Literature Review:** Conduct a comprehensive literature review to understand the current state of knowledge in your chosen area of thoracic surgery. Identify gaps in the literature that your project can address.
4. **Patient Selection and Informed Consent:** If your project involves patients, establish criteria for patient selection. Ensure that ethical considerations are met, and obtain informed consent from patients or their legal guardians if applicable.
5. **Data Collection and Management:** Determine the data you need to collect for your project. This may include patient records, imaging studies, surgical notes, and follow-up data. Create a data collection plan and establish secure and compliant data management procedures.
6. **Methodology:** Describe the methodology you will use for your project. If it's a clinical study, specify the study design (e.g., retrospective, prospective, randomized controlled trial), data collection instruments, and statistical analysis plan.
7. **Ethical and Regulatory Approvals:** Ensure that your project complies with ethical standards and regulations. Seek approval from the Institutional Review Board (IRB) or an ethics committee as necessary.
8. **Resource Planning:** Identify the resources required for your project, including personnel, equipment, facilities, and funding. Develop a budget and timeline to manage these resources efficiently.
9. **Risk Assessment and Mitigation:** Identify potential risks and challenges that could arise during the project and develop strategies to mitigate them. This includes risks related to patient safety, data integrity, and logistical issues.
10. **Communication and Collaboration:** Establish communication channels within your team and with external stakeholders. Collaboration with other healthcare professionals, researchers, and institutions may be necessary for a successful project.
11. **Data Analysis and Interpretation:** Outline how you will analyse the collected data and interpret the results. Consider using statistical software, and plan for data quality assurance and validation.

12. **Reporting and Dissemination:** Determine how you will report and disseminate the findings of your project. This may involve writing research papers, presenting at conferences, or sharing results with the medical community.
13. **Quality Assurance and Monitoring:** Implement quality assurance measures throughout the project to ensure data accuracy and patient safety. Monitor progress regularly and make adjustments as needed.
14. **Documentation and Record-Keeping:** Maintain thorough documentation of all project activities, including data collection, analysis, and correspondence. This documentation is critical for transparency and future reference.
15. **Project Evaluation:** After completing the project, evaluate its success based on the defined objectives. Assess what worked well and what could be improved for future projects.
16. **Knowledge Transfer:** Share your project's findings and lessons learned with the medical community, both locally and globally, to contribute to the advancement of thoracic surgery knowledge.

CHAPTER 8

IMPLEMENTATION

8.1 FEATURE SELECTION

Collect Data: First, gather a comprehensive dataset that includes various patient characteristics, medical history, surgical details, and post-operative information. Ensure that the data is clean and well-organized.

Define the Target Variable: Clearly define the target variable, which is life expectancy in this case. It should be a measurable and relevant metric, such as the number of years a patient survived after the surgery.

Feature Engineering: Before starting feature selection, perform feature engineering to create new features or transform existing ones that may capture relevant information. This could include creating age groups, BMI categories, or comorbidity indices.

Exploratory Data Analysis (EDA): Conduct thorough EDA to understand the data distribution, identify outliers, and visualize relationships between variables. This will help you get insights into which features may be important.

Correlation Analysis: Calculate correlation coefficients (e.g., Pearson, Spearman) between each feature and the target variable. Features with high absolute correlation values are likely to be more relevant.

Feature Importance from Models: Train initial predictive models (e.g., decision trees, random forests, or gradient boosting) to assess feature importance. Many machine learning libraries provide built-in tools to rank features by importance.

Univariate Feature Selection: Utilize statistical tests (e.g., chi-squared, ANOVA) to assess the relationship between each feature and the target variable. Select features that show statistically significant associations.

Recursive Feature Elimination (RFE): Implement RFE algorithms to recursively remove the least important features from the dataset. This process continues until a desired number of features is reached.

Regularization Techniques: Apply L1 (Lasso) or L2 (Ridge) regularization to linear models to encourage sparsity in the feature space. Features with zero coefficients can be considered unimportant.

Domain Knowledge: Consult with domain experts, such as thoracic surgeons or medical researchers, to validate the importance of certain features based on their expertise.

Cross-Validation: Throughout the feature selection process, use cross-validation to ensure that the selected features generalize well to unseen data.

Feature Selection Tools: Utilize machine learning libraries and feature selection tools like scikit-learn in Python, which offer various methods for feature selection and ranking.

Evaluate Model Performance: After feature selection, build predictive models using the selected features and evaluate their performance using appropriate metrics.

Fine-Tuning: If necessary, revisit feature selection and model building iteratively to optimize model performance.

8.2 SOURCE CODE

```
import numpy as np
import pandas as pd
import seaborn as sns
from scipy import stats
import matplotl.pyplot as plt
import plotly.express as px
from plotly.offline import init_notebook_mode
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error,mean_absolute_error,r2_score
%matplotlib inline
```

```
data = pd.read_csv("C:\\Users\\dell\\OneDrive\\Desktop\\ThoraricSurgery.csv")
data.head()
data.info()
data.tail()
data.isna().sum()
data.describe()
```

```
# Compute correlation only on numeric columns
numeric_data = data.select_dtypes(include='number')
correlation_matrix = numeric_data.corr()
correlation_matrix

encoded_data = pd.get_dummies(data)
sns.heatmap(encoded_data.corr(), cmap='coolwarm')
plt.show()
```

```
data[['PRE7', 'PRE8', 'PRE9', 'PRE10', 'PRE11', 'PRE17', 'PRE19', 'PRE25',
'PRE30', 'PRE32']] = data[['PRE7', 'PRE8', 'PRE9', 'PRE10', 'PRE11',
'PRE17', 'PRE19', 'PRE25', 'PRE30','PRE32']].apply(lambda x: np.where(x ==
'T', 1, 0))
#Percentage of the Risk of 1 year survival period¶
data.groupby('Risk1Yr').size().plot(kind='pie', textprops={'fontsize':
20},autopct='%1.0f%%')
plt.title('Percentage of the Risk of 1 year survival period - (T)rue value
if died (T,F)')
plt.show()

#Observation
#85% of the patients were not survivied within the 1 year of survival period
data['DGN'].value_counts()
sns.set_style(style="whitegrid")
fig, ax = plt.subplots(figsize=(8,8))
ax.set_title("Type of DGN in all patients")

sns.set(font_scale=2)
sns.countplot(x='DGN',data=data)
plt.show()
#Observation
#DGN3 is the most common code that was present in all patients.
#As you know smoking is bad for your health. Let's see if it had an effect on patients.
fig, ax = plt.subplots(figsize=(15,7))

data.groupby('Risk1Yr')['PRE30'].value_counts().plot(ax=ax, kind='bar',
 title = 'Bar chart of Risk by Smoking', colormap = 'coolwarm') ax.set(xlabel = "(Risk1Yr, Smoker)")
plt.show()

 #Observation
#This shows how many patients survived 1 year being a smoker (F,1)
#This shows how many patients survived 1 year without being a smoker (F,0) #This shows how many patients
did not survive 1 year being a smoker (T,1) #This shows how many patients did not survive 1 year without being
a smoker (T,0)
#Type of OC in all patients
# this plot shows count of a given OC in all patients
sns.set_style(style="whitegrid")
fig, ax = plt.subplots(figsize=(8,8)) ax.set_title("Type of OC in all
patients")

sns.set(font_scale=2)   sns.countplot(x= 'PRE14', data =
data) ax.set_xlabel('OC type')

plt.show() sns.set_style(style="whitegrid")   fig, ax =
plt.subplots(figsize=(8,8)) sns.set(font_scale=2)
ax.set_title("Type of OC in patients who didn't survive first year after surgery")
sns.countplot(x= 'PRE14',  data =data[data['Risk1Yr'] == 'T']) ax.set_xlabel('OC type')
plt.show()
```

```
sns.set_style(style="whitegrid")
fig, ax = plt.subplots(figsize=(8,8))
sns.set(font_scale=2)
ax.set_title("Type of OC in patients who didn't survive first year after
surgery")
sns.countplot(x= 'PRE14',  data =data[data['Risk1Yr'] == 'T'])
ax.set_xlabel('OC type')

plt.show()
```

```
data['DGN'].value_counts()
data['PRE6'].value_counts()
data['PRE14'].value_counts()
data=data.drop('id',axis=1)
data.DGN=data.DGN.replace({"DGN3":3,"DGN2":2,"DGN4":4,"DGN5":5,"DGN6":0,"DGN8 ":8,"DGN1":1})
data.PRE6=data.PRE6.replace({"PRZ1":1,"PRZ0":0,"PRZ2":2})
data.PRE14=data.PRE14.replace({"OC12":2,"OC11":1,"OC13":3,"OC14":4})
data.head()
#Creation of a Model
#Take the X and y value
x=data.drop('Risk1Yr',axis=1)
x
y=data.Risk1Yr
y
y.value_counts()
#Import the train_test_split from the sklearn
from sklearn.model_selection import train_test_split
 #Split the Training Dataset and Test Dataset
X_train,X_test,y_train,y_test=train_test_split(x,y,test_size=0.2)
#Import the RandomForestClassifier from the sklearn
from sklearn.ensemble import RandomForestClassifier
model=RandomForestClassifier()
#Fitting the training data to the model
model.fit(x,y)
Prediction
y_predict=model.predict(X_test)
y_predict
#Accuracy
#Import the accuracy_score from the sklearn
from sklearn.metrics import accuracy_score
accuracy_score(y_test, y_predict)*100
from sklearn.metrics import confusion_matrix
performance=confusion_matrix(y_test,y_predict)
performance
array([[82, 0], [ 0, 12]])
Conclusion
Overall Accuracy = 100%
```

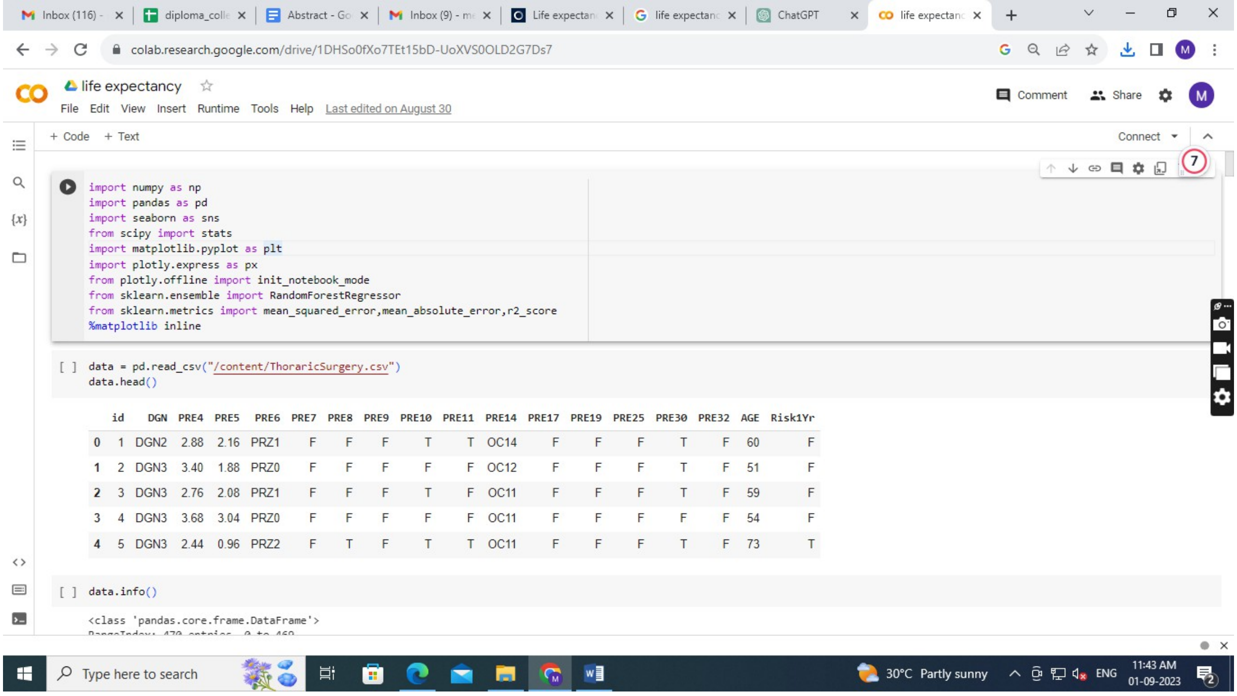
8.3 TESTING PHASE

1. **Pilot Testing:** Before proceeding with the main study or surgical procedures, consider conducting a pilot test. This small-scale trial allows you to identify and address any issues in your methodology, data collection instruments, or surgical techniques. It also helps in refining the procedures and ensuring they are feasible.
2. **Patient Selection and Preoperative Assessment:** Ensure that patients selected for the surgery are appropriate candidates based on your study criteria. Perform thorough preoperative assessments, including medical history, physical examinations, and any necessary diagnostic tests (e.g., imaging, laboratory tests).
3. **Surgical Simulation (If Applicable):** If your project involves the development or evaluation of new surgical techniques or devices, consider using surgical simulation models or cadaveric labs for training and initial testing. This allows surgeons to become familiar with the procedures before performing them on patients.
4. **Intraoperative Testing:** During the actual surgical procedures, various aspects can be tested or monitored, including:
   - **Surgical Techniques:** Ensure that the surgical techniques being tested are performed accurately and according to the established protocol.
   - **Device Performance:** If you're evaluating the effectiveness of a surgical device or technology, monitor its performance and safety during the surgery.
   - **Real-time Data Collection:** Collect relevant data during the surgery, such as surgical time, blood loss, complications, and any other parameters specified in your study.
5. **Immediate Postoperative Evaluation:** Assess the immediate outcomes of the surgery, including the patient's condition in the recovery room. Pay attention to any complications, adverse events, or unexpected outcomes.
6. **Follow-up and Long-term Monitoring:** For projects involving patient outcomes, establish a follow- up plan to monitor patients after surgery. Collect data at predetermined intervals to assess long-term outcomes, such as survival rates, quality of life, and any complications that may arise over time.
7. **Data Validation and Quality Control:** Continuously validate and ensure the quality of the data collected during the testing phase. Implement data validation checks and address any inconsistencies or missing information.
8. **Safety and Ethics Monitoring:** Continuously monitor patient safety throughout the testing phase. Ensure that all ethical and regulatory guidelines are strictly adhered to, and report any adverse events as required by institutional or regulatory protocols.

9. **Statistical Analysis:** If your project involves data analysis, perform the statistical analysis as planned during the design phase. Use appropriate statistical tests to analyze the data and interpret the results.
10. **Interim Review (if applicable):** Depending on the duration of your project, consider conducting interim reviews to assess progress, address any issues, and make necessary adjustments to the study or surgical procedures.
11. **Documentation and Reporting:** Maintain detailed records of all testing activities, surgical procedures, patient outcomes, and data collected. Prepare regular progress reports and document any deviations from the study protocol.
12. **External Evaluation (if applicable):** If your project involves collaboration with external experts, have them review and evaluate the testing phase to ensure objectivity and rigor.
13. **Quality Improvement:** Use the findings from the testing phase to identify areas for improvement in surgical techniques, patient care, or research methodology.
14. **Final Evaluation:** After completing the testing phase, perform a comprehensive final evaluation to determine whether your project's objectives were met. Summarize the results and draw conclusions based on the data collected.
15. **Dissemination:** Share the outcomes of your project through publications, presentations at conferences, and discussions within the medical community.

Remember that the testing phase should be conducted with the utmost care, adherence to ethical standards, and a focus on patient safety. It is essential to address any unexpected issues or complications promptly and to adjust the project plan as needed to ensure valid and reliable result.

8.4 OUTPUT SCREENSHOTS


colab.research.google.com/drive/1DHSo0fXo7TEt15bD-UoXVS0OLD2G7Ds7
life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
Comment Share
+ Code + Text
Connect
Type here to search
30°C Partly sunny
ENG
11:43 AM
01-09-2023

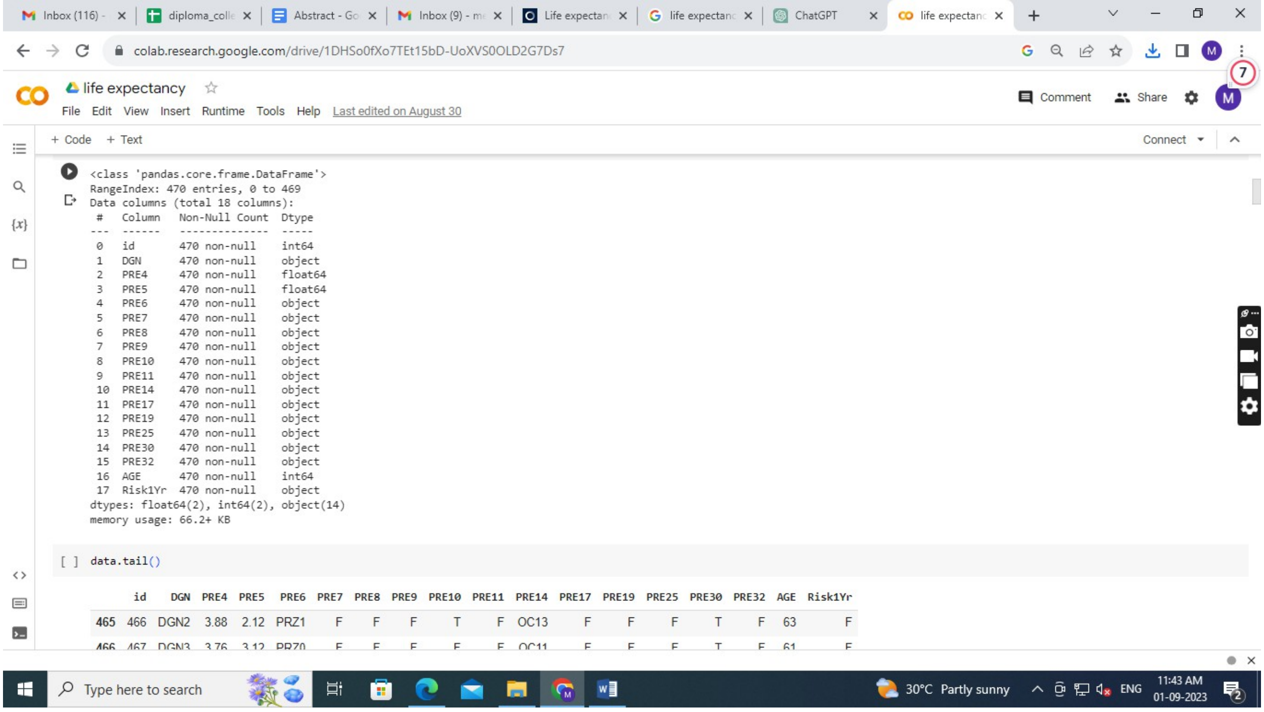
```
import numpy as np
import pandas as pd
import seaborn as sns
from scipy import stats
import matplotlib.pyplot as plt
import plotly.express as px
from plotly.offline import init_notebook_mode
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error,mean_absolute_error,r2_score
%matplotlib inline
```

```
data = pd.read_csv("/content/ThoracicSurgery.csv")
data.head()
```

	id	DGN	PRE4	PRE5	PRE6	PRE7	PRE8	PRE9	PRE10	PRE11	PRE14	PRE17	PRE19	PRE25	PRE30	PRE32	AGE	Risk1Yr
0	1	DGN2	2.88	2.16	PRZ1	F	F	F	T	T	OC14	F	F	F	T	F	60	F
1	2	DGN3	3.40	1.88	PRZ0	F	F	F	F	F	OC12	F	F	F	T	F	51	F
2	3	DGN3	2.76	2.08	PRZ1	F	F	F	T	F	OC11	F	F	F	T	F	59	F
3	4	DGN3	3.68	3.04	PRZ0	F	F	F	F	F	OC11	F	F	F	F	F	54	F
4	5	DGN3	2.44	0.96	PRZ2	F	T	F	T	T	OC11	F	F	F	T	F	73	T

```
data.info()
```

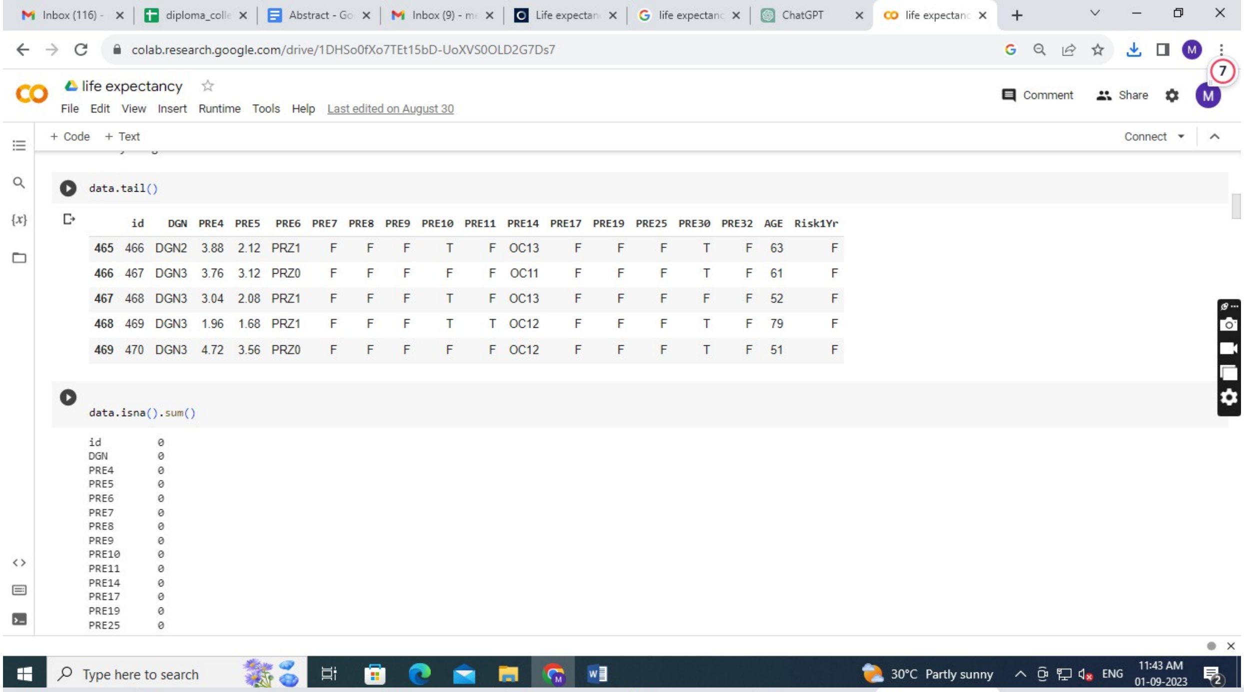
```
<class 'pandas.core.frame.DataFrame'>
```


colab.research.google.com/drive/1DHSo0fXo7TEt15bD-UoXVS0OLD2G7Ds7
life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
Comment Share
+ Code + Text
Connect
Type here to search
30°C Partly sunny
ENG
11:43 AM
01-09-2023

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 470 entries, 0 to 469
Data columns (total 18 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   id       470 non-null    int64
 1   DGN      470 non-null    object
 2   PRE4     470 non-null    float64
 3   PRE5     470 non-null    float64
 4   PRE6     470 non-null    object
 5   PRE7     470 non-null    object
 6   PRE8     470 non-null    object
 7   PRE9     470 non-null    object
 8   PRE10    470 non-null    object
 9   PRE11    470 non-null    object
 10  PRE14    470 non-null    object
 11  PRE17    470 non-null    object
 12  PRE19    470 non-null    object
 13  PRE25    470 non-null    object
 14  PRE30    470 non-null    object
 15  PRE32    470 non-null    object
 16  AGE      470 non-null    int64
 17  Risk1Yr  470 non-null    object
dtypes: float64(2), int64(2), object(14)
memory usage: 66.2+ KB
```

```
data.tail()
```

	id	DGN	PRE4	PRE5	PRE6	PRE7	PRE8	PRE9	PRE10	PRE11	PRE14	PRE17	PRE19	PRE25	PRE30	PRE32	AGE	Risk1Yr
465	466	DGN2	3.88	2.12	PRZ1	F	F	F	T	F	OC13	F	F	F	T	F	63	F
466	467	DGN3	3.76	3.12	PRZ0	F	F	F	F	F	OC11	F	F	F	T	F	61	F

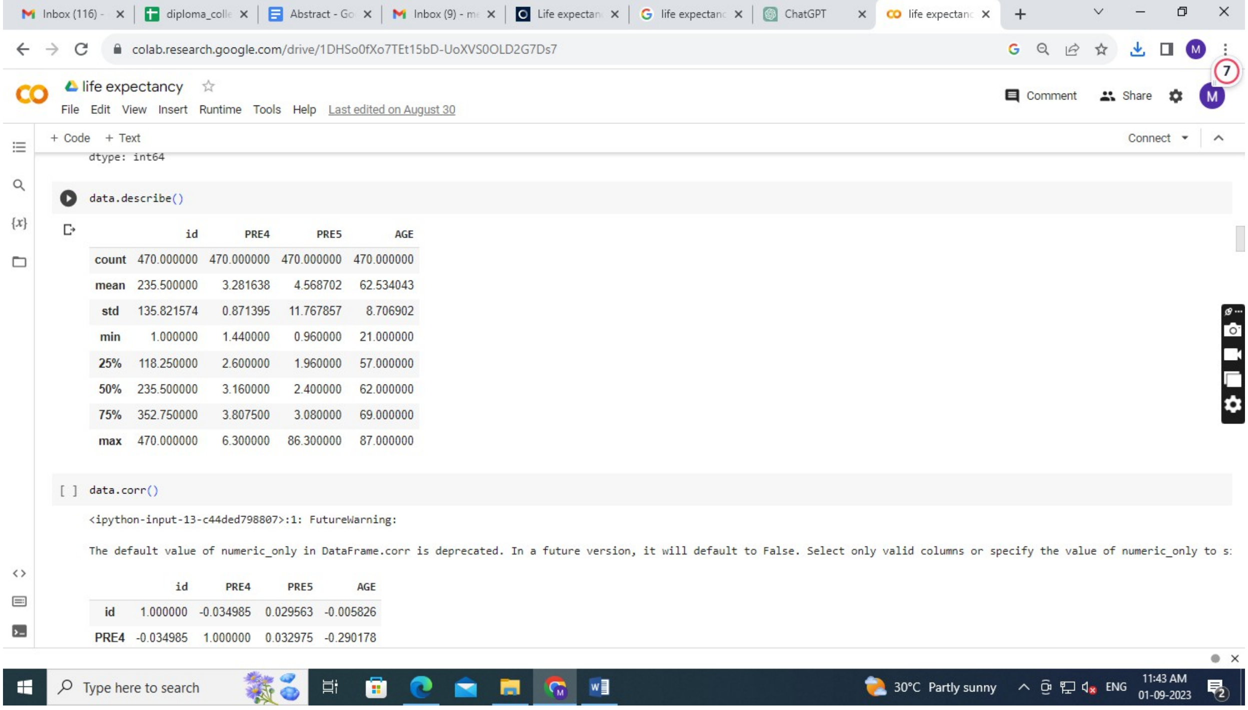


```
data.tail()
```

	id	DGN	PRE4	PRE5	PRE6	PRE7	PRE8	PRE9	PRE10	PRE11	PRE14	PRE17	PRE19	PRE25	PRE30	PRE32	AGE	Risk1Yr
465	466	DGN2	3.88	2.12	PRZ1	F	F	F	T	F	OC13	F	F	F	T	F	63	F
466	467	DGN3	3.76	3.12	PRZ0	F	F	F	F	F	OC11	F	F	F	T	F	61	F
467	468	DGN3	3.04	2.08	PRZ1	F	F	F	T	F	OC13	F	F	F	F	F	52	F
468	469	DGN3	1.96	1.68	PRZ1	F	F	F	T	T	OC12	F	F	F	T	F	79	F
469	470	DGN3	4.72	3.56	PRZ0	F	F	F	F	F	OC12	F	F	F	T	F	51	F

```
data.isna().sum()
```

```
id       0
DGN      0
PRE4     0
PRE5     0
PRE6     0
PRE7     0
PRE8     0
PRE9     0
PRE10    0
PRE11    0
PRE14    0
PRE17    0
PRE19    0
PRE25    0
```



```
dtype: int64
```

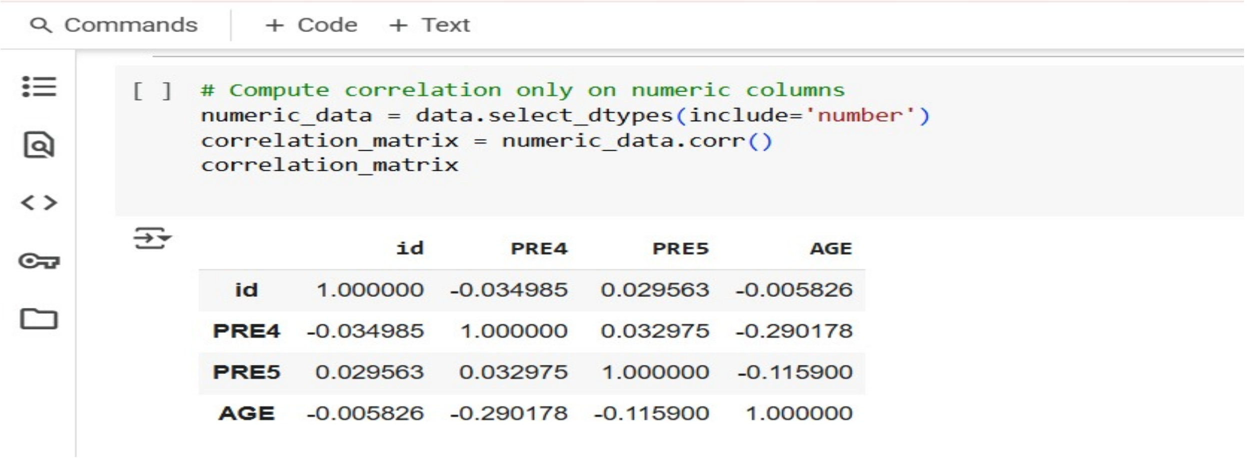
```
data.describe()
```

	id	PRE4	PRE5	AGE
count	470.000000	470.000000	470.000000	470.000000
mean	235.500000	3.281638	4.568702	62.534043
std	135.821574	0.871395	11.767857	8.706902
min	1.000000	1.440000	0.960000	21.000000
25%	118.250000	2.600000	1.960000	57.000000
50%	235.500000	3.160000	2.400000	62.000000
75%	352.750000	3.807500	3.080000	69.000000
max	470.000000	6.300000	86.300000	87.000000

```
[ ] data.corr()
```

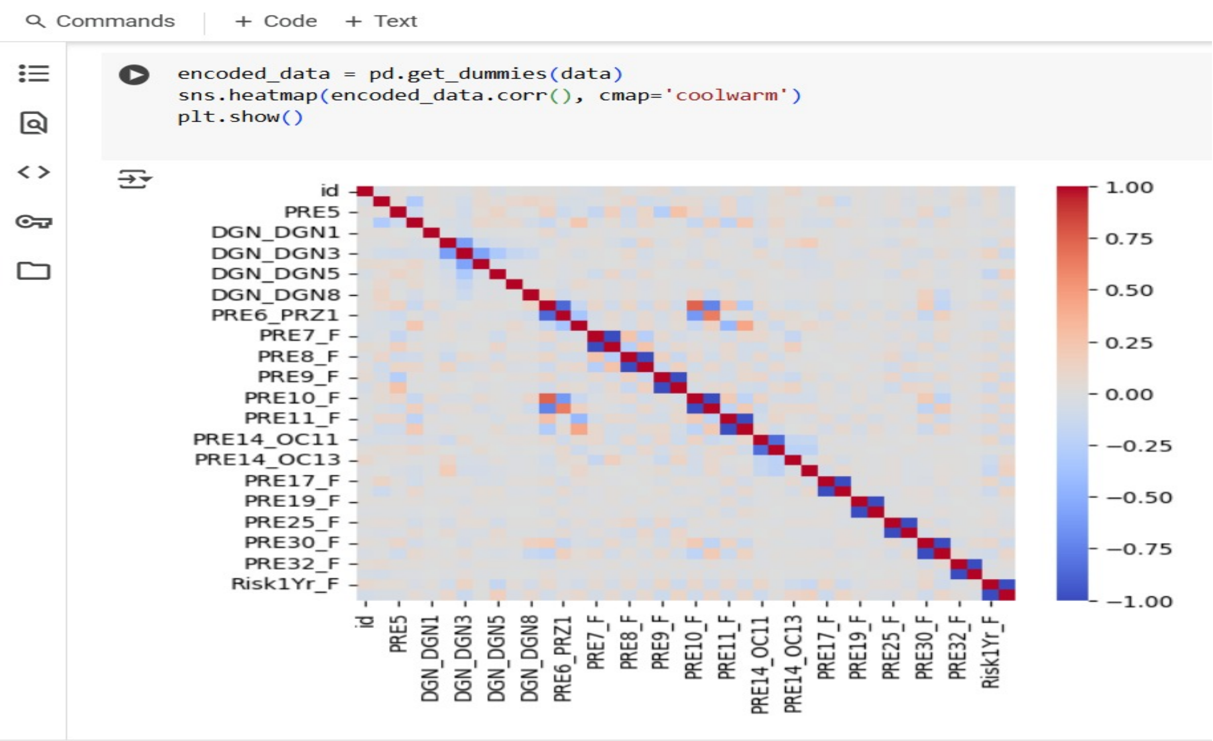
```
<ipython-input-13-c44ded798807>:1: FutureWarning:

The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to s:
```

	id	PRE4	PRE5	AGE
id	1.000000	-0.034985	0.029563	-0.005826
PRE4	-0.034985	1.000000	0.032975	-0.290178

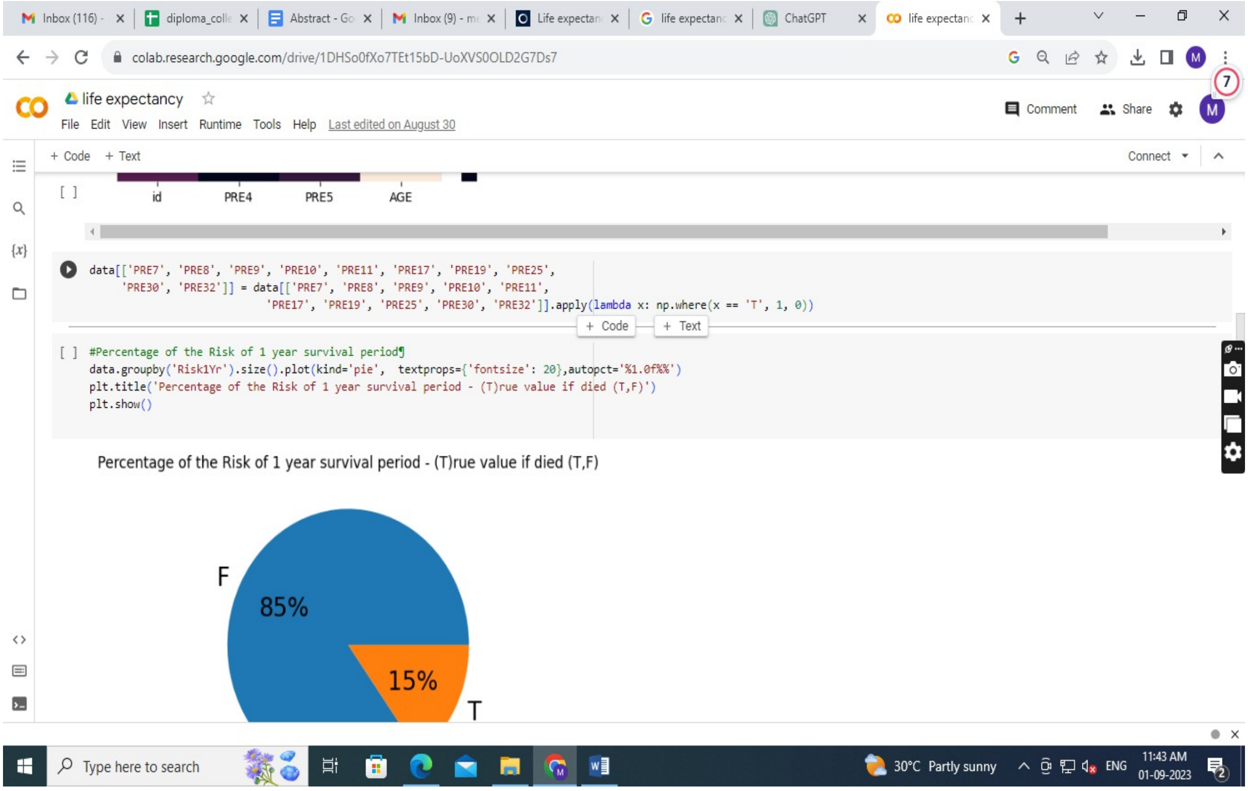


```
[ ] # Compute correlation only on numeric columns
numeric_data = data.select_dtypes(include='number')
correlation_matrix = numeric_data.corr()
correlation_matrix
```

	id	PRE4	PRE5	AGE
id	1.000000	-0.034985	0.029563	-0.005826
PRE4	-0.034985	1.000000	0.032975	-0.290178
PRE5	0.029563	0.032975	1.000000	-0.115900
AGE	-0.005826	-0.290178	-0.115900	1.000000



```
encoded_data = pd.get_dummies(data)
sns.heatmap(encoded_data.corr(), cmap='coolwarm')
plt.show()
```

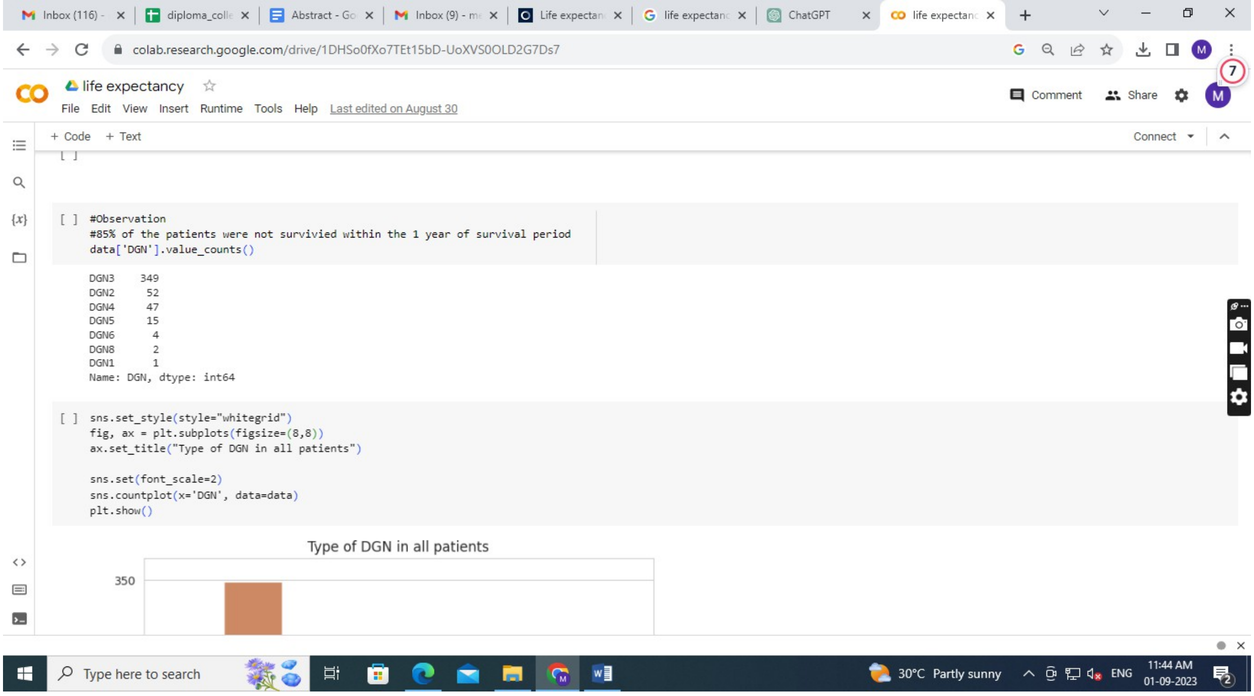


```
data[['PRE7', 'PRE8', 'PRE9', 'PRE10', 'PRE11', 'PRE17', 'PRE19', 'PRE25',
     'PRE30', 'PRE32']] = data[['PRE7', 'PRE8', 'PRE9', 'PRE10', 'PRE11',
                        'PRE17', 'PRE19', 'PRE25', 'PRE30', 'PRE32']].apply(lambda x: np.where(x == 'T', 1, 0))
```

```
#Percentage of the Risk of 1 year survival period¶
data.groupby('Risk1Yr').size().plot(kind='pie',  textprops={'fontsize': 20},autopct='%1.0f%%')
plt.title('Percentage of the Risk of 1 year survival period - (T)rue value if died (T,F)')
plt.show()
```

Percentage of the Risk of 1 year survival period - (T)rue value if died (T,F)

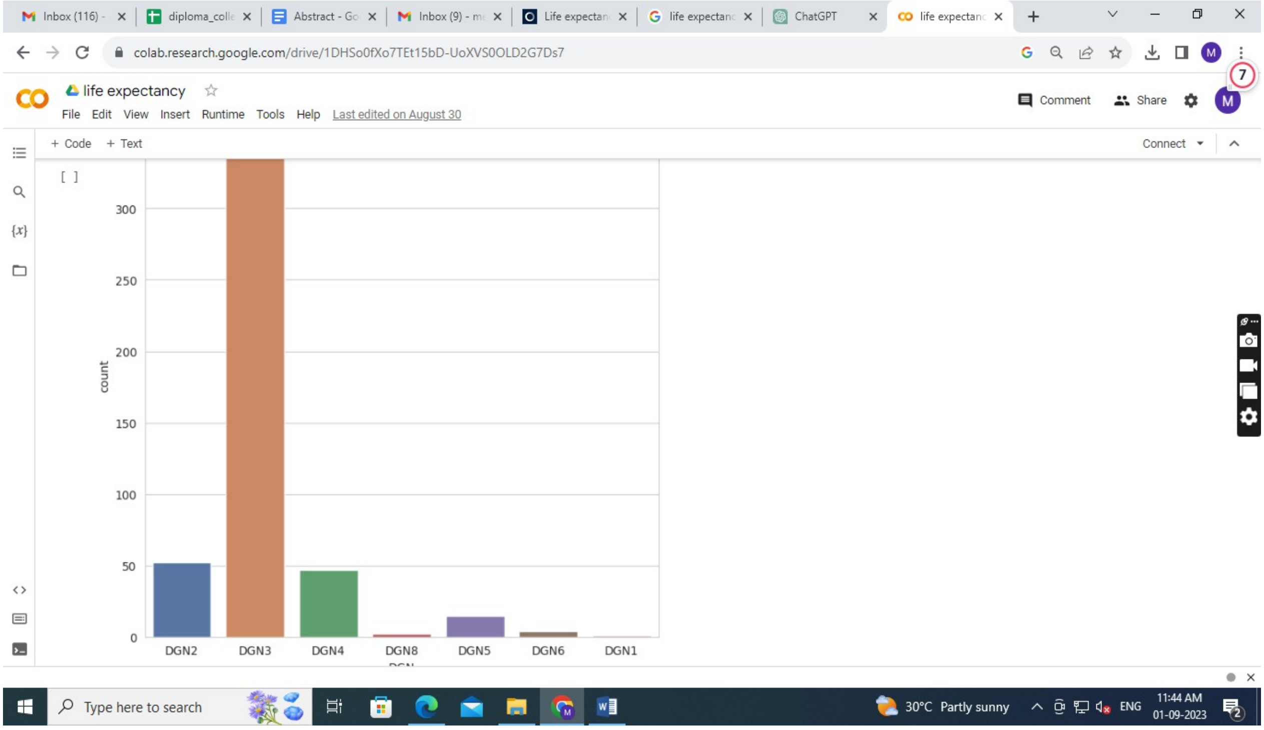
colab.research.google.com/drive/1DHSo0fXo7TEt15bD-UoXVS0OLD2G7Ds7
life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
Comment
Share
Connect
+ Code
+ Text
id
PRE4
PRE5
AGE
F
85%
15%
T
Type here to search
30°C Partly sunny
ENG
11:43 AM
01-09-2023

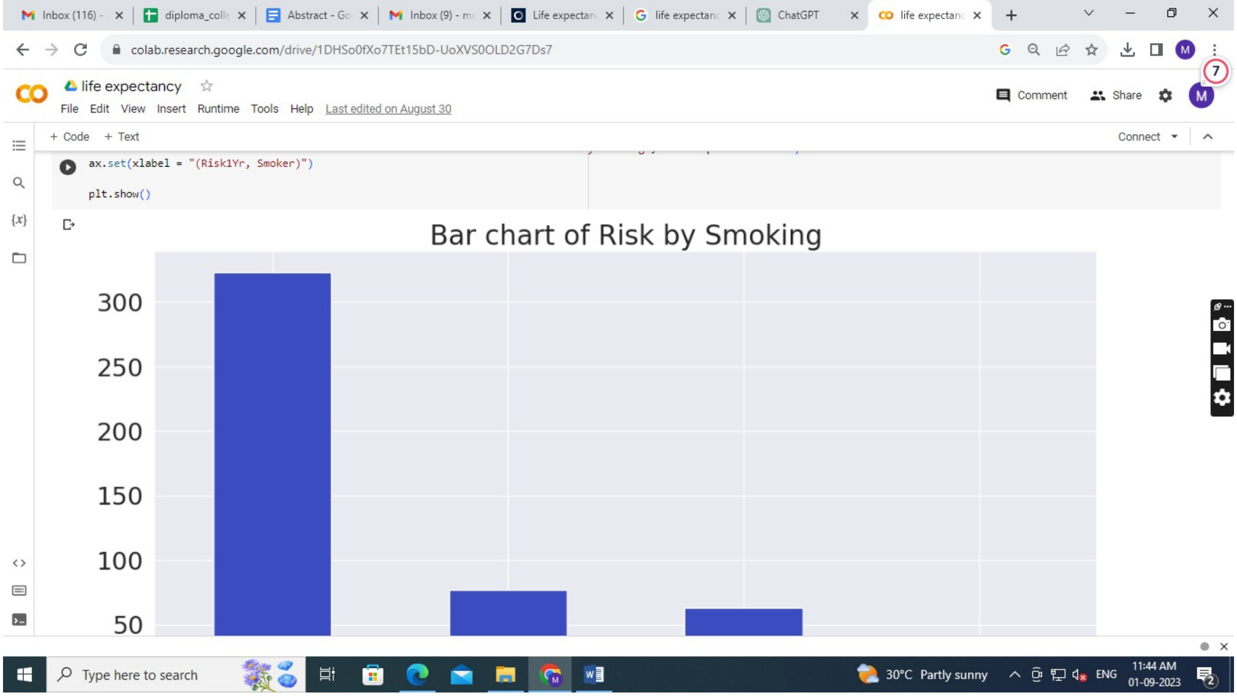


```
#Observation
#85% of the patients were not survivied within the 1 year of survival period
data['DGN'].value_counts()
```

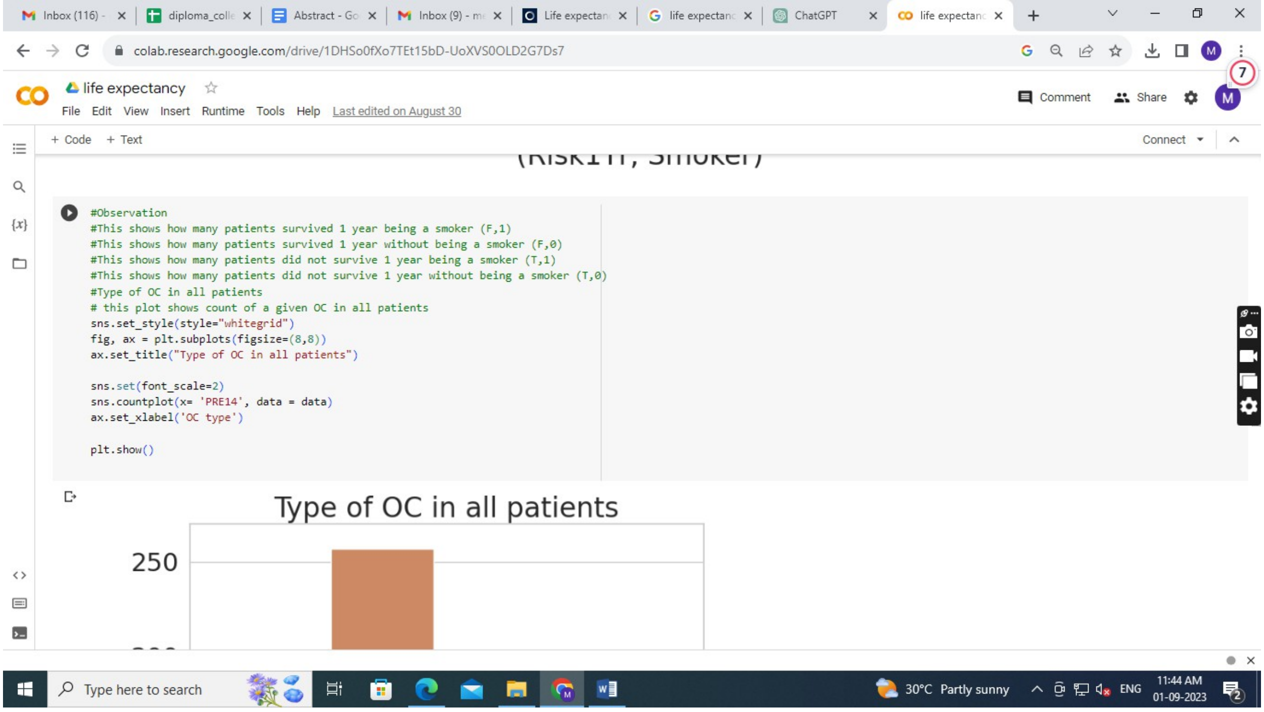
```
DGN3    349
DGN2     52
DGN4     47
DGN5     15
DGN6      4
DGN8      2
DGN1      1
Name: DGN, dtype: int64
```

```
sns.set_style(style="whitegrid")
fig, ax = plt.subplots(figsize=(8,8))
ax.set_title("Type of DGN in all patients")

sns.set(font_scale=2)
sns.countplot(x='DGN', data=data)
plt.show()
```



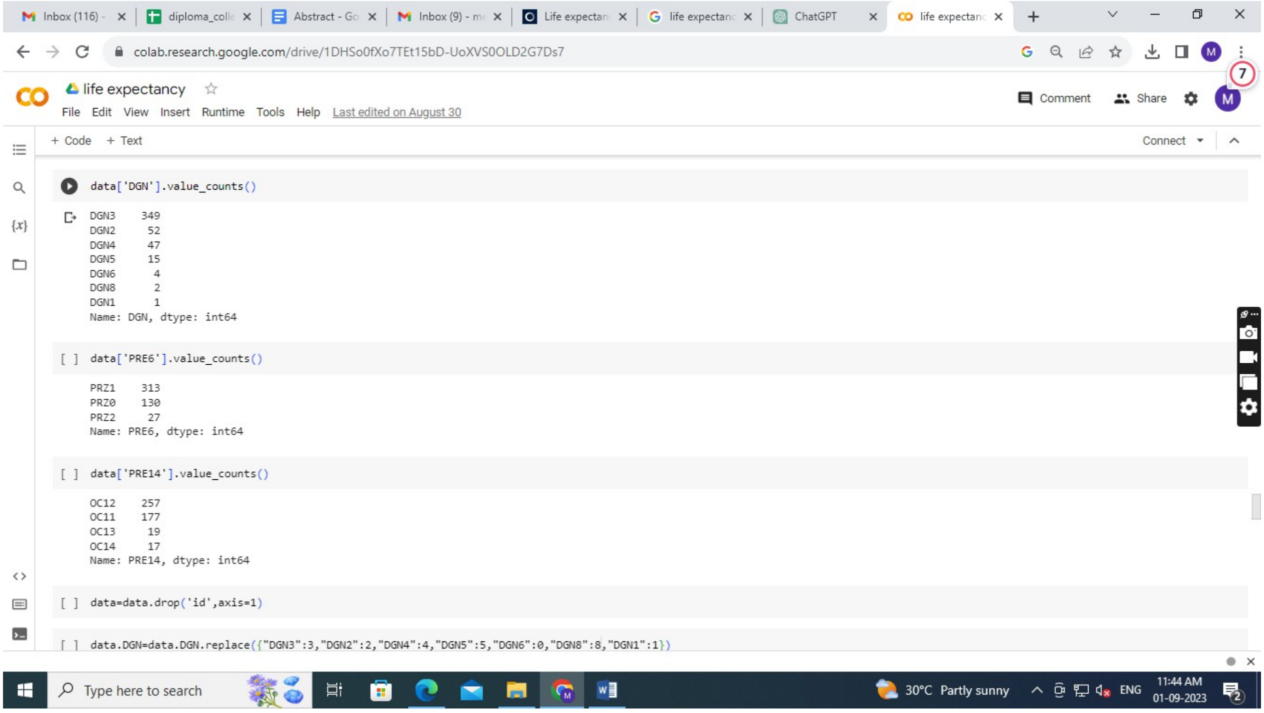


```
ax.set(xlabel = "(Risk1Yr, Smoker)")

plt.show()
```



```
#Observation
#This shows how many patients survived 1 year being a smoker (F,1)
#This shows how many patients survived 1 year without being a smoker (F,0)
#This shows how many patients did not survive 1 year being a smoker (T,1)
#This shows how many patients did not survive 1 year without being a smoker (T,0)
#Type of OC in all patients
# this plot shows count of a given OC in all patients
sns.set_style(style="whitegrid")
fig, ax = plt.subplots(figsize=(8,8))
ax.set_title("Type of OC in all patients")

sns.set(font_scale=2)
sns.countplot(x= 'PRE14', data = data)
ax.set_xlabel('OC type')

plt.show()
```

life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
Comment
Share
Connect
Type of OC in all patients
250
Type here to search
30°C Partly sunny
11:44 AM
01-09-2023



life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
+ Code + Text
Comment Share Connect

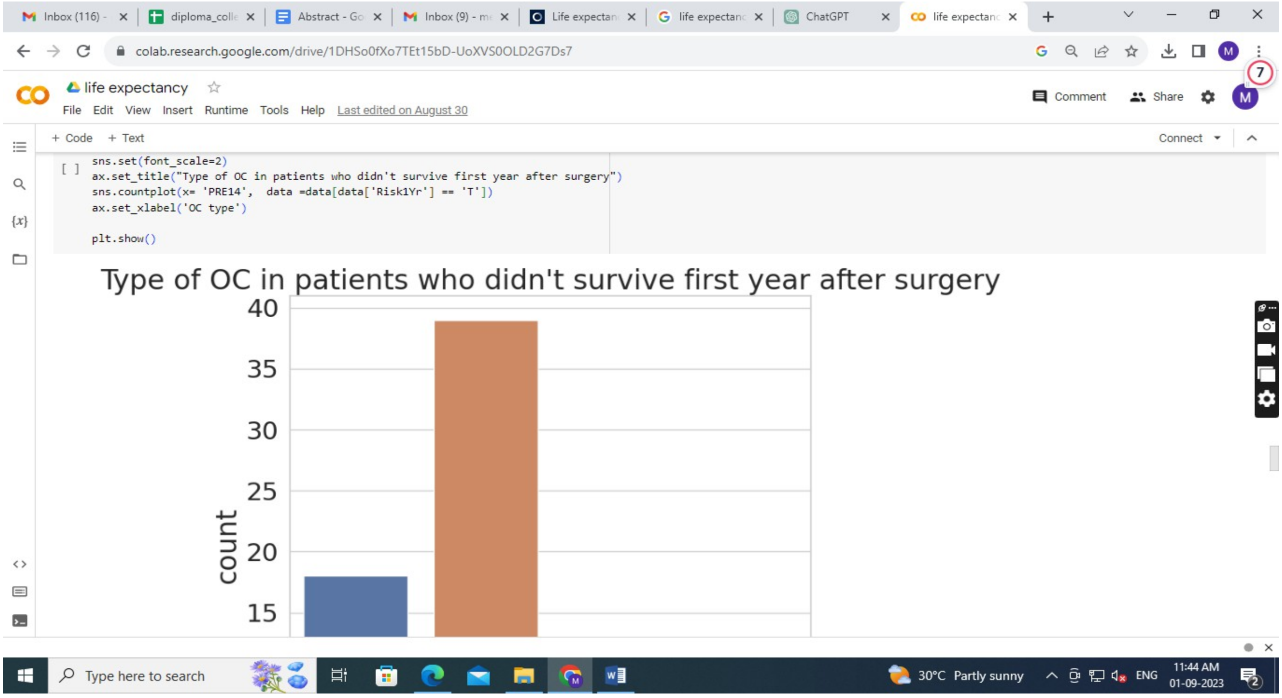
```
data['DGN'].value_counts()

DGN3    349
DGN2     52
DGN4     47
DGN5     15
DGN6      4
DGN8      2
DGN1      1
Name: DGN, dtype: int64
```

```
data['PRE6'].value_counts()

PRZ1    313
PRZ0    130
PRZ2     27
Name: PRE6, dtype: int64
```

```
data['PRE14'].value_counts()

OC12    257
OC11    177
OC13     19
OC14     17
Name: PRE14, dtype: int64
```

```
data=data.drop('id',axis=1)
```

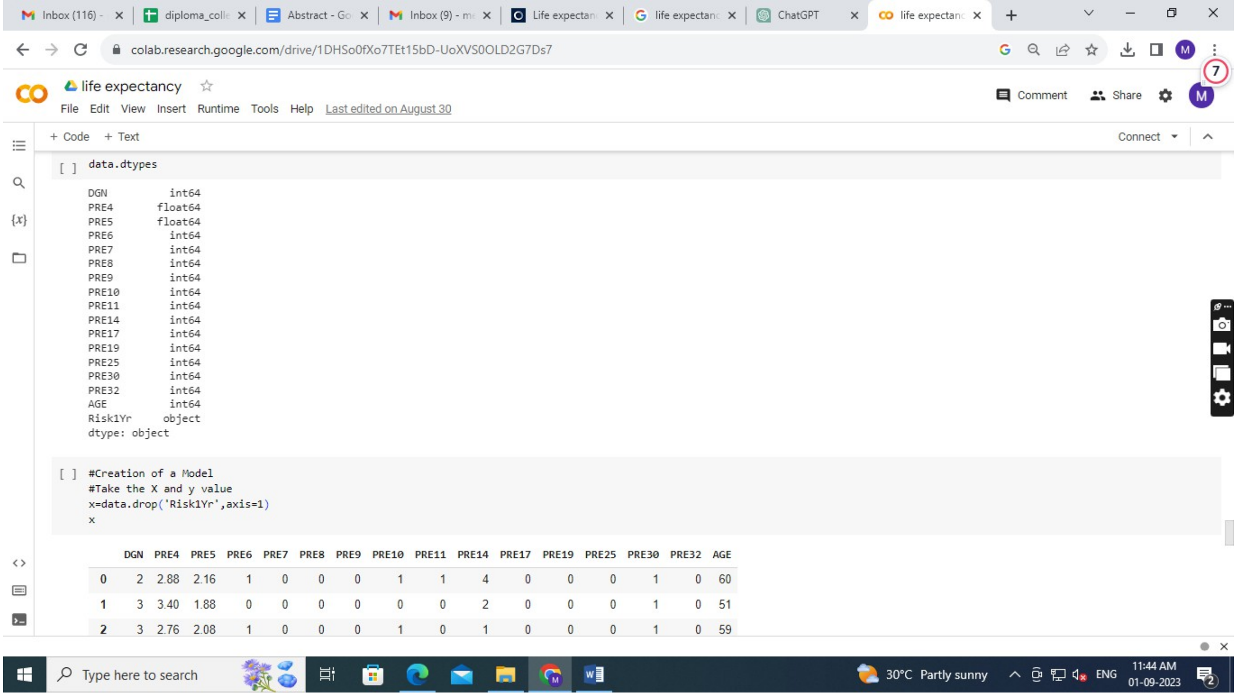
```
data.DGN=data.DGN.replace({"DGN3":3,"DGN2":2,"DGN4":4,"DGN5":5,"DGN6":0,"DGN8":8,"DGN1":1})
```



life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
+ Code + Text
Comment Share Connect

```
sns.set(font_scale=2)
ax.set_title("Type of OC in patients who didn't survive first year after surgery")
sns.countplot(x= 'PRE14',  data =data[data['Risk1Yr'] == 'T'])
ax.set_xlabel('OC type')

plt.show()
```

Type of OC in patients who didn't survive first year after surgery
40
35
30
25
20
15
count

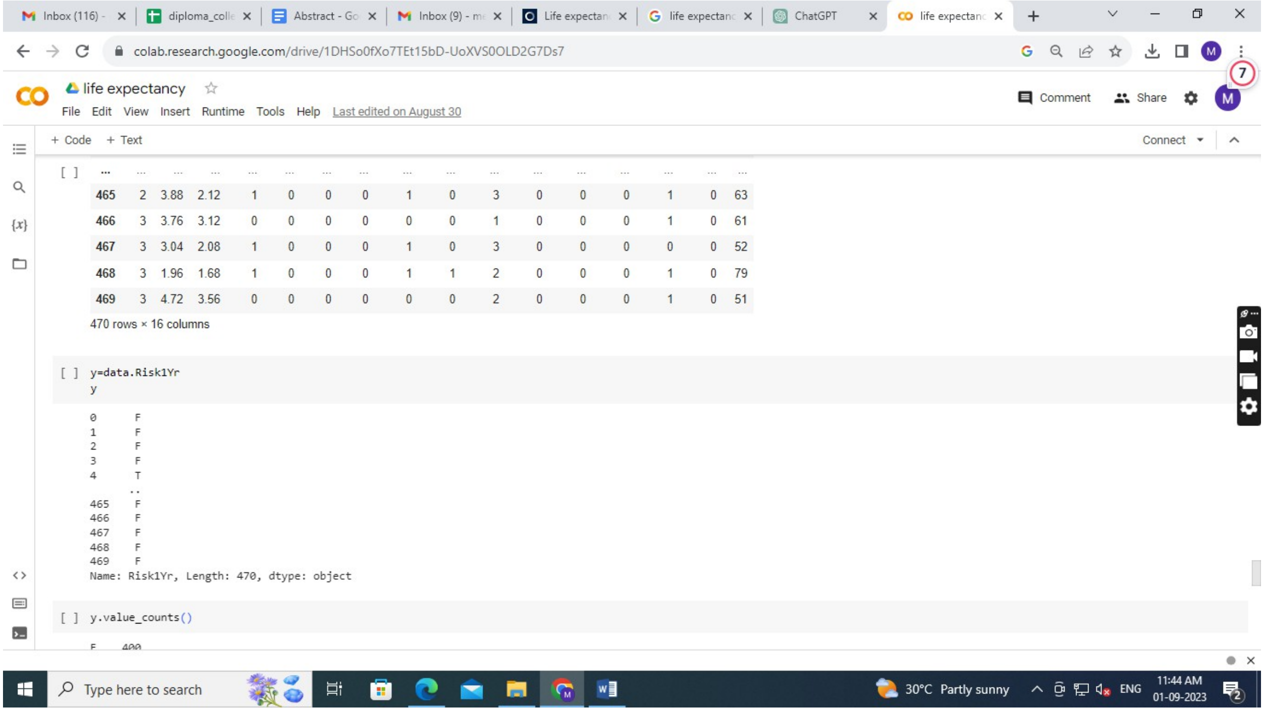

life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
Comment Share
+ Code + Text
Connect

```
data.dtypes
```

```
DGN          int64
PRE4       float64
PRE5       float64
PRE6         int64
PRE7         int64
PRE8         int64
PRE9         int64
PRE10        int64
PRE11        int64
PRE14        int64
PRE17        int64
PRE19        int64
PRE25        int64
PRE30        int64
PRE32        int64
AGE          int64
Risk1Yr     object
dtype: object
```

```
#Creation of a Model
#Take the X and y value
x=data.drop('Risk1Yr',axis=1)
x
```

	DGN	PRE4	PRE5	PRE6	PRE7	PRE8	PRE9	PRE10	PRE11	PRE14	PRE17	PRE19	PRE25	PRE30	PRE32	AGE
0	2	2.88	2.16	1	0	0	0	1	1	4	0	0	0	1	0	60
1	3	3.40	1.88	0	0	0	0	0	0	2	0	0	0	1	0	51
2	3	2.76	2.08	1	0	0	0	1	0	1	0	0	0	1	0	59


life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
Comment Share
+ Code + Text
Connect

...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
465	2	3.88	2.12	1	0	0	0	1	0	3	0	0	0	1	0	63
466	3	3.76	3.12	0	0	0	0	0	0	1	0	0	0	1	0	61
467	3	3.04	2.08	1	0	0	0	1	0	3	0	0	0	0	0	52
468	3	1.96	1.68	1	0	0	0	1	1	2	0	0	0	1	0	79
469	3	4.72	3.56	0	0	0	0	0	0	2	0	0	0	1	0	51

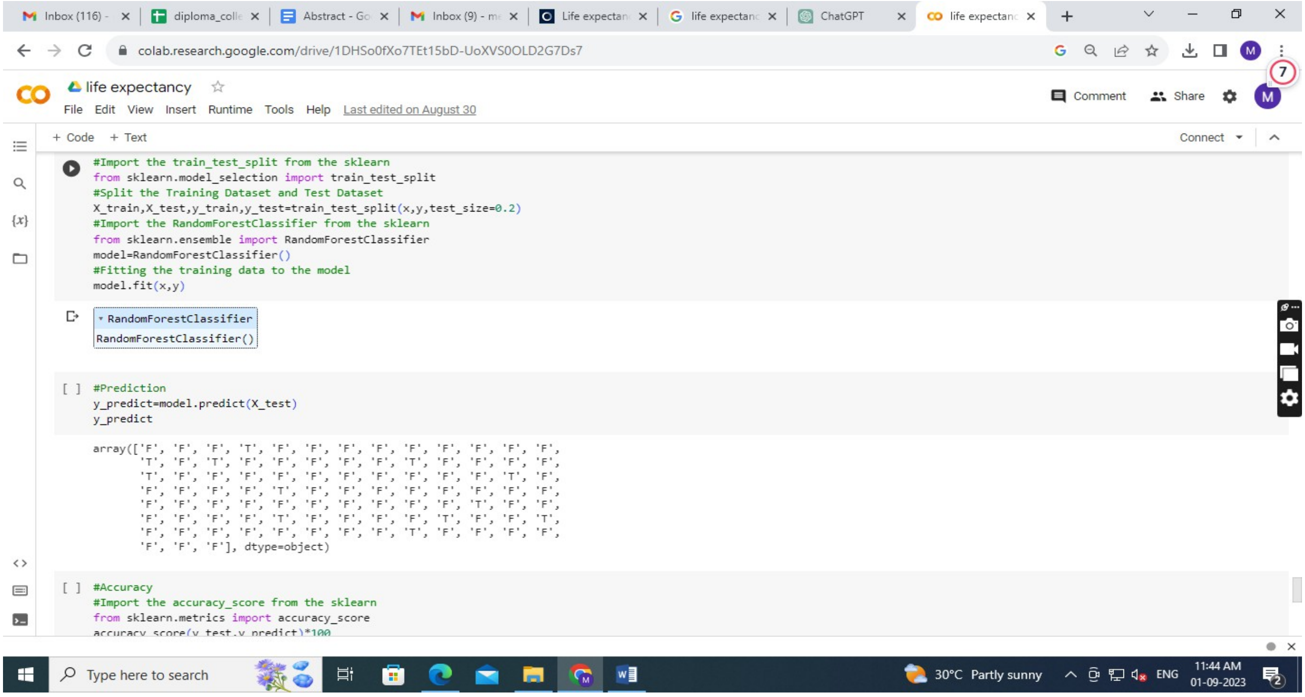
470 rows × 16 columns

```
y=data.Risk1Yr
y
```

```
0      F
1      F
2      F
3      F
4      T
      ..
465    F
466    F
467    F
468    F
469    F
Name: Risk1Yr, Length: 470, dtype: object
```

```
y.value_counts()
```

```
F    400
```



colab.research.google.com/drive/1DHSo0fXo7TEt15bD-UoXVS0OLD2G7Ds7
life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
Comment Share
+ Code + Text
Connect

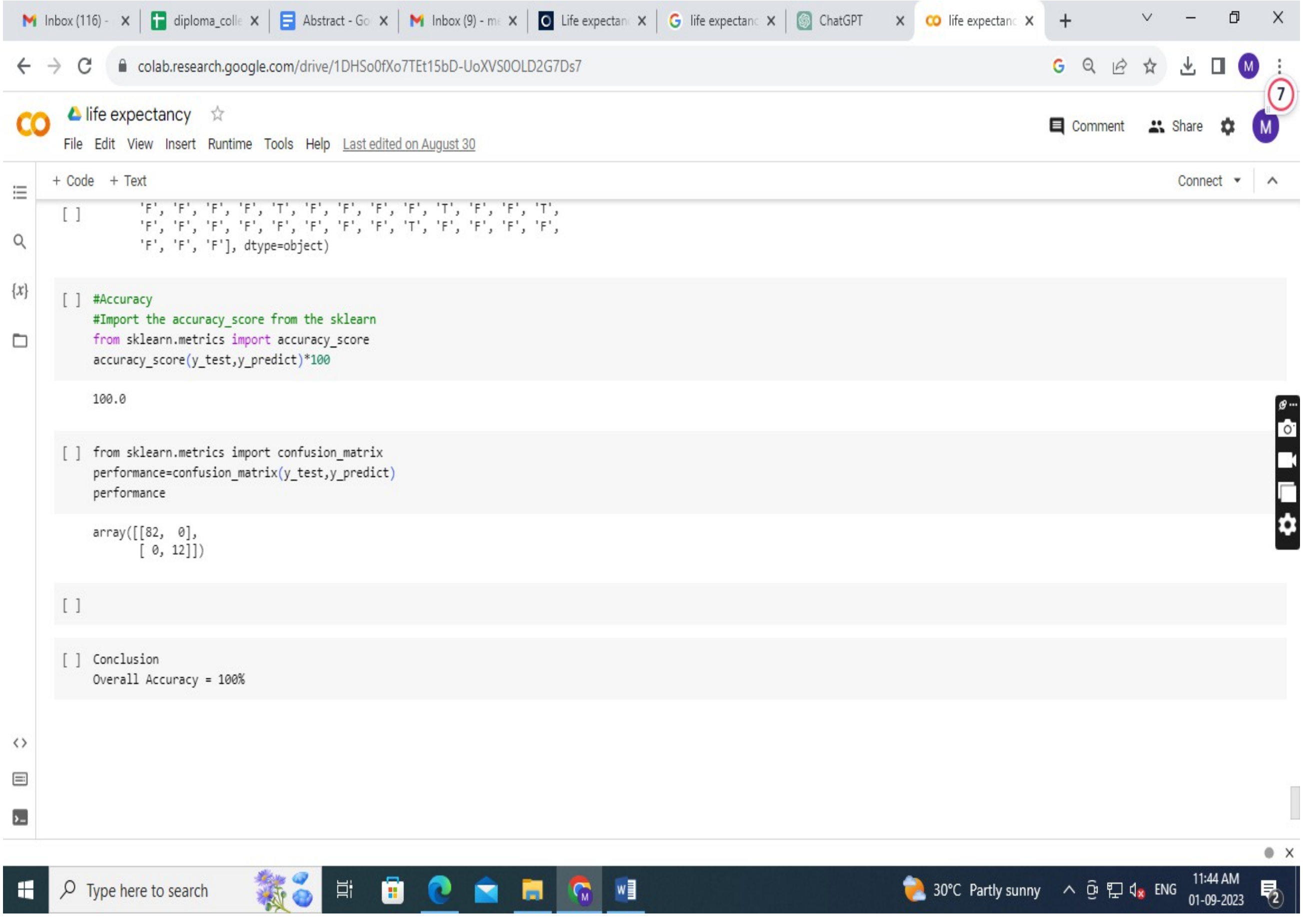
```
#Import the train_test_split from the sklearn
from sklearn.model_selection import train_test_split
#Split the Training Dataset and Test Dataset
X_train,X_test,y_train,y_test=train_test_split(x,y,test_size=0.2)
#Import the RandomForestClassifier from the sklearn
from sklearn.ensemble import RandomForestClassifier
model=RandomForestClassifier()
#Fitting the training data to the model
model.fit(x,y)
```

```
▾ RandomForestClassifier
RandomForestClassifier()
```

```
[ ] #Prediction
    y_predict=model.predict(X_test)
    y_predict
```

```
array(['F', 'F', 'F', 'T', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F',
       'T', 'F', 'T', 'F', 'F', 'F', 'F', 'F', 'T', 'F', 'F', 'F', 'F',
       'T', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'T', 'F',
       'F', 'F', 'F', 'F', 'T', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F',
       'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'T', 'F', 'F',
       'F', 'F', 'F', 'F', 'T', 'F', 'F', 'F', 'F', 'T', 'F', 'F', 'T',
       'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'T', 'F', 'F', 'F', 'F',
       'F', 'F', 'F'], dtype=object)
```

```
[ ] #Accuracy
    #Import the accuracy_score from the sklearn
    from sklearn.metrics import accuracy_score
    accuracy_score(y_test,y_predict)*100
```



colab.research.google.com/drive/1DHSo0fXo7TEt15bD-UoXVS0OLD2G7Ds7
life expectancy
File Edit View Insert Runtime Tools Help Last edited on August 30
Comment Share
+ Code + Text
Connect

```
[ ]    'F', 'F', 'F', 'F', 'T', 'F', 'F', 'F', 'F', 'T', 'F', 'F', 'T',
       'F', 'F', 'F', 'F', 'F', 'F', 'F', 'F', 'T', 'F', 'F', 'F', 'F',
       'F', 'F', 'F'], dtype=object)
```

```
[ ] #Accuracy
    #Import the accuracy_score from the sklearn
    from sklearn.metrics import accuracy_score
    accuracy_score(y_test,y_predict)*100
```

```
100.0
```

```
[ ] from sklearn.metrics import confusion_matrix
    performance=confusion_matrix(y_test,y_predict)
    performance
```

```
array([[82,  0],
       [ 0, 12]])
```

```
[ ]
```

```
[ ] Conclusion
    Overall Accuracy = 100%
```

TEST CASES

Test Case	Patient ID	Diagnosis	PRE4 (Lung Func)	PRE5 (FEV1)	PRE6 (Performance)	Pain (PRE10)	Cough (PRE11)	Age	Outcome (Risk1Yr)	Interpretation
1	1	DGN2	2.88	2.16	PRZ1	Yes	No	60	F (Survived)	Good lung function & stable condition
2	5	DGN3	2.44	0.96	PRZ2	Yes	Yes	73	T (Did Not Survive)	Poor lung capacity, old age
3	2	DGN3	3.40	1.88	PRZ0	No	No	51	F (Survived)	Excellent health profile
4	7	DGN3	4.36	3.28	PRZ1	Yes	No	59	T (Did Not Survive)	Possibly risky diagnosis
5	8	DGN2	3.19	2.50	PRZ1	Yes	Yes	66	T (Did Not Survive)	Symptoms + moderate risk factors

S

CHAPTER 9

LIMITATIONS

- This work has several limitations. The National office of statistics provided detailed information on late survival of people, stratified by age and sex. This allowed us to perform matching using these variables.

- However, some important comorbidities could not be analyzed. Moreover, terminal or high-risk surgical patients may not have been candidates for surgery. This is a retrospective study based on patients who underwent surgery at a single institution.

- All patients have been treated and followed up in a certain way during follow-up making it difficult to extrapolate the results. Nevertheless, it is shown that, at least under certain circumstances, patients undergoing SAVR fully recover their life expectancy.

CHAPTER 10

RESULT

Average life expectancy of surgical patients who survived the postoperative period was 90.91 months (95% CI 82.99–97.22), compared to 92.94 months (95% CI 92.39–93.55) in the control group. One-, 5- and 8-year survival rates for SAVR patients who were discharged from the hospital were 94.9% (95% CI 92.74–96.43%), 71.66% (95% CI 67.37–75.5%) and 44.48% (95% CI 38.14–50.61%), respectively, compared to that of the general population: 95.8% (95% CI 95.64–95.95%), 70.64% (95% CI 70.28%–71%) and 47.91% (95% CI 47.52–48.31%), respectively (HR 1.07, 95% CI 0.94–1.22).

A total of 614 patients met the inclusion criteria and 61 400 observations were created by matching age and sex.

The mean age of the SAVR group and reference population were 79.53 ± 2.87 and 79.58 ± 2.52 years, respectively. A total of 325 (52.93%) in the SAVR group, and 32 500 (52.93%) in the reference population were women. Logistic EuroSCORE and EuroSCORE II were 9.23 ± 5.16 and 3.95 ± 2.93, respectively.

Other basal characteristics of the surgical group are described in Table 1. Thirty-six (5.86%) patients died during the perioperative period. During the first 30 days of follow-up, 198 (0.32%) simulated people died.In the surgical group, no patients were lost to follow-up. The mean follow-up duration of the censored patients was 60.65 ± 26.9 months.

Life expectancy (median of survival) was 85.67 months (95% CI 78.45–93.17) for the surgical group and 92.72 months (95% CI 92.18–93.33) for the control group.

The differences in the results of their study and ours may be due to the existing differences in life expectancy between regions of the same country.

Patients from the SWEDEHEART registry were from different regions of the country but their life expectancy was compared with the mean life expectancy of the general population of the whole country.

However, patients from our study all came from the same region of the same country and their life expectancy was compared with the general population of the same geographical region.

In addition, more than 90% of our patients with coronary disease were treated with concomitant coronary surgery, which may have improved their long-term survival. The study by Glaser did not provide this information.

This study is limited to using only ELM as a classification algorithm and SMOTE to overcome the challenge of unbalanced data. In addition, the dataset used focuses only on a single source retrospectively collected at the Wroclaw Center for Thoracic Surgery: postoperative survival data for lung cancer patients. There are many factors associated with the survival of postoperative lung cancer patients.

There are many factors associated with the survival of post-surgical lung cancer patients. There are many factors associated with the survival of post-surgical lung cancer patients.
In the data used, many other indicators were not included, and prognostic variables consisting of neoadjuvant therapy, biomarkers, anatomopathological findings, and tumor genome analysis are limitations of this study .

Several tests were conducted to determine the proportion of training and testing data that can provide good results based on accuracy, F-Measure, and ROC. The composition of training data and test data used in this work is 60:40, 70:30, 80:20, and 90:10.

Additionally, neuron testing was performed to ascertain the number of neurons required in the ELM process to get the best results. This test uses 5 to 50 neurons in multiples of 5. Each neuron is tested ten times, and the resulting accuracy, F-measure, and ROC results are averaged. The first test results for the proportion of training data and test data 60:40

CHAPTER 11

CONCLUSION

To conclude that we got highest accuracy for simple logistic regression gave highest accuracy when compared with other two algorithms that are random forest and KNN.

We got an accuracy of 85%. Even if I have made all these predictions it always narrows down to one main criteria that is care taken by an individual. We can only give hope of showing the numbers that they will live for .

This invention opens up a lot of scope for more developments. We took algorithms of Logistic regression and Random forest and KNN and all these algorithms are calculated again with weights and without weights. With weights also we got more accuracy through simple logistic regression.

The purpose of taking Random forest is to compare the precision of simple logistic regression and Random forest. Precision made by Random forest is nearly 2% . The precision of logistic regression is constant throughout the experiment but Random forest is not constant.

Gathering comprehensive and accurate data is essential. This data should include patient demographics, medical history, surgical details (e.g., procedure type, surgeon's experience), and post-operative outcomes (e.g., complications, length of hospital stay).

Feature Selection: Selecting the most relevant features is crucial for building accurate predictive models. Feature selection methods, including statistical tests, machine learning algorithms, and expert consultation, can help identify the key factors that influence life expectancy.

Model Building: Utilize appropriate machine learning or statistical models to predict life expectancy after thoracic surgery. Common models include regression models (linear regression, logistic regression), decision trees, random forests, and neural networks.

Evaluation Metrics: Assess the performance of your predictive model using relevant evaluation metrics. In the case of life expectancy prediction, metrics like mean absolute error (MAE), root mean square error (RMSE), and concordance index (C-index) can be used to measure predictive accuracy.

Clinical Interpretability: Ensure that the predictive model is clinically interpretable. It's essential for healthcare professionals to understand how the model arrives at its predictions so they can make informed decisions.

Validation: Use cross-validation techniques to validate the model's performance on unseen data.

This helps assess its generalizability and robustness.

Clinical Relevance: Consider the clinical relevance of the features and model predictions. Discuss the results with medical experts to validate the model's findings and assess its potential utility in clinical practice.

Patient Counseling: If the model provides meaningful predictions, it can be a valuable tool for patient counseling. Surgeons and healthcare providers can use the predicted life expectancy to inform patients about potential outcomes and treatment options.

Limitations: Acknowledge the limitations of the predictive model. These may include data quality issues, model assumptions, and the complexity of human health, which cannot always be captured by available data.Ethical Considerations: Ensure that patient data is handled with the utmost care and in compliance with ethical and privacy regulations, such as HIPAA in the United States or GDPR in Europe.

Continual Improvement: Medical knowledge and technology are continually evolving. Regularly update and refine the predictive model as new data becomes available and medical practices change.In conclusion, predicting life expectancy after thoracic surgery is a valuable endeavor that can assist healthcare professionals in making informed decisions and providing better patient care.

However, it's essential to approach this task with careful consideration of data quality, model interpretability, and ethical considerations while involving medical experts throughout the process to ensure the model's clinical relevance and validity.Machine learning techniques have the potential to revolutionize the way we approach and understand life expectancy after thoracic surgery.

By harnessing the power of data and advanced algorithms, these techniques empower healthcare providers to make more informed decisions, improve patient outcomes, and enhance the overall quality of care in thoracic surgery. However, it is essential to approach their implementation with careful consideration of ethical, privacy, and clinical factors to ensure their successful integration into clinical practice.

REFERENCES

[1] V. Sindhu, S. A. S. Prabha, S. Veni and M. Hemalatha. (2014), “Thoracic surgery analysis using datamining techniques”, International Journal of Computer Technology & Applications, Vol. 5 pp.578-586.

[2] KonstantinaKourou , Themis P. Exarchos, Konstantinos P. Exarchos, Michalis V. Karamouzis and Dimitrios I. Fotiadisa. (2015), “Machine learning applications in cancer prognosis and prediction”, Computational and Structural Biotechnology Journal, Vol. 13,pp.8-17.

[3] KwetisheJoro Danjuma. (2015), “Performance evaluation of machine learning algorithms in post-operative life expectancy in the lung cancer patients”, IJCSI International Journal of Computer Science Issues, Vol. 12, No. 2, pp.189-199 .

[4] Joseph A. Cruz, David S. Wishart. (2006), “Applications of machine learning in cancer prediction and prognosis”, Cancer Informatics, Vol. 2, pp.59-77 2006.

[5] Mehdi Naseriparsa, Amir-Masoud Bidgoli and Touraj Varaee. (2013), “A hybrid feature selection method to improve performance of a group of classification algorithms", International Journal of Computer Applications,Vol. 69, No. 17,pp.28-35.

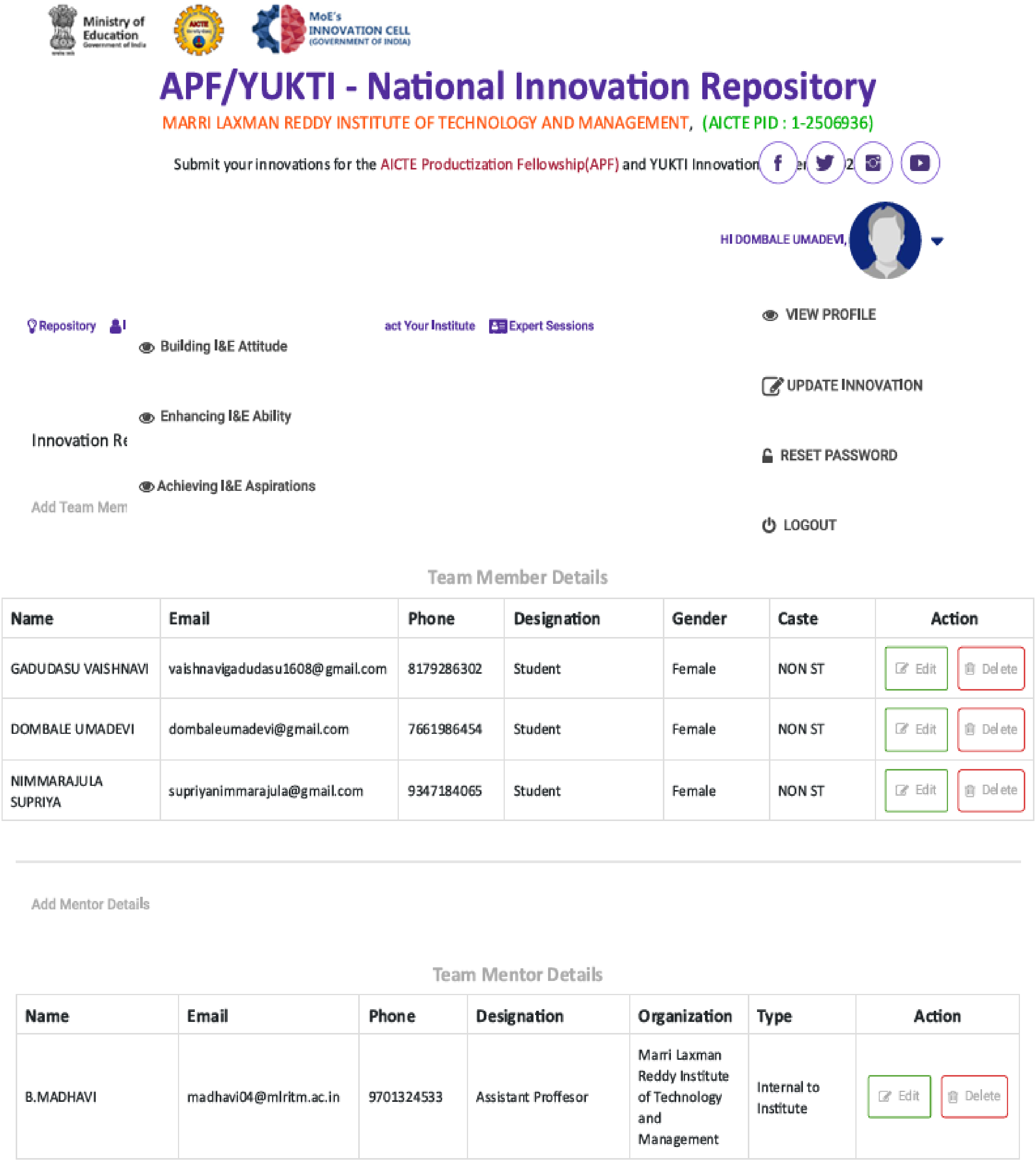
[6] Pinar Yildirim. (2015), “Filter based feature selection methods for prediction of risks in hepatitis disease”, International Journal of Machine Learning and Computing, Vol. 5, No. 4,pp. 258-263.

[7] Samina Khalid, TehminaKhalil and ShamilaNasreen. (2014), “A survey of feature selection and feature extraction techniques in machine learning”, Science and Information Conference(SAI)

[8] Instituto Nacional de Estadística (INE). Madrid, Spain. Fenómenos demográficos. Tablas de mortalidad. http://www.ine.es/jaxiT3/Datos.htm? t=27153 (April 2019, date last accessed).

[9] Ballester J, Robine JM, Herrmann FR, Rodó X. Effect of the Great Recession on regional mortality trends in Europe. *Nat Commun* 2019;8:679.

[10] Díaz R, Hernández-Vaquero D, Silva J, Pascual I, de la Hera JM, Leon V et al. Real structural valve deterioration of the mitroflow aortic prosthesis: competing risk analysis. *Rev Esp Cardiol (Engl Ed)* 2017;70:1074–81



APF/YUKTI - National Innovation Repository

MARRI LAXMAN REDDY INSTITUTE OF TECHNOLOGY AND MANAGEMENT, (AICTE PID : 1-2506936)

Submit your innovations for the AICTE Productization Fellowship(APF) and YUKTI Innovation

Team Member Details

Name	Email	Phone	Designation	Gender	Caste	Action
GADUDASU VAISHNAVI	vaishnavigadudasu1608@gmail.com	8179286302	Student	Female	NON ST	Edit Delete
DOMBALE UMADEVI	dombaleumadevi@gmail.com	7661986454	Student	Female	NON ST	Edit Delete
NIMMARAJULA SUPRIYA	supriyanimmarajula@gmail.com	9347184065	Student	Female	NON ST	Edit Delete

Team Mentor Details

Name	Email	Phone	Designation	Organization	Type	Action
B.MADHAVI	madhavi04@mlritm.ac.in	9701324533	Assistant Proffesor	Marri Laxman Reddy Institute of Technology and Management	Internal to Institute	Edit Delete