

PREDICTION OF LUNG CANCER PATIENT SURVIVAL USING MACHINE LEARNING TECHNIQUES

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Engineering degree in Computer Science and

Engineering

By

Indumathi (38110206)

Harini (38110194)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade “A” by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY – 2022



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC
(Established under Section 3 of UGC Act, 1956)
JEPPIAAR NAGAR, RAJIV GANDHI SALAI
CHENNAI– 600119
www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Harini R (38110194)** and **Indumathi T (38110206)** who carried out the project entitled “**Prediction of Lung cancer patient survival using Machine learning techniques**” under my supervision from DEC 2021 to MAY 2022.

Internal Guide

Dr. L. Lakshamanan

Head of the Department

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We, **Harini R and Indumathi T** hereby declare that the project report entitled **Prediction of lung cancer patient survival using machine learning techniques** done by us under the guidance of **Dr. L.Lakshmanan** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

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

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D, Dean, School of Computing, Dr. L. Lakshmanan, M.E., Ph.D., and Dr. S. Vigneshwari, M.E., Ph.D. Heads of the Department of Computer Science and Engineering** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. L. Lakshmanan, M.E., Ph.D.,** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

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

ABSTRACT

Sound health is indispensable for every individual to live a robust life. To lead a conventional and healthy life, an individual is supposed to diagnose their physiques constantly. The lung is one of the vital organs which contributes predominantly to the process of respiration. In particular for lung cancer, it is not well understood which types of techniques would yield more predictive information, and which data attributes should be used in order to determine this information. Lung carcinoma is now mutated into a miserable routine among mankind. In accordance with the American Cancer Society, 235,760 new instances of lung cancer are evaluated for the year 2021 which also incorporates 131,880 deaths. It has been claimed that cancer is curable when it is diagnosed in the initial stage. The proposed system assists oncologist in predicting patient survival using miscellaneous machine learning algorithms. As part of this study, a number of supervised learning techniques were applied to the SEER database to classify lung cancer patients by survival, including logistic regression, decision trees, shallow neural network, support vector machines, and an ensemble. Key data attributes were tumor size, stage, age, genetic risk, smoker and Gender. Based on the evaluation, logistic regression provided an accuracy of 96%. The target of the system is to estimate the most accurate algorithm to classify the lung cancer patient survival period.

LIST OF FIGURES

Figure No.	Figure name	Pg.no
4.1	Survival season of cellular breakdown	10
4.2	Selected data attributes along with description from SEER	10
4.3	Architecture of machine learning	11
4.4	System flow diagram	14
5.1	Survival prediction graph	19
5.2	Basis of field examination	20

TABLE OF CONTENTS

Chapter no.	Title	Page no.
	Abstract	i
	List of figures	ii
1	INTRODUCTION	1
2	LITERATURE SURVEY	4
3	AIM AND SCOPE	
	3.1 Aim	6
	3.2 Scope	6
4	METHODOLOGY AND IMPLEMENTATION	
	4.1 Technology Used	
	4.1.1 Hardware requirements	7
	4.1.2 Software requirements	7
	4.1.3 Python	7
	4.2 Steps Involved	8
	4.3 Machine Learning	10
	4.4 Algorithm	
	4.4.1 Logistic Regression	13
	4.4.2 Decision Tree	14
	4.4.3 Support Vector Machine	15
	4.4.4 Shallow neural network	16
	4.4.5 K-Nearest Neighbor	16
	4.4.6 Custom Ensemble	16
5	RESULT AND FUTURE WORKS	
	5.1 Results	18
	5.2 Performance Analysis	18
6	CONCLUSION	21
	REFERENCES	22
	APPENDIX	
	1. Code	27
	2. Ouput Screenshots	45

CHAPTER 1

INTRODUCTION

In machine learning, algorithms are designed to learn and improve with more experience. By applying machine learning to lung cancer prediction models, clinicians can help manage incidental cases of lung cancer. The use of machine learning techniques to estimate disease outcomes can be challenging, however, since patient data is generally not publicly available. The Surveillance, Epidemiology, and End Results (SEER) database program of the National Cancer Institute (NCI) makes exceptions, however. This dataset is the largest publicly available dataset on cancer. The database contains de-identified information on cancer statistics. The SEER Program provides information on cancer statistics to help reduce the cancer burden among the U.S. population. SEER has been funded by the NCI since 1973. SEER collects and publishes cancer incidence and survival data from population-based cancer registries covering approximately 34 percent of the U.S. population. These registries routinely collect data on patient demographics, primary tumor site, tumor morphology and stage at diagnosis, first course of treatment, and follow-up for vital status (survival). These data are collected on every cancer case reported from 19 U.S. geographic areas.

Lung cancer generally occurs in both male and female due to uncontrollable growth of cells in the lungs. This causes a serious breathing problem in both inhale and exhale part of chest. Cigarette smoking and passive smoking are the principal contributor for the cause of lung cancer as per world health organization. The mortality rate due to lung cancer is increasing day by day in youths as well as in old persons as compared to other cancers. Even though the availability of high tech Medical facility for careful diagnosis and effective medical treatment, the mortality rate is not yet controlled up to a good extent. Therefore it is highly necessary to take early precautions at the initial stage such that it's symptoms and effect can be found at early stage for better diagnosis. Machine learning now days has a great influence to health care sector because of its high computational capability for early prediction of the diseases with accurate data analysis. In our paper we have analyzed various

machine learning classifiers techniques to classify available lung cancer data in UCI machine learning repository in to benign and malignant. The input data is prepossessed and converted in to binary form followed by use of some well-known classifier technique.

Our goal is to assess the predictive power of machine-learning techniques using data specific to lung cancer. Lung carcinoma is the leading cause of cancer deaths around the world. The disease is normally classified as Small Cell Lung Cancer (SCLC) or Non-Small Cell Lung Cancer (NSCLC), with the diagnosis being based on cellular characteristics evaluated through histopathological examination. It is critical to provide clinical providers and patients with information regarding the chances of survival based on a specific medical diagnosis. Survival time analysis may be considered clinically important for predicting patient outcomes, but clinicians have had difficulty estimating lung cancer patient survival times.

The main weight of ailment overall is as Lung malignancy that is the most inescapable disease in the two men and women. A few other reports estimate some 221,200 new cases of pulmonary cancer occur and represent approximately 13% of all cancer diagnoses in 2015. Approximately 27 percent of all cancer deaths are attributed to lung cancer. Lung nodules must therefore be closely examined and monitored when at an early stage. In this study, the ML & DL approaches used in cancer progression modeling are reviewed. The predictive models discussed here are based on different supervised ML techniques, input and data samples.

Applying a correlational approach via machine learning to predict survivability could help to improve such predictions. In this study, patients diagnosed with lung cancer during the years 2011–2020 were selected in order to be able to predict their survival time. A number of supervised learning methods was employed to classify patients based on survival time as function of key attributes and, thus, help illustrate the predictive value of the various methods.

Registrars and residents of physician training programs predicted median survival times of 21.4 and 21.5 months, respectively, for patients on average with an average survival time of 11.7 months. Additionally, only about 60% of patients with physicians'

estimated survival times > 3 months actually lived that long. Researchers found that 10% of the time physicians were correct in their predictions of survival times up to the month, 59% of the time they were accurate at 3 months, and 71% of the time they were inaccurate at 4 months, and they tended to overestimate short-term survival but underestimate long-term survival times. The application of machine learning-based correlations to predict survivability may enhance these predictions.

A sample of lung cancer patients diagnosed between 2011 and 2020 was selected for this study in order to estimate survival times. The survival time of patients was classified using supervised learning methods based on key attributes and the results illustrated the predictive power of the different methods. The techniques chosen were Logistic regression, Decision Trees, Support vector machine, shallow neural network, and KNN. Furthermore, the analysis of the lung cancer patient data allows comparison of the predictive value of these methods using the selected attributes. The data in this study comprises measurements obtained at or near the time of diagnosis, offering the potential for a more proactive set of survival predictors.

This report also enables comparing the predictive value of the methods when applied with the chosen attributes to analyse the lung cancer patient data. The data-set in this study focuses on measurements available at or near the time of diagnosis, which represents a more proactive set of survival predictors.

CHAPTER 2

LITERATURE SURVEY

Innumerable surveys have been conducted regarding lung cancer patient survival rates. The distributed information was breaking down in the SEER data set utilizing factual and consecutive techniques. The primary review utilized a bunch to bunch a gathering of malignant growth patients. In many regards, the Cancer Data Grouping Algorithm has been proposed as a method for foreseeing results and coordinate patients as indicated by endurance standards. Huge informational index in SEER data set connected with cancer size, size, and lymph hub structure.[1-6]

In various progressed bosom disease avoidance strategies [7], two techniques for information extraction calculations have been utilized: muscle and choice trees, just as obsolete, factual techniques. An out of line computation was acquired utilizing the cross multiple times to look at the exhibition of the three strategies for prescience. For the example got, Decision Tree (C5) was 93.6% valid, the neural organization was 91.2% great, and the opposite model was 89.2%. In a review [8], prostate disease endurance was concentrated on utilizing helped helps (SVMs) and three methods recently depicted. The outcomes showed that SVM was the best indicator (92.85% valid), trailed by the choice tree and the extraordinary organization. Moreover, [9] he was determined to have prostate malignant growth utilizing muscle strands, choice sticks, and backslides. In view of SEER information on patients with colorectal malignant growth [10], you looked at changed wellbeing indicators and observed that muscle networks were sensible. [11] Demonstrating a mix of the three practices with the best presentation is a superior indicator of colorectal disease and is nearer to the line of Receiver Operating Characteristic (ROC).

To survey the wellbeing of patients with cellular breakdown in the lungs, various investigations have been directed utilizing AI procedures to dissect the SEER information base, including bunch coordination [12], SVM and strategic relapse [13], and uncontrolled techniques [14]. The capacity to foresee the danger of cellular breakdown in the lungs in patients with explicit side effects is considered in [15]. As per [16-20], the normal future of patients with cellular breakdown in the lungs in the SEER information base was assessed to be ~ 90% for C4.5 and Naive-Bayes.

Choice of the five parts of the request in view of grouping and meta-order [21] observed that majority determination was essentially improved with a region beneath the ROC line while foreseeing cellular breakdown in the lungs.

Mining was utilized as a strategy for deciding the intriguing associations and connections among objects, and different methods and measures were needed to find the best rundown of rules and models for a given number of boundaries [22]. The area tree, in light of space information, was created in [23] involving a robotized technique for identifying cellular breakdown in the lungs, some of which are various and physically chose. They utilized insatiability to get into numbers and profound techniques utilizing a tree in light of an informational index-based calculation. The review thought about three factors: the way to building up a branch, adding new branches, and the components used to add another branch. The government rules apply to each tree hub.

In spite of the fact that there is no data on chemotherapy, it is feasible to survey the significance of treatment and medical procedure by breaking down the SEER information base. The remedial impact is examined in [24]. The reason for this review is to decide if patients with cellular breakdown in the lungs live longer with a medical procedure or radiation. The change of the considered covariates to the Propensity score is steady with the suspicion that the middle is given by the part founded on the characterized covariates. In light of the outcomes got and the tree grouping, the determined scores were analyzed utilizing two techniques. Since patients could go through radiation and medical procedure, scores were determined for each gathering and their qualities were observed. Each gathering was carried out utilizing direct time and radiation insights. These individuals who didn't get radiation lived significantly longer, whether or not they got therapy [25].

CHAPTER 3

AIM AND SCOPE

3.1 AIM

This project aims to predict the survival period of a lung cancer patient using machine learning techniques.

3.2 SCOPE

The prominent cause of cancer-related mortality throughout the globe is “Lung Cancer”. Hence beforehand detection, prediction and diagnosis of lung cancer has become essential as it expedites and simplifies the consequent clinical board. To erect the progress and medication of cancerous conditions machine learning techniques have been utilized because of its accurate outcomes. The disease has a tendency to be asymptomatic mostly in its earlier stages thus making it nearly impossible to detect. That's why early cancer detection plays an important part in saving lives. An early detection can give a patient a better chance to cure and recover. Various types of machine learning algorithms (ML) like Decision Tree, Support Vector Machine (SVM), Logistic regression, Shallow Neural Network (SNN), KNN and Custom Ensemble have been applied in the healthcare sector for analysis and prognosis of lung cancer. In this review, factors that cause lung cancer and application of ML algorithms are discussed up to date and also draws special attention to their relative strengths and weaknesses. This paper will help the researchers to quickly go through the related literature instead of referring to the many papers. Therefore it is highly necessary to take early precautions at the initial stage such that it's symptoms and effect can be found at early stage for better diagnosis. Machine learning now days has a great influence to health care sector because of its high computational capability for early prediction of the diseases with accurate data analysis.

CHAPTER 4

METHODOLOGY AND IMPLEMENTATION

4.1 TECHNOLOGY USED

4.1.1 HARDWARE REQUIREMENTS

- System : Pentium IV 2.4 GHz.
- Hard Disk : 40 GB.
- Monitor : 15 inch VGA Color.
- Mouse : Logitech Mouse.
- Ram : 512 MB
- Keyboard : Standard Keyboard

4.1.2 SOFTWARE REQUIREMENTS

- Operating System : Windows XP.
- Platform : PYTHON TECHNOLOGY
- Tool , Python 3.5
- Back End : python anaconda script

4.1.3 PYTHON

Python is a **high-level, interpreted, interactive** and **object-oriented scripting language**. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted:** Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive:** You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

- **Python is Object-Oriented:** Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language:** Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.
- **Python History:** Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands. Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages. Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL). Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

4.2 STEPS INVOLVED

- **Data Preprocessing**

Data mining is a process of extracting meaningful data from a raw dataset. The different stages of data mining are data cleaning where we can insert some attribute to the missing values, or remove the duplicate value, then data integration which can be define as heterogeneous data from multiple source combined in a common source that is data warehouse. After data integration data we transform the data into appropriate form required by mining process this step is called data transformation, it includes binning, aggregation, normalization, clustering. Next it is followed by data reduction where we reduce the dataset into smaller volume with respect to maintain the integrity of original dataset, we can reduce the high dimensional data using principle component analysis and by numerosity reduction using histogram, sampling and clustering. The last step for pre-processing is data discretization which a part of data reduction and it is important for numerical data, it involve following methods binning, histogram, entropy based and clustering. After pre-processing feature extraction or feature selection can be done by using any

learning algorithm like K nearest neighbor or K mean then on the basis of feature Machine learning algorithm can be applied.

- **Selection of patient attributes**

A set of key attributes was selected that consisted of both continuous (numeric) and categorical (discrete) variable types, for a total list of 19 attributes (including the target “Survival Time,” Figure 2). The models were then used to correlate the remaining 18 attributes to the patient outcome, i.e., survival time. The data taken from the SEER database is from patients diagnosed from the years 2011–2020. Beginning in 2011, both stage and grade criteria were changed in definition, which makes 2011 the first available year for consistent five-year survival time evaluation. The techniques employed in this study are focused on creating a correlation between patient survival time and the attributes listed. Therefore, the range of survival time is from 0 to 72 months, and the output from the models is expected to be within this range.

- **Comparison of predicted to actual survival times:**

Besides the weighting factor and RMSE values, the standard deviation of the predicted values respective to each model may provide further insight. Fig. 1 shows that the predicted values line up with the actual values well for low (~5 months) to moderate values (~30 months). Past the 30–35 month mark, however, the predicted values do not line up well with the actual values, as ideally these graphs would show a linear correlation of predicted to actual values. A nearly exponential curve is delineated around 30 months for every model with the exception of Decision Trees, for which the model could not find sufficient discrete branching or splitting points due to the size of the data and its vague nature. For comparison to a classical technique, a Cox proportional hazards model of survivability for the validation data actors the Age, Stage, Grade and Tumor Size groupings. Each frame has multiple lines showing the decrease in survival probability for each value, e.g., Stage has lines for stages 0, IA, IB, IIA, etc.

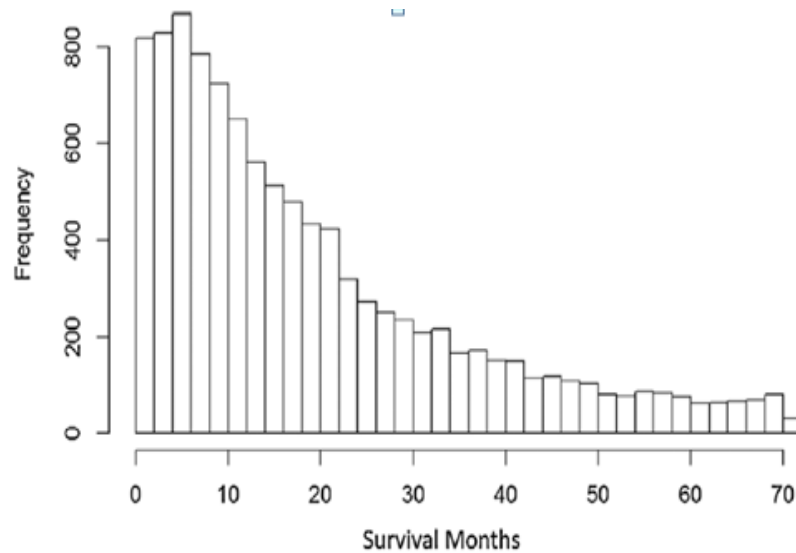


Figure 4.1 survival season of cellular breakdown

Number	Attribute	Description	Type
1	Age	Age at time of diagnosis.	Discrete
2	Grade	Appearance of cancer cells and how fast they may grow.	Numeric
3	Radiation Sequence with Surgery	Order of surgery and radiation therapy administered for patients who received both.	Numeric
4	Number of Primaries	Number of malignant tumors other than lung.	Discrete
5	T	AJCC component describing tumor size.	Numeric
6	N	AJCC component describing lymph node involvement.	Numeric
7	M	AJCC component describing tumor dissemination to other organs.	Numeric
8	Radiation	Indication of whether patient has received radiation.	Numeric
9	Stage	Stage of tumor – based on T, N, and M.	Numeric
10	Primary Site	Location of tumor within the lungs.	Numeric
11	First Malignant Primary Indicator	Based on cancers reported in SEER database for patient.	Numeric
12	Sequence Number	Order of lung cancer occurrence with respect to other cancers for this patient.	Discrete
13	CS Lymphnodes	Number of lymph nodes involved.	Numeric
14	Histology Recode – Broad Groupings	Microscopic composition of cells and/or tissues for specific primary. Used for staging and treatment determination.	Numeric
15	RXSumm – ScopeRegLNSur(2003+)	(Scope of Regional Lymph Node Surgery) – Procedure of removal, biopsy, or aspiration of regional lymph nodes.	Numeric
16	RXSumm – SurgPrimSite(1998+)	(Surgery of Primary Site) – Procedure to remove or destroy tissue of the primary site.	Numeric
17	DerivedSS1977	This item is derived “SEER Summary Stage 1977” from the Collaborative Stage (CS) algorithm, effective with 2004+ diagnosis.	Numeric
18	TumorSize	Measurement of tumor size.	Numeric
19	Survival Time	Number of months that patient is alive from date of diagnosis.	Discrete

Figure 4.2 Selected data attributes along with description from SEER

4.3 MACHINE LEARNING

Machine learning is a subset of Artificial Intelligence the main difference between artificial intelligence and machine learning is that artificial intelligence allows the computer to think like human whereas machine learning is a process where system learns from its past

experience. In machine learning a set of data is trained using different machine learning methods which is called as training dataset. Once the training is complete then the machine is feed with new dataset which is called testing data set. Now machine can predict the object based on its training data. Machine learning is categorized into three categories supervised, unsupervised and reinforcement learning. Fig-3 demonstrates the basic machine learning architecture.

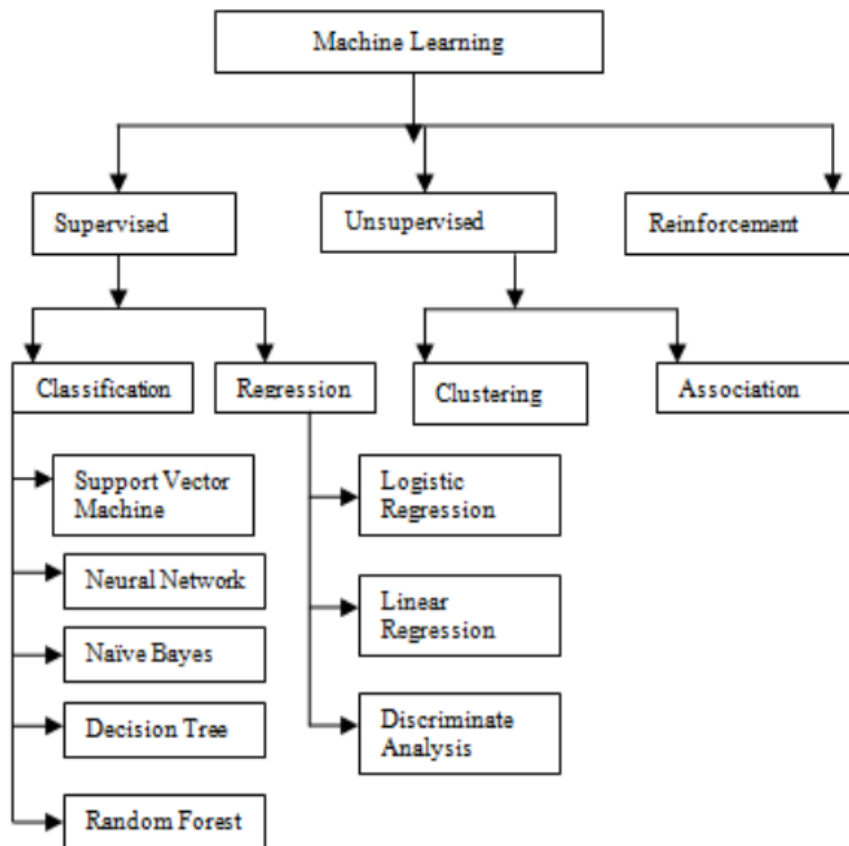


Figure 4.3 architecture of machine learning

Learning is categorized into three categories:

1) Supervised learning: Supervised learning is a learning in which we train the dataset which are already marked with correct answer (labeled data) after training we provide a new data that is test data set so that supervised approach can analysis the testing data set and based on the training data set it can provide correct result. Supervised learning is classified into two categories Classification (when the output variables are categorical (or discrete)) and Regression (when the output variable is continuous (or numerical)).

2) Unsupervised Learning: Unsupervised learning is a training of machine where machine learn by itself that is no information is given to the machine or we can say the data are not labeled so it allow the algorithm to act without any guidelines so it learn by the taken the similarities, patter and difference between two classes form a group for example a student learning itself based on its experience. Under supervised learning clustering in one of the technique, clustering can be defined as forming groups based on their types, and second is association which describe how thing are related to each other.

3) Reinforcement learning: It is based on reward and environment, its objective is to give best solution on the bases of maximum reward. The environment first send a state to the agent based on its knowledge to take the action, once the action take place environment send a pair of next state and reward back to the agent. The agent updates its knowledge with the reward. This continues till it collect maximum reward or we can say the algorithm is properly trained. Here the environment refers to object and agent refers to reinforcement learning algorithm. This concept is used in game theory.

4.4 ALGORITHM

Only pre-processing the data is not sufficiency because it need to be classify whether the patient is suffering from cancer or not, what is its survival rate, if the patient is suffering from tumor then tumor is in which stage so that based on all this question it is very much important to have accurate prediction so that based on the prediction treatment can be given to the patient. For predicting the dataset various kind of machine learning techniques are used like support vector machine, neural network, decision tree etc, all the algorithm have different approaches of prediction and it is also possible that one algorithm can give better accuracy over another algorithm. The techniques utilized in this review were joined with logistic regression, Decision Trees, KNN, Support Vector Machine (SVM) using a polynomial kernel function of a non-probabilistic method, and shallow neural networks. The models displayed here show a rundown of present day and normally utilized, yet there are many control advances. This proposal's system architecture is illustrated in Figure 3. Execution is in Python,

since a broadly useful language can do a great deal of AI. Endurance investigation recognizes data scores from successive AI in view of future rather than foreseeing future. RMSE esteem was utilized for each model to look at and test the models. The RMSE esteem model is the near distance between the speculation and the real outcome. Furthermore, in view of the precise portrayal of each model, the particular load of the majority was considered when looking at the models.

4.4.1 LOGISTIC REGRESSION

Logistic regression is used when the dependent variable are in binary in nature, it is widely used in classification problem by using probability. The value of the dependent variable ranges from 0 to 1 and it can be represented by an equation that is probability of event occurrence divided by probability of not event occurrence. Logistic regression does not require linear relationship because it applies a nonlinear log transformation to predict odd ratio. At the same time it is important that the independent variable should not be correlated with each other. Logistic regression is a method of foreseeing the future utilizing potential standards. Stage subordinate changes anticipated by inversion. Thusly, the odds of accomplishing the best outcomes are anticipated. Because of the job of repeat in various parallel variables, patient endurance was changed by 1/0 each time the example was evaluated and the retreat in the media. Variable years are viewed as ceaseless whole numbers, and the pace of progress is viewed as the amount of the factors in the three states. One of the calculations picked was inversion, as the variable-based investigation was presence utilizing two qualities to better contrast with the model. Logistic regression has been used to predict the death rate due to lung cancer. As the dataset has only two attribute either yes or no for which logistic regression can work more efficiently than other algorithms.

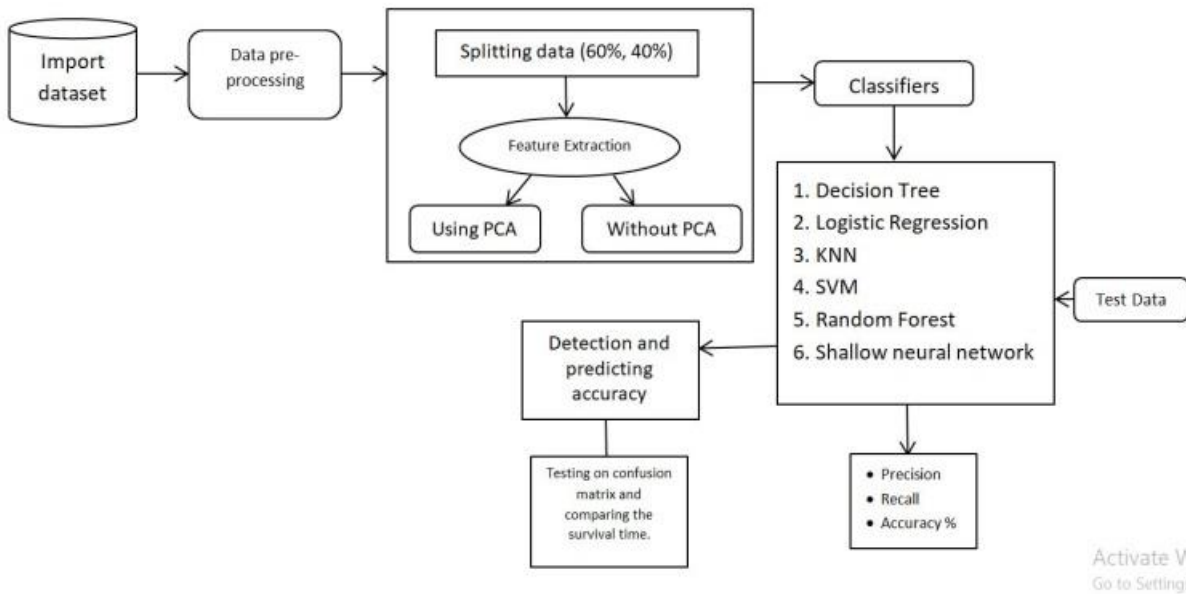


Figure 4.4 System flow diagram

4.4.2 DECISION TREE

Layouts are made of wood utilizing a Tree Certificate. This innovation simultaneously lessens how much data implanted in little information, and joins the information to consolidate this data with the choice tree, bringing about the making of a leaf or an end, bringing about the sharing of data. We can clarify the parts here. For this situation, the life not entirely settled. As indicated by the converse trees, the tree arranging choices were more successful than the customary tree. Cross-segments were performed utilizing a python sklearn library, and the tree was chosen with a greatest tallness of 10, a partition of at minimum, a profundity of somewhere around 35, and an intricacy of 0.1. We utilized normal qualities for the two things. Online inquiry didn't show a critical distinction in yield. Albeit the measures have been transformed, we have chopped down the choice tree to a profundity of three levels, and we have not had the option to make it troublesome or power a decent score, showing the basic idea of this technique.

As per the exceptional backwoods' strategy, certificate trees can be planted during preparing and may differ beginning to end. Utilizing algorithmic memory techniques, dynamic gatherings can be joined to give preferred consistency over explicit trees. As

such, the choice tree depends on the set-up data, if any - around then - that is, the law is clear, which makes it one of the most intelligent. Unnecessary adjusting is one of the issues connected with a similar choice tree, which makes the thoughts great examine practice, yet not really great for genuine numbers. Despite the fact that our intersection strategy prepared one tree for preparing, the choice tree included ten degrees of cross-line intersection to guarantee precision. Since all tree up-and-comers had comparative execution, the full tree was utilized to contrast it and different techniques. In extraordinary woods, the quantity of trees was set at 500 and the decision of hand, and it was found that the exhibition underneath this number stayed stable. In the wake of trying different things with various numbers, we chose to have no less than 50 hubs in the wake of taking a gander at its little impacts. Entropy has been used to make decision whether the tumor is normal or abnormal. The information gain value is calculated for the process decision making by using all the input image features such as transition regions, range, morphological region, pixels as items sets. These all features with the highest normalized information gain value have been selected for making decision. Decision tree has been applied on clinical dataset which consist of attributes like gender, age, reformed smoker, pack-year, race, length of smoking. Raw data has been inputted one by one and one attribute is selected at a time for descriptive value dataset. If the selected attribute contain valid dataset then it is stored in valid value dataset otherwise it is removed. This process continues till all descriptive value is used. So here decision tree has been used for pre-processing of clinical dataset.

4.4.3 SUPPORT VECTOR MACHINE

SVMs are the double line of the bring line back. Assume that the preparation list is one of two sections, the calculation shows the hyper plane a long way from the two sets and analyzes each set-in space. At the point when line partition is preposterous, the programmed technique is utilized to look for disconnected guides. Since hyper = planes highlighting non-straight limits, SVMs have the best trouble in associating when the information space is commotion. SVM is used to detect whether the tumour is malignant or benign. After extracting the feature, those features were used to train SVM. Based on training classification is done and after prediction on the predicted

output scatter plot is built. Kernel function is used when the data set are arranged in nonlinear way. SVM has been applied for the predicting the performance of the feature extraction algorithm.

4.4.4 SHALLOW NEURAL NETWORK

Shallow neural networks consist of only 1 or 2 hidden layers. SNNs are the double line of the bring line back. Assume that the preparation list is one of two sections; the calculation shows the hyper plane a long way from the two sets and analyzes each set-in space. At the point when line partition is preposterous, the programmed technique is utilized to look for disconnected guides. Since hyper -planes highlighting non-straight limits, SNNs have the best trouble in associating when the information space is commotion.

4.4.5 K-NEAREST NEIGHBOR

K-nearest neighbors (KNN) is a type of supervised learning algorithm used for both regression and classification. KNN tries to predict the correct class for the test data by calculating the distance between the test data and all the training points. Then select the K number of points which is closet to the test data. The algorithm can be used to solve both classification and regression problem statements. KNN is a strategy for figuring out various techniques in view of comparative scales. This is a straightforward, proficient and strong method. The calculation prepares countless information and performs well to decipher data paying little heed to the irrelevant nature. The request for the example not set in stone by the determination of the example closest to the example. The similar model is the predominant class of the chose factors. Many elements influence the exhibition of the KNN calculation, like tallness, number of neighbors, and weight. It expects to give its neighbors weight and make them more various.

4.4.6 CUSTOM ENSEMBLE

Ensemble method is a technique where it combine the series of K learned model (or base classifier) that focused on creating an improved prediction model. It use

decision tree classifier in a randomize way. A Training dataset of h tuple is given and for each iteration j a training set, H_i of h tuple is sampled with replacement from H (it work like a bootstrap). Which doing this some tuple may occur more than one tin and it may also happen some of the tuple does not occur at all. Once the bootstrap dataset is formed from the original dataset it used to make the decision tree. Let g be the number of attribute to determine the split at each node, where g is smaller than the available attribute and to build a decision tree classifier D_i randomly select node from g attribute as a candidate for splitting at the node. Cart is used to grow the tree and over fitting is not a issue in random forest. Its accuracy depends on the strength of the individual classifier. For great readiness, these models were joined utilizing an exceptional strategy. To accomplish this, we utilized exceptional woodland. Devoted woodlands join the quantity of trees to order various snippets of data, use equilibrium to all the more likely anticipate forecasts and control them. By and large, custom practices are more powerful than specific strategies, and gatherings are more straightforward to carry out and simpler to adjust to changes in plan. Random forest is one of the ensemble methods to estimate the performance of the performance of the model on new data, each dataset are trained thousand times on bootstrapped cohort's samples from training dataset. Due to feature extraction large amount of feature are high dimensional due to which over fitting issue has arisen to avoid this situation random forest has been used.

CHAPTER 5

RESULTS AND FUTURE WORKS

5.1 RESULTS

When the elements have been created, the return innovation is utilized to think about future. It incorporates an outline of exercises, blends of weighty articles, ordinary deviations, how to foresee future, just as standard deviations between the anticipated time and future. The normal endurance time for preparing and affirmation is 19.59 months and the normal seclusion time frame is 16.77 months. Cox's endurance model was contrasted and antiquated information approval strategies joined with age, stage, grade, and growth size (Figure 2). The vast majority of the lines in everything show a diminishing in feasibility, as displayed in Figure 1. For instance, the characterization is displayed as a column of classifications 0, IA, IB, IIA, etc. The vast majority of the lines on the left show a 19-month endurance rate. One of the vital discoveries of this review is that most of survivors change early, however by far most of survivors go too far in the red. Figure 4 shows an arrangement for coordinating reinforcement tasks, Decision Tree, KNN, and SVM to sensibly look at results. To assess the data set of patients with cellular breakdown in the lungs, a few destinations have been created to assess the impacts on patients, and the connection between unfavourably susceptible disease and cellular breakdown in the lungs is delineated in Figure 5. Logistic regression yielded the highest accuracy of 96%.

5.2 PERFORMANCE ANALYSIS

Despite the issues with the other models investigated, no model other than Decision Trees seems truly lacking, and the SVM being worthy of independent attention as it predicts a distinct distribution but scores similarly to the others. The performance of the classification methods applied in this study was comparable to that of classical techniques such as the Cox proportional hazards model, which contradicts the working hypothesis that the more modern methods can take any type of dataset and yield improved results.

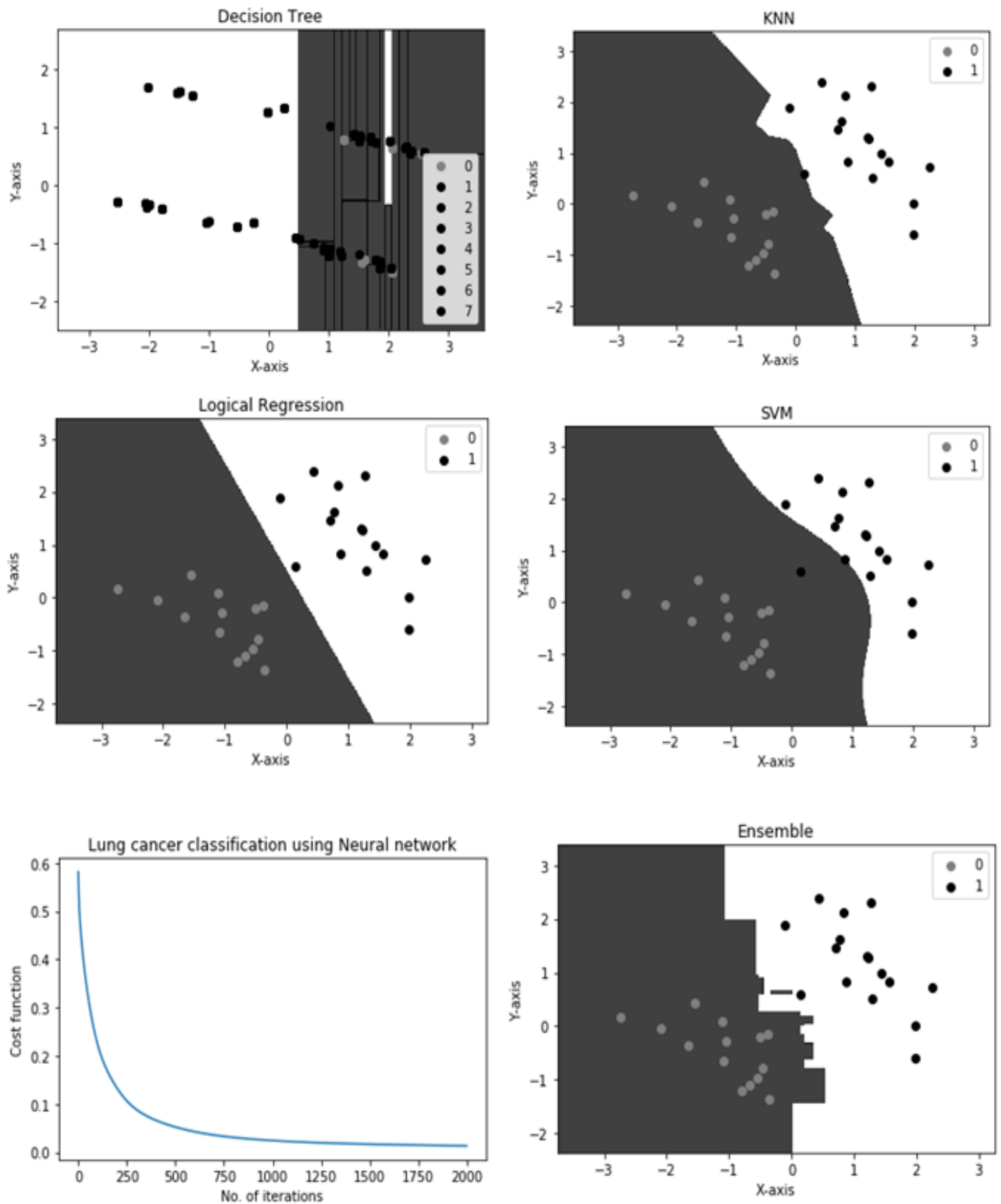


Figure 5.1 survival prediction graph

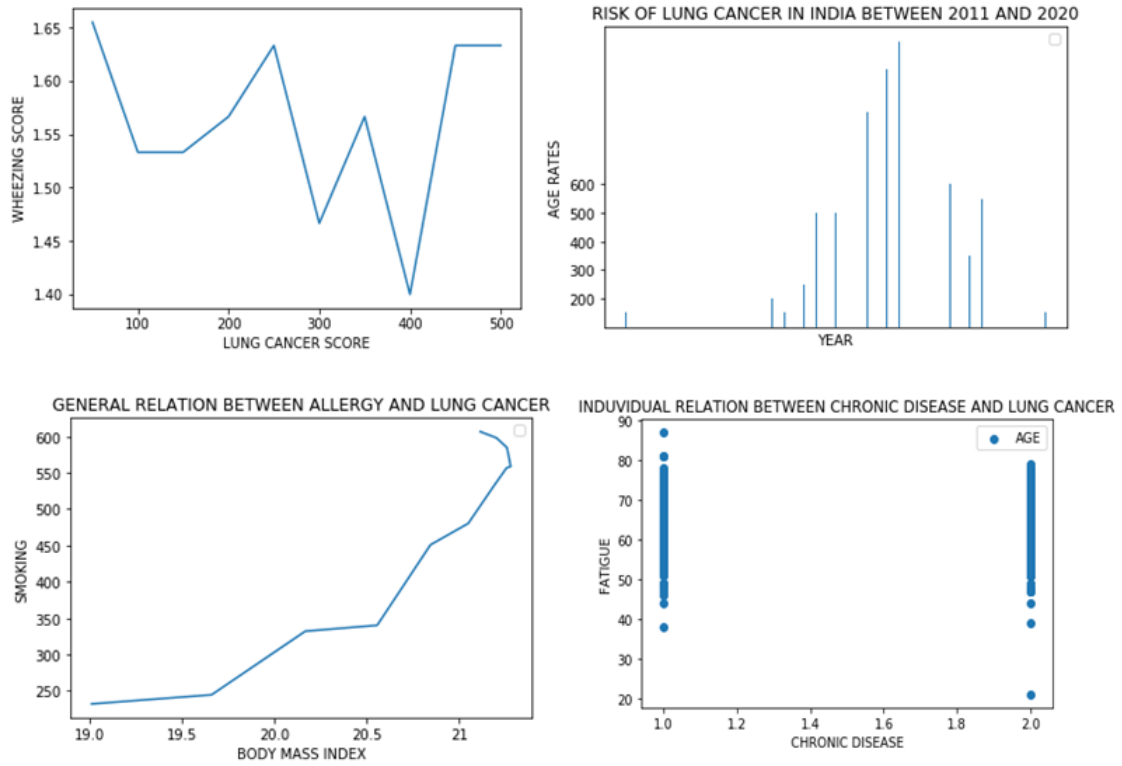


Figure 5.2 Basis for field examination

CHAPTER 6

CONCLUSION

The utilization of controlled gadgets to change the existences of patients with cellular breakdown in the lungs recommends that it can happen in an anticipated populace; The subsequent numbers can address the right lines. For this situation, choice trees can't be utilized on the grounds that they are less compelling. The presentation of the successive technique in this review was like more established advances, for example, the Cox corresponding risk model, which goes against the presumption that cutting edge strategies function admirably on a wide range of informational indexes. Because of these arrangements, antiquated methods can be similarly pretty much as helpful as current ones, so their exhibition should be estimated and contrasted with present day innovation. Boundary factors should be investigated in additional examinations.

REFERENCES

- [1] A. Agrawal, S. Misra, R. Narayanan, L. Polepeddi, A. Choudhary, Lung cancer survival prediction using ensemble data mining on SEER data, *Sci. Program.* 20 (2012).
- [2] A. Bhaskarla, P.C. Tang, T. Mashtare, C.E. Nwogu, T.L. Demmy, A.A. Adjei, et al., Analysis of second primary lung cancers in the SEER database, *J. Surg. Res.* 162 (2010) 1–6.
- [3] A clustering-based approach to predict outcome in cancer patients, in: K. Xing, D. Chen, D. Henson, L. Sheng (Eds.), *Sixth International Conference on Machine Learning and Applications (ICMLA)*, IEEE, 2007.
- [4] ACS_Cancer_Facts, *Cancer Facts & Figures*, American Cancer Society, 2015.
- [5] Amato, K.A.D., Hyland, A., Reed, R., Mahoney, M.C., Marshall, J., Giovino, G., Bansal-Travers, M., Ochs-Balcom, H.M., Zevon, M.A., Cummings, K.M. and Nwogu, C., 2015. Tobacco cessation may improve lung cancer patient survival. *Journal of Thoracic Oncology*, 10(7), pp.1014-1019.
- [6] Bueno, Raphael, et al. "Validation of a molecular and pathological model for five-year mortality risk in patients with early stage lung adenocarcinoma." *Journal of Thoracic Oncology* 10.1 (2015): 67-73.
- [7] Bueno, R., Hughes, E., Wagner, S., Gutin, A.S., Lanchbury, J.S., Zheng, Y., Archer, M.A., Gustafson, C., Jones, J.T., Rushton, K. and Saam, J., 2015. Validation of a molecular and pathological model for five-year mortality risk in patients with early stage lung adenocarcinoma. *Journal of Thoracic Oncology*, 10(1), pp.67-73.
- [8] Chip M. Lynch, Behnaz Abdollahi, Joshua D. Fuqua, Alexandra R. de Carlo, James A. Bartholomai, Rayeane N. Balgemann, Victor H. van Berkel, Hermann B. Frieboes 2021. Prediction of lung cancer patient survival via supervised machine learning. *International journal of medical informatics*.
- [9] Christodoulou, E., Ma, J., Collins, G.S., Steyerberg, E.W., Verbakel, J.Y. and Van Calster, B., 2019. A systematic review shows no performance benefit of machine learning over logistic regression for clinical prediction models. *Journal of clinical*

epidemiology, 110, pp.12-22.

[10] Christodoulou, Evangelia, et al. "A systematic review shows no performance benefit of machine learning over logistic regression for clinical prediction models." *Journal of clinical epidemiology* 110 (2019): 12-22.

[11] C.M. Lynch, V.H. van Berkel, H.B. Frieboes, Application of unsupervised analysis techniques to lung cancer patient data, *PLoS One* 12 (2017).

[12] Colon cancer survival prediction using ensemble data mining on SEER data, in: R. Al-Bahrani, A. Agrawal, A. Choudhary (Eds.), *IEEE Big Data Workshop on Bioinformatics and Health Informatics* (2013).

[13] D. Chen, K. Xing, D. Henson, L. Sheng, A.M. Schwartz, X. Cheng, Developing prognostic systems of cancer patients by ensemble clustering, *J. Biomed. Biotechnol.* 2009.

[14] D. Delen, Analysis of cancer data: a data mining approach, *Expert Syst.* 26 (2009) 100–112.

[15] D. Delen, G. Walker, A. Kadam, Predicting breast cancer survivability: a comparison of three data mining methods, *Artif. Intell. Med.* 34 (2005) 113–127.

[16] D. Fradkin, *Machine Learning Methods in the Analysis of Lung Cancer Survival Data*, (2006).

[17] Finding survival groups in SEER lung cancer data. machine learning and applications (ICMLA), in: I. Skrypnik (Ed.), *11th International Conference On*; 2012, IEEE, 2012.

[18] G. Dimitoglu, J.A. Adams, C.M. Ji, Comparison of the C4: 5 and a naive bayes classifier for the prediction of lung cancer survivability, *J. Comput.* 4 (2012).

[19] Gomez, D.R., Liao, K.P., Swisher, S.G., Blumenschein, G.R., Erasmus Jr, J.J., Buchholz, T.A., Giordano, S.H. and Smith, B.D., 2015. Time to treatment as a quality metric in lung cancer: Staging studies, time to treatment, and patient survival. *Radiotherapy and Oncology*, 115(2), pp.257-263.

[20] Guo, Nancy L., et al. "Confirmation of gene expression–based prediction of survival in non–small cell lung cancer." *Clinical Cancer Research* 14.24 (2008): 8213-

8220.

[21] Huang, Lin, et al. "Machine learning of serum metabolic patterns encodes early-stage lung adenocarcinoma." *Nature communications* 11.1 (2020): 1-11.

[22] Huang L, Wang L, Hu X, Chen S, Tao Y, Su H, Yang J, Xu W, Vedarethinam V, Wu S, Liu B. Machine learning of serum metabolic patterns encodes early-stage lung adenocarcinoma. *Nature communications*. 2020 Jul 16;11(1):1-1.

[23] Identifying hotspots in lung cancer data using association rule mining, in: A. Agrawal, A. Choudhary (Eds.), 11th International Conference on Data Mining Workshops (ICDMW), IEEE, 2011.

[24] Johnson, B.E., 1998. Second lung cancers in patients after treatment for an initial lung cancer. *JNCI: Journal of the National Cancer Institute*, 90(18), pp.1335-1345.

[25] Knowledge extraction from prostate cancer data, in: D. Delen, N. Patil (Eds.), 39th Hawaii International Conference on System Sciences, Hawaii, 2006.

[26] Mining association rules between sets of items in large databases, in: R. Agrawal, T. Imieliński, A. Swami (Eds.), *SIGMOD Record*, ACM, 1993.

[27] M.J. Thun, L.M. Hannan, L.L. Adams-Campbell, P. Boffetta, J.E. Buring, D. Feskanich, et al., Lung cancer occurrence in never-smokers: an analysis of 13 cohorts and 22 cancer registry studies, *PLoS Med.* 5 (2008) e185.

[28] N.A. Noohi, M. Ahmadzadeh, M. Fardaer, Medical data mining and predictive model for colon cancer survivability, *Int. J. Innov. Res. Eng. Sci.* (2013) 2.

[29] NCI_Lung_Cancer_Info, What You Need To Know About Lung Cancer, National Cancer Institute, 2015 (Available from: <http://www.cancer.gov/publications/patient-education/wyntk-lung-cancer>).

[30] NCI_SEER_Overview, Overview of the SEER Program. Surveillance Epidemiology and End Results, (2015) (Available from: <http://seer.cancer.gov/about/>).

[31] NCI_SEER_Training_Lung_Cancer_Stats, Introduction to Lung Cancer. SEER Training Modules, National Cancer Institute, 2015 (Available from: <http://training.seer>.

cancer.gov/lung/).

[32] N. Kapadia, F. Vigneau, W. Quarshie, A. Schwartz, F. Kong, Patterns of practice and outcomes for stage I non-small cell lung cancer (NSCLC): analysis of SEER-17 data, 1999–2008, *Int. J. Radiat. Oncol.* Biol.* Phys.* 84 (2012) S545.

[33] Nusinovici, S., Tham, Y.C., Yan, M.Y.C., Ting, D.S.W., Li, J., Sabanayagam, C., Wong, T.Y. and Cheng, C.Y., 2020. Logistic regression was as good as machine learning for predicting major chronic diseases. *Journal of clinical epidemiology*, 122, pp.56-69.

[34] Russell, Prudence A., et al. "Does lung adenocarcinoma subtype predict patient survival?: A clinicopathologic study based on the new International Association for the Study of Lung Cancer/American Thoracic Society/European Respiratory Society international multidisciplinary lung adenocarcinoma classification." *Journal of Thoracic Oncology* 6.9 (2011): 1496-1504.

[35] SEER_Program. Surveillance, Epidemiology, and End Results (SEER) Program (www.seer.cancer.gov) Research Data 1973–2009, National Cancer Institute, DCCPS, Surveillance Research Program, Surveillance Systems Branch, released April 2012, based on the November 2011 submission.

[36] S. Ramalingam, K. Pawlish, S. Gadgeel, R. Demers, G. Kalemkerian, Lung cancer in young patients: analysis of a Surveillance, Epidemiology, and End Results database, *J. Clin. Oncol.* 16 (1998) 651–657.

[37] Thomas, A., Chen, Y., Yu, T., Jakopovic, M. and Giaccone, G., 2015. Trends and characteristics of young non-small cell lung cancer patients in the United States. *Frontiers in oncology*, 5, p.113.

[38] V. Krishnaiah, G. Narsimha, N.S. Chandra, Diagnosis of lung cancer prediction system using data mining classification techniques, *Int. J. Comput. Sci. Inf. Technol.* 4 (2013).

[39] Yoshida, Junji, et al. "Visceral pleura invasion impact on non-small cell lung cancer patient survival: its implications for the forthcoming TNM staging based on a large-scale nation-wide database." *Journal of Thoracic Oncology* 4.8 (2009): 959-963.

[40] Y. Wu, Propensity Score Analysis to Compare Effects of Radiation and Surgery on Survival Time of Lung Cancer Patients from National Cancer Registry (SEER) [Master's], Epidemiology and Biostatistics: School of Public Health, SUNY-Albany, 2006.

APPENDIX

1 CODE

a) Main.py

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
#
dataset =pd.read_csv('lung_cancer_examples.csv')
X=dataset.iloc[:,2:6].values
y=dataset.iloc[:,6].values
#
from sklearn.preprocessing import LabelEncoder
y= LabelEncoder().fit_transform(y)
#
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test
=train_test_split(X,y,test_size=0.5,random_state=0)
#
from sklearn.preprocessing import StandardScaler
sc_X= StandardScaler()
X_train=sc_X.fit_transform(X_train)
X_test=sc_X.transform(X_test)
#
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators=10,
criterion='entropy',random_state=0)
classifier.fit(X_train,y_train)
#
Y_pred = classifier.predict(X_test)
#
from sklearn.metrics import confusion_matrix
```

```

cm = confusion_matrix(y_test,Y_pred)
#
print("Accuracy of the Model is :
",(cm[0][0]+cm[1][1])*100/(cm[0][0]+cm[1][1]+cm[0][1]+cm[1][0]))
print("Precision of the Model is : ",(cm[0][0])*100/(cm[0][0]+cm[1][0]))
print("Recall of the Model is : ",(cm[0][0])*100/(cm[0][0]+cm[0][1]))
#
from sklearn.decomposition import PCA
pca= PCA(n_components=2)
X_train = pca.fit_transform(X_train)
X_test = pca.transform(X_test)
explained_variance= pca.explained_variance_ratio_
#
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators=10,
criterion='entropy',random_state=0)
classifier.fit(X_train,y_train)
#
Y_pred = classifier.predict(X_test)
#
from matplotlib.colors import ListedColormap
X_set,y_set=X_test,y_test
X1,X2= np.meshgrid(np.arange(start=X_set[:,0].min()-
1,stop=X_set[:,0].max()+1,step=0.01),np.arange(start=X_set[:,1].min()-
1,stop=X_set[:,1].max()+1,step=0.01))
plt.contourf(X1,X2,classifier.predict(np.array([X1.ravel(),X2.ravel()]).T).reshap
e(X1.shape),alpha=0.75,cmap =ListedColormap(('black','white')))
plt.xlim(X1.min(),X1.max())
plt.ylim(X2.min(),X2.max())
for i,j in enumerate(np.unique(y_set)):

```

```

plt.scatter(X_set[y_set==j,0],X_set[y_set==j,1],c=L ListedColormap(('grey','black'))(i),label=j)
plt.title('Ensemble')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend()
plt.show()

```

b) Logistic Regression

```

classifier.fit(X_train,y_train)
#
Y_pred = classifier.predict(X_test)
#
from matplotlib.colors import ListedColormap
X_set,y_set=X_test,y_test
X1,X2= np.meshgrid(np.arange(start=X_set[:,0].min()-1,stop=X_set[:,0].max()+1,step=0.01),np.arange(start=X_set[:,1].min()-1,stop=X_set[:,1].max()+1,step=0.01))
plt.contourf(X1,X2,classifier.predict(np.array([X1.ravel(),X2.ravel()]).T).reshape(X1.shape),alpha=0.75,cmap =ListedColormap(('black','white')))
plt.xlim(X1.min(),X1.max())
plt.ylim(X2.min(),X2.max())
for i,j in enumerate(np.unique(y_set)):

plt.scatter(X_set[y_set==j,0],X_set[y_set==j,1],c=L ListedColormap(('grey','black'))(i),label=j)
plt.title('Logical Regression')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend()
plt.show()

```

c) Decision tree

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
#
dataset =pd.read_csv('cancer patient data sets.csv')
X=dataset.iloc[:,2:6].values
y=dataset.iloc[:,6].values
#
from sklearn.preprocessing import LabelEncoder
y= LabelEncoder().fit_transform(y)
#
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test
=train_test_split(X,y,test_size=0.5,random_state=0)
#
from sklearn.preprocessing import StandardScaler
sc_X= StandardScaler()
X_train=sc_X.fit_transform(X_train)
X_test=sc_X.transform(X_test)
#
from sklearn.tree import DecisionTreeClassifier
classifier = DecisionTreeClassifier(criterion ='entropy',random_state=0)
classifier.fit(X_train,y_train)
#
Y_pred = classifier.predict(X_test)
#
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test,Y_pred)
#
print("Accuracy of the Model is :
",(cm[0][0]+cm[1][1])*100/(cm[0][0]+cm[1][1]+cm[0][1]+cm[1][0]))
```

```

print("Precision of the Model is : ",(cm[0][0])*100/(cm[0][0]+cm[1][0]))
print("Recall of the Model is : ",(cm[0][0])*100/(cm[0][0]+cm[0][1]))
#
from sklearn.decomposition import PCA
pca= PCA(n_components=2)
X_train = pca.fit_transform(X_train)
X_test = pca.transform(X_test)
explained_variance= pca.explained_variance_ratio_
#
from sklearn.tree import DecisionTreeClassifier
classifier = DecisionTreeClassifier(criterion = 'entropy',random_state=0)
classifier.fit(X_train,y_train)
#
Y_pred = classifier.predict(X_test)
#
from matplotlib.colors import ListedColormap
X_set,y_set=X_test,y_test
X1,X2= np.meshgrid(np.arange(start=X_set[:,0].min()-
1,stop=X_set[:,0].max()+1,step=0.01),np.arange(start=X_set[:,1].min()-
1,stop=X_set[:,1].max()+1,step=0.01))
plt.contourf(X1,X2,classifier.predict(np.array([X1.ravel(),X2.ravel()]).T).reshape
(X1.shape),alpha=0.75,cmap =ListedColormap(('black','white')))
plt.xlim(X1.min(),X1.max())
plt.ylim(X2.min(),X2.max())
for i,j in enumerate(np.unique(y_set)):

plt.scatter(X_set[y_set==j,0],X_set[y_set==j,1],c=ListedColormap(('grey','black
'))(i),label=j)
plt.title('Decision Tree')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend()

```

```
plt.show()
```

d) KNN

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
import pandas as pd
```

```
#
```

```
dataset =pd.read_csv('lung_cancer_examples.csv')
```

```
X=dataset.iloc[:,2:6].values
```

```
y=dataset.iloc[:,6].values
```

```
#
```

```
from sklearn.preprocessing import LabelEncoder
```

```
y= LabelEncoder().fit_transform(y)
```

```
#
```

```
from sklearn.model_selection import train_test_split
```

```
X_train, X_test, y_train, y_test
```

```
=train_test_split(X,y,test_size=0.5,random_state=0)
```

```
#
```

```
from sklearn.preprocessing import StandardScaler
```

```
sc_X= StandardScaler()
```

```
X_train=sc_X.fit_transform(X_train)
```

```
X_test=sc_X.transform(X_test)
```

```
#
```

```
from sklearn.neighbors import KNeighborsClassifier
```

```
classifier = KNeighborsClassifier(n_neighbors=3,metric='minkowski',p=2)
```

```
classifier.fit(X_train,y_train)
```

```
#
```

```
Y_pred = classifier.predict(X_test)
```

```
#
```

```
from sklearn.metrics import confusion_matrix
```

```
cm = confusion_matrix(y_test,Y_pred)
```

```
#
```

```

print("Accuracy of the Model is :
",(cm[0][0]+cm[1][1])*100/(cm[0][0]+cm[1][1]+cm[0][1]+cm[1][0]))
print("Precision of the Model is : ",(cm[0][0])*100/(cm[0][0]+cm[1][0]))
print("Recall of the Model is : ",(cm[0][0])*100/(cm[0][0]+cm[0][1]))
#
from sklearn.decomposition import PCA
pca= PCA(n_components=2)
X_train = pca.fit_transform(X_train)
X_test = pca.transform(X_test)
explained_variance= pca.explained_variance_ratio_
#
from sklearn.neighbors import KNeighborsClassifier
classifier = KNeighborsClassifier(n_neighbors=3,metric='minkowski',p=2)
classifier.fit(X_train,y_train)
#
Y_pred = classifier.predict(X_test)
#
from matplotlib.colors import ListedColormap
X_set,y_set=X_test,y_test
X1,X2= np.meshgrid(np.arange(start=X_set[:,0].min()-
1,stop=X_set[:,0].max()+1,step=0.01),np.arange(start=X_set[:,1].min()-
1,stop=X_set[:,1].max()+1,step=0.01))
plt.contourf(X1,X2,classifier.predict(np.array([X1.ravel(),X2.ravel()]).T).reshape
(X1.shape),alpha=0.75,cmap =ListedColormap(('black','white')))
plt.xlim(X1.min(),X1.max())
plt.ylim(X2.min(),X2.max())
for i,j in enumerate(np.unique(y_set)):

plt.scatter(X_set[y_set==j,0],X_set[y_set==j,1],c=ListedColormap(('grey','black
'))(i),label=j)
plt.title('KNN ')
plt.xlabel('X-axis')

```

```
plt.ylabel('Y-axis')
```

```
plt.legend()
```

```
plt.show()
```

e) Shallow neural network

```
import csv
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
def normalize(X):
```

```
    X_norm = X
```

```
    [r,c]=np.shape(X)
```

```
    mu = np.zeros([c,1])
```

```
    sigma = np.zeros([c,1])
```

```
    for i in range(c):
```

```
        mu[i] = np.mean(X[:,i])
```

```
        sigma[i] = np.std(X[:,i])
```

```
        X_norm[:,i] =(X[:,i]-mu[i])/sigma[i];
```

```
    return [X_norm,mu,sigma]
```

```
def sigmoid(z):
```

```
    return (1/(1+np.exp(-z)))
```

```
a=np.zeros((259,1))
```

```
b=np.zeros((259,1))
```

```
c=np.zeros((259,1))
```

```
d=np.zeros((259,1))
```

```
e=np.zeros((259,1))
```

```
f=np.zeros((259,1))
```

```
g=np.zeros((259,1))
```

```
h=np.zeros((259,1))
```

```
p=np.zeros((259,1))
```

```
j=np.zeros((259,1))
```

```
k=np.zeros((259,1))
```

```
l=np.zeros((259,1))
```

```
m=np.zeros((259,1))
```

```

n=np.zeros((259,1))
o=np.zeros((259,1))
Y=np.zeros((259,1))
X1=np.zeros((259,15))
i=0
#lung_cancer_train.csv file includes 259 training examples
with open("lung_cancer_train.csv", "r") as csv_file:
    csv_reader = csv.DictReader(csv_file, delimiter=',')
    for lines in csv_reader:
        b[i]=lines['age']
        if(lines['gender']=='M'):
            a[i]=1
        else:
            a[i]=0
        if(lines['lung_cancer']=='true'):
            Y[i]=1
        else:
            Y[i]=0
        c[i]=lines['smoking']
        d[i]=lines['yellow_fingers']
        e[i]=lines['anxiety']
        f[i]=lines['peer_pressure']
        g[i]=lines['chronic_disease']
        h[i]=lines['fatigue']
        p[i]=lines['allergy']
        j[i]=lines['wheezing']
        k[i]=lines['alcohol_consuming']
        l[i]=lines['coughing']
        m[i]=lines['shortness_of_breath']
        n[i]=lines['swallowing_difficulty']
        o[i]=lines['chest_pain']
        i=i+1

```

```
for i in range(259):
```

```
    X1[i][0]=a[i]
```

```
    X1[i][1]=b[i]
```

```
    X1[i][2]=c[i]
```

```
    X1[i][3]=d[i]
```

```
    X1[i][4]=e[i]
```

```
    X1[i][5]=f[i]
```

```
    X1[i][6]=g[i]
```

```
    X1[i][7]=h[i]
```

```
    X1[i][8]=p[i]
```

```
    X1[i][9]=j[i]
```

```
    X1[i][10]=k[i]
```

```
    X1[i][11]=l[i]
```

```
    X1[i][12]=m[i]
```

```
    X1[i][13]=n[i]
```

```
    X1[i][14]=o[i]
```

```
[X,mu,sigma]=normalize(X1)
```

```
X=X.T
```

```
Y=Y.T
```

```
n_h=500 #hidden layer size
```

```
n_x=X.shape[0] #ip neuron size =15
```

```
n_y=Y.shape[0] #op neuron size =1
```

```
W1=np.random.randn(n_h,n_x)*0.01
```

```
b1=np.zeros(shape=(n_h,1))
```

```
W2=np.random.randn(n_y,n_h)*0.01
```

```
b2=np.zeros(shape=(n_y,1))
```

```
m=309
```

```
cost1=[]
```

```
for i in range(2000):
```

```
    Z1= np.dot(W1,X)+ b1
```

```
    A1=np.tanh(Z1) #activation function at ip side of hidden layer
```

```

Z2= np.dot(W2,A1) + b2
A2=sigmoid(Z2)#activation function at ip side of output layer
#A2=A2.reshape(-1,1)
#print(A2.shape)
logprobs = np.multiply(np.log(A2), Y) + np.multiply((1 - Y), np.log(1 -
A2))

cost = - np.sum(logprobs) / m
cost = float(np.squeeze(cost))
cost1.append(cost)
#cost= - (np.dot(np.log(A2),Y) + np.dot((1-Y),np.log(1-A2)))/m
#print(cost)
dZ2 = A2 - Y
dW2 = (1 / m) * np.dot(dZ2, A1.T)
db2 = (1 / m) * np.sum(dZ2, axis=1, keepdims=True)
dZ1 = np.multiply(np.dot(W2.T, dZ2), 1 - np.power(A1, 2))
dW1 = np.dot(dZ1, X.T)
db1 = (1 / m) * np.sum(dZ1, axis=1, keepdims=True)
lr=0.05
W1 = W1- lr*dW1
W2 = W2- lr*dW2
b1= b1- lr*db1
b2= b2- lr*db2

plt.plot(range(len(cost1)),cost1)
plt.xlabel('No. of iterations')
plt.ylabel('Cost function')
plt.title('Lung cancer classification using Neural network')
plt.show()
Ypredict=np.round(A2)
print('Training Accuracy: %d'%float((np.dot(Y,Ypredict.T)+ np.dot(1-Y,1-
Ypredict.T))/float(Y.size)*100)+ '%')
at=np.zeros((50,1))
bt=np.zeros((50,1))

```

```

ct=np.zeros((50,1))
dt=np.zeros((50,1))
et=np.zeros((50,1))
ft=np.zeros((50,1))
gt=np.zeros((50,1))
ht=np.zeros((50,1))
pt=np.zeros((50,1))
jt=np.zeros((50,1))
kt=np.zeros((50,1))
lt=np.zeros((50,1))
mt=np.zeros((50,1))
nt=np.zeros((50,1))
ot=np.zeros((50,1))
Yt=np.zeros((50,1))
X1t=np.zeros((50,15))
#lung_cancer_test.csv file includes 50 testing examples
i=0
with open("lung_cancer_test.csv", "r") as csv_file:
    csv_reader = csv.DictReader(csv_file, delimiter=',')
    for lines in csv_reader:
        bt[i]=lines['age']
        if(lines['gender']=='M'):
            at[i]=1
        else:
            at[i]=0
        if(lines['lung_cancer']=='true'):
            Yt[i]=1
        else:
            Yt[i]=0
        ct[i]=lines['smoking']
        dt[i]=lines['yellow_fingers']
        et[i]=lines['anxiety']

```

```

    ft[i]=lines['peer_pressure']
    gt[i]=lines['chronic_disease']
    ht[i]=lines['fatigue']
    pt[i]=lines['allergy']
    jt[i]=lines['wheezing']
    kt[i]=lines['alcohol_consuming']
    lt[i]=lines['coughing']
    mt[i]=lines['shortness_of_breath']
    nt[i]=lines['swallowing_difficulty']
    ot[i]=lines['chest_pain']
    i=i+1
for i in range(50):
    X1t[i][0]=at[i]
    X1t[i][1]=bt[i]
    X1t[i][2]=ct[i]
    X1t[i][3]=dt[i]
    X1t[i][4]=et[i]
    X1t[i][5]=ft[i]
    X1t[i][6]=gt[i]
    X1t[i][7]=ht[i]
    X1t[i][8]=pt[i]
    X1t[i][9]=jt[i]
    X1t[i][10]=kt[i]
    X1t[i][11]=lt[i]
    X1t[i][12]=mt[i]
    X1t[i][13]=nt[i]
    X1t[i][14]=ot[i]
[Xtest,mu,sigma]=normalize(X1t)
Xtest=Xtest.T
Ytest=Yt.T
Z1test=np.dot(W1,Xtest)+ b1
Atest=np.tanh(Z1test)

```

```

Z2test=np.dot(W2,Atest) + b2
A2test=sigmoid(Z2test)
Ytest_predict= np.round(A2test)
print('Testing Accuracy: %d'%float((np.dot(Ytest,Ytest_predict.T)+ np.dot(1-
Ytest,1-Ytest_predict.T))/float(Ytest.size)*100)+ '%')

```

f) Support Vector Machine

```

import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
#
dataset =pd.read_csv('lung_cancer_examples.csv')
X=dataset.iloc[:,2:6].values
y=dataset.iloc[:,6].values
#
from sklearn.preprocessing import LabelEncoder
y= LabelEncoder().fit_transform(y)
#
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test
=train_test_split(X,y,test_size=0.5,random_state=0)
#
from sklearn.preprocessing import StandardScaler
sc_X= StandardScaler()
X_train=sc_X.fit_transform(X_train)
X_test=sc_X.transform(X_test)
#
from sklearn.svm import SVC
classifier = SVC(kernel ='poly', random_state=0)
classifier.fit(X_train,y_train)
#
Y_pred = classifier.predict(X_test)
#

```

```

from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test,Y_pred)
#
print("Accuracy of the Model is :
",(cm[0][0]+cm[1][1])*100/(cm[0][0]+cm[1][1]+cm[0][1]+cm[1][0]))
print("Precision of the Model is : ",(cm[0][0])*100/(cm[0][0]+cm[1][0]))
print("Recall of the Model is : ",(cm[0][0])*100/(cm[0][0]+cm[0][1]))
#
from sklearn.decomposition import PCA
pca= PCA(n_components=2)
X_train = pca.fit_transform(X_train)
X_test = pca.transform(X_test)
explained_variance= pca.explained_variance_ratio_
#
from sklearn.svm import SVC
classifier = SVC(kernel ='poly', random_state=0)
classifier.fit(X_train,y_train)
#
Y_pred = classifier.predict(X_test)
#
from matplotlib.colors import ListedColormap
X_set,y_set=X_test,y_test
X1,X2= np.meshgrid(np.arange(start=X_set[:,0].min()-
1,stop=X_set[:,0].max()+1,step=0.01),np.arange(start=X_set[:,1].min()-
1,stop=X_set[:,1].max()+1,step=0.01))
plt.contourf(X1,X2,classifier.predict(np.array([X1.ravel(),X2.ravel()]).T).reshape
(X1.shape),alpha=0.75,cmap =ListedColormap(('black','white')))
plt.xlim(X1.min(),X1.max())
plt.ylim(X2.min(),X2.max())
for i,j in enumerate(np.unique(y_set)):

```

```

plt.scatter(X_set[y_set==j,0],X_set[y_set==j,1],c=ListedColormap(('grey','black'))(i),label=j)
plt.title('SVM')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend()
plt.show()

```

g) Custom ensemble

```

import pandas as pd
#
dataset =pd.read_csv('lung_cancer_examples.csv')
X=dataset.iloc[:,2:6].values
y=dataset.iloc[:,6].values
#
from sklearn.preprocessing import LabelEncoder
y= LabelEncoder().fit_transform(y)
#
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test
=train_test_split(X,y,test_size=0.5,random_state=0)
#
from sklearn.preprocessing import StandardScaler
sc_X= StandardScaler()
X_train=sc_X.fit_transform(X_train)
X_test=sc_X.transform(X_test)
#
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators=10,
criterion='entropy',random_state=0)
classifier.fit(X_train,y_train)
#

```

```

Y_pred = classifier.predict(X_test)
#
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test,Y_pred)
#
print("Accuracy of the Model is :
", (cm[0][0]+cm[1][1])*100/(cm[0][0]+cm[1][1]+cm[0][1]+cm[1][0]))
print("Precision of the Model is : ", (cm[0][0])*100/(cm[0][0]+cm[1][0]))
print("Recall of the Model is : ", (cm[0][0])*100/(cm[0][0]+cm[0][1]))
#
from sklearn.decomposition import PCA
pca= PCA(n_components=2)
X_train = pca.fit_transform(X_train)
X_test = pca.transform(X_test)
explained_variance= pca.explained_variance_ratio_
#
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators=10,
criterion='entropy',random_state=0)
classifier.fit(X_train,y_train)
#
Y_pred = classifier.predict(X_test)
#
from matplotlib.colors import ListedColormap
X_set,y_set=X_test,y_test
X1,X2= np.meshgrid(np.arange(start=X_set[:,0].min()-
1,stop=X_set[:,0].max()+1,step=0.01),np.arange(start=X_set[:,1].min()-
1,stop=X_set[:,1].max()+1,step=0.01))
plt.contourf(X1,X2,classifier.predict(np.array([X1.ravel(),X2.ravel()]).T).reshape
(X1.shape),alpha=0.75,cmap =ListedColormap(('black','white')))
plt.xlim(X1.min(),X1.max())
plt.ylim(X2.min(),X2.max())

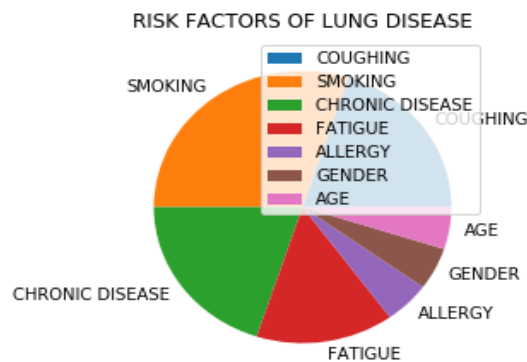
```

```
for i,j in enumerate(np.unique(y_set)):
plt.scatter(X_set[y_set==j,0],X_set[y_set==j,1],c=ListedColormap(('grey','black
'))(i),label=j)
plt.title('Ensemble')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend()
plt.show()
```

2 OUTPUT SCREENSHOTS

	Patient Id	Age	...	Fatigue	Weight Loss
Patient Id	1.000000	0.002674	...	0.042346	0.026393
Age	0.002674	1.000000	...	0.095059	0.106946
Gender	-0.025739	-0.202086	...	-0.116467	-0.057993
Air Pollution	0.053307	0.099494	...	0.211724	0.258016
Alcohol use	0.041374	0.151742	...	0.237245	0.207851
Dust Allergy	0.037960	0.035202	...	0.332472	0.321756
OccuPational Hazards	0.032355	0.062177	...	0.267844	0.176226
Genetic Risk	0.030725	0.073151	...	0.230530	0.271743
chronic Lung Disease	0.025177	0.128952	...	0.247697	0.104080
Balanced Diet	0.030743	0.004863	...	0.400678	-0.006544
Obesity	0.050584	0.034337	...	0.552788	0.313495
Smoking	0.018407	0.075333	...	0.200029	-0.212937
Passive Smoker	0.019517	0.004908	...	0.377919	0.058336
Chest Pain	0.022210	0.012864	...	0.251135	-0.001092
Coughing of Blood	0.049401	0.053006	...	0.481540	0.105857
Fatigue	0.042346	0.095059	...	1.000000	0.469517
Weight Loss	0.026393	0.106946	...	0.469517	1.000000

a) correlation in data



b) Pie chart of risk factors

count	309.000000	count	309.000000
mean	62.673139	mean	1.563107
std	8.210301	std	0.496806
min	21.000000	min	1.000000
25%	57.000000	25%	1.000000
50%	62.000000	50%	2.000000
75%	69.000000	75%	2.000000
max	87.000000	max	2.000000

Name: AGE, dtype: float64 Name: SMOKING, dtype: float64

c) Mean and standard deviation of age and smoking category

Accuracy of the Model is : 96.66666666666667
Precision of the Model is : 93.33333333333333
Recall of the Model is : 100.0

d) Logistic regression accuracy and precision