Prediction of Lung Cancer Using Supervised Machine Learning

Gourab Mukhopadhaya [1], Suman Sett [2], Sumit Basu [3], Susmit Chakraborty [4], Swarnendu Shil [5], Arijit Sen [6], Priti Munyan [7]

[1,2,3,5,6,7]Student, Department of Computer Science & Engineering (CS&DS), Brainware University
[4]Assistant Professor, Department of Computer Science and Engineering (CS&DS), Brainware University

Abstract

Lung cancer remains one of the leading causes of cancer-related mortality worldwide, necessitating early detection for improved survival rates. This study focuses on analysing various detection techniques for lung cancer, integrating supervised machine learning and logistic regression-based approaches. Lung cancer cannot be avoided but it can be controlled. It is better to predict lung cancer at stages I & II, so that the chances of getting come round is high. This model offers a hand-held grief solution by enabling early and cost-effective detection of lung cancer. This study encloses us a model that can predict the risk of lung cancer affection by working out on the possibilities. Here all insights are invented using Supervised Machine learning logistic regression to be more exact. It focuses on creating models that can be trained from extensive datasets. The key features of this approach are gender, age, smoking, yellow fingers, anxiety, peer pressure, chronic disease, fatigue, allergy, wheezing, alcohol consumption, coughing, shortness of breath, swallowing difficulty and chest pain. The whole system is trained in Jupyter notebook. It is evaluated across diverse datasets and it demonstrates robustness, interpretability, and potentiality for clinical integration.

Keywords: Cancer prediction, Machine learning model, Logistic regression, Confusion matrix, NumPy, Pandas, Jupyter notebook

1. Introduction

Lung cancer (LC) is the leading cause of cancer-related deaths worldwide, mostly because it is frequently asymptomatic and detected in advanced stages when metastases to other important organs have taken place previously [1]. With a 5-year survival rate that varies from 5% to 57% depending on the type and timing of cancer detection, and a mortality rate of 1.76 million out of 2.09 million recorded cases, prompt diagnosis and early treatment are crucial [2]. According to the World Health Organization, the mortality rate from this severe sickness is 84.21%, which puts it close to 11.4% of India's growing risks [3]. Traditional methods of diagnosing cancer, such as visual inspection and biopsy, can be laborious, arbitrary, and prone to errors [4]. To find cancer early on, researchers use screening, identification, and classification techniques [5]. It is difficult to detect and diagnose lung cancer in a timely manner since early-stage lung cancer frequently exhibits minor clinical symptoms and only 70% of patients have Lung Cancer discovered at an advanced stage [2]. Early disease prediction based on textual information and symptoms could improve the diagnostic process. In addition to medical approaches, an expert may identify the disease early by using soft computing techniques such as applying machine learning algorithms to the key characteristics of huge, complex lung cancer datasets [6]. In recent years, the usage of gadgets to benefit the understanding of (ML) algorithms in healthcare has received plenty of attention due to its capacity to decorate diagnosis and treatment consequences [7]. In order to predict the recurrence of lung cancer and its survivability, a number of researchers proposed machine learning algorithms on clinical datasets, including neural networks, support vector machines and decision trees, convolutional neural networks based non-linear cellular automata, Random Forest, XGBoost, and logistic regression [6]. Zhang et al. employed Risk prediction of lung cancer using ML. The purpose of this study was to create machine learning (ML) models and explain the relationship between risk factors and lung cancer by incorporating these variables based on UK Biobank (UKB) data into a new, wider range of variables [8]. Jamil Ahmmed et al. conducted recent research. In order to predict lung cancer, this study assesses the effectiveness of five machine learning algorithms: XGBoost, LightGBM, AdaBoost, Logistic Regression, and support Vector Machines (SVM) [9]. Hossain et al. researched the efficacy of machine learning models in lung cancer risk prediction with explainability. Here, this study examines the accuracy and performance of several machine learning models using a numerical dataset of factors related to lung cancer [10]. This study explores early detection techniques for lung cancer using supervised machine learning and logistic regression-based approaches. It aims to predict lung cancer risk at stages I and II, reducing the chances of recurrence. The model uses features like gender, age, smoking, anxiety, peer pressure, chronic disease, fatigue, allergies, and chest pain. The system is trained on Jupiter Notebook and evaluated across diverse datasets, demonstrating robustness and potential for clinical integration. The results indicate that the model can effectively identify individuals at high risk for lung cancer, enabling timely interventions and personalized treatment plans. Future research will focus on refining the algorithm and expanding the dataset to enhance accuracy and generalizability across different populations.

2. System Model

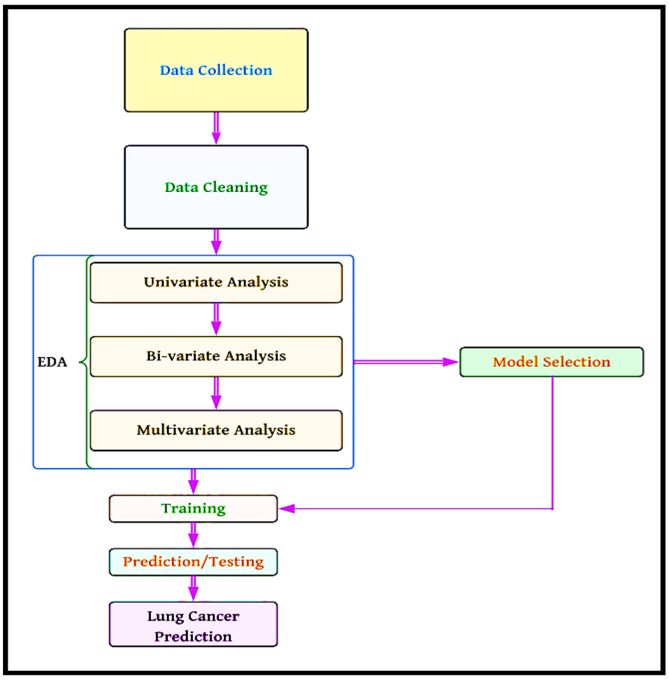


Figure 1: Schematic diagram of the proposed method

Figure. 1 displays the diagram of the system architecture of the proposed system to predict lung cancer based on a structured machine learning pipeline. It starts with the data collection, in which all the meaningful details about the patient including age, history of smoking, the symptoms and the clinical signs are collected. Finally, we proceed with data cleaning which involves dealing with missing values, correcting anomalies and tidying up the data set for analysis. The pipeline is mostly driven by Exploratory Data Analysis (EDA) which can be divided into three steps: Univariate Analysis which examines the distribution of individual features; Bi-variate Analysis that highlights associations between two features; and Multivariate Analysis that uncovers deeper insights on interactions of more than two features. Model Selection (modeling), this is how we will choose the rightest algorithm like logistic regression. After selection, the model is trained to recognize patterns of lung cancer on the cleaned dataset. In the Prediction/Testing phase, the trained model is validated on unseen data to assess its performance using metrics such as Accuracy, Precision, Recall, F1 score, and AUC. The final outcome is Lung Cancer Prediction, which provides critical insights that can support early diagnosis and clinical decision-making. The process is iterative, allowing continuous refinement of both model and data strategy to achieve optimal results.

3. Logistic Regression

Logistic regression is a fundamental classification algorithm commonly utilized in predictive modelling tasks involving binary response variables [12]. Logistic regression models are the likelihood of a categorical result, usually denoted by 0 or 1, as opposed to linear regression, which is used to predict continuous outcomes. Based on a collection of physiological, behavioural, and environmental factors, logistic regression is a strong statistical method for predicting the likelihood of lung cancer, classifying outcomes as either healthy or having cancer**.** Logistic regression functions mathematically by expressing the log odds of the dependent variable as a linear function of the input variables. Given an input vector, the logistic regression model computes a linear combination where $_0$ is the intercept term and β_i is the coefficient associated with each predictor variable. The sigmoid function is then used to alter this linear combination, and it is defined as the eq. (1).

$$P(Y) = \frac{1}{1+e^{-z}} = \frac{1}{e^{-(\beta_0+\beta_1 lc_1+\beta_2 lc_2+\cdots+\beta_n lc_n)}} \tag{1}$$

The sigmoid function is a probabilistic approach used in health-related modelling to convert real-valued inputs into intervals (0,1). It shrinks input values, turning high positive values into 1 and negative values into 0. The output is the probability of a positive class, which is crucial for uncertainty and risk assessment. The sigmoid function's graphic notation is an S-shaped curve, and the cut-off threshold is the decision factor that classifies results into classes or categories [13].

$$Y = \sigma(Z) = \frac{1}{1+e^{-Z}} \tag{2}$$

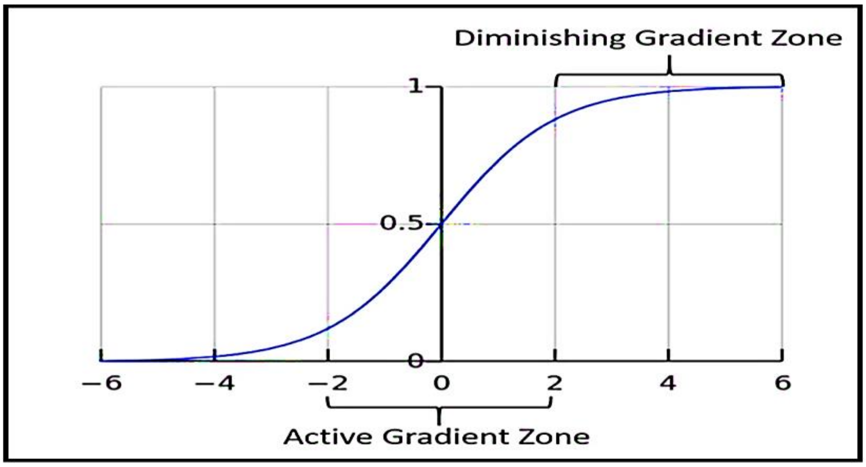


Figure 2: Graphical view of sigmoid function

The coefficients βᵢ measure the impact of each predictor on outcome probabilities, with positive coefficients increasing likelihood and negative coefficients decreasing it.

$$L(\beta) = \prod_{i=0}^{n}\left[h_\beta\left(a^{(i)}\right)\right]^{b^{(i)}}\left[1 - h_\beta\left(a^{(i)}\right)\right]^{1-b^{(i)}} \qquad (3)$$

Where $h_\beta\left(a^{(i)}\right) = \frac{1}{\left(1+e^{-\left(\beta^T a^{(i)}\right)}\right)}$ is the predicted probability for the $i^{\,th}$sample. To simplify computation, the logarithm of the likelihood function is used, resulting in the log-likelihood illustrated in eq. (4).

$$l(\beta) = \sum_{i=0}^{n}\left[b^{(i)}\, log\, log\, \left(h_\beta\left(a^{(i)}\right)\right) + \left(1 - b^{(i)}\right) log\, log\, \left(1 - h_\beta\left(a^{(i)}\right)\right)\ \right] \qquad (4)$$

By minimizing the negative log-likelihood, which acts as the model's cost function, the parameters are optimized as it describes in the eq. (5).

$$J(\beta) = -\left(\frac{1}{m}\right)\sum_{i=0}^{n}\left[b^{i}\, log\, log\, \left(h_\beta\left(a^{(i)}\right)\right) + \left(1 - b^{(i)}\right) log\, log\, \left(1 - h_\beta\left(a^{(i)}\right)\right)\right] \qquad (5)$$

The convex cost function suggests a global minimum achievable through iterative optimization techniques like gradient descent with an update rule for each coefficient during training eq. (6).

$$\beta_j = \beta_j - \eta\frac{\partial J(\beta)}{\partial \beta_j} \qquad (6)$$

η represents the learning rate, and $\frac{\partial j(\beta)}{\partial \beta_j}$ denotes the partial derivative of the cost function with respect to ωⱼ. The gradient for each parameter is calculated as it describes in eq. (7).

$$\frac{\partial j(\beta)}{\partial \beta_j} = \left(\frac{1}{m}\right)\sum_{i=0}^{n}\left(h_\beta\left(a^{(i)}\right) - b^{(i)}\right)a_j^{(i)} \qquad (7)$$

The final model produces a probability score for each class, which is then converted into binary predictions using a classification threshold of 0.5. However, this threshold can be adjusted to meet the specificity as required.

Confusion matrix parameters like precision, recall, F-1 score are considered here for model evaluation. The required matrices are discussed in the later part of this section. A confusion matrix is a visual representation of classifier algorithms' output, used for performance evaluation in multi-class classification jobs [14]. It measures classification overlap and evaluates a model's performance by counting true positives, true negatives, false positives, and false negatives [15]. The matrix shows the classes the results should have been for, while projections are displayed in columns. The F-score is a key metric used to determine which predictions are incorrect [16]. Sensitivity measures the model's ability to predict favourable results, examining both false negatives and real positives. This helps in determining which predictions are incorrect.

$$Recall = \frac{TP}{TP+FN} \tag{8}$$

Specificity quantifies the model's ability to predict unfavourable outcomes. It is comparable to sensitivity, except it is seen from the standpoint of unfavourable outcomes.

$$Specificity = \frac{TN}{TN+FP} \tag{9}$$

The model's accuracy indicates how frequently it is accurate. This means the proportion of observations that were accurately predicted to all observations.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{10}$$

The precision can be defined as the proportion of accurately anticipated positive observations to all predicted positive observations.

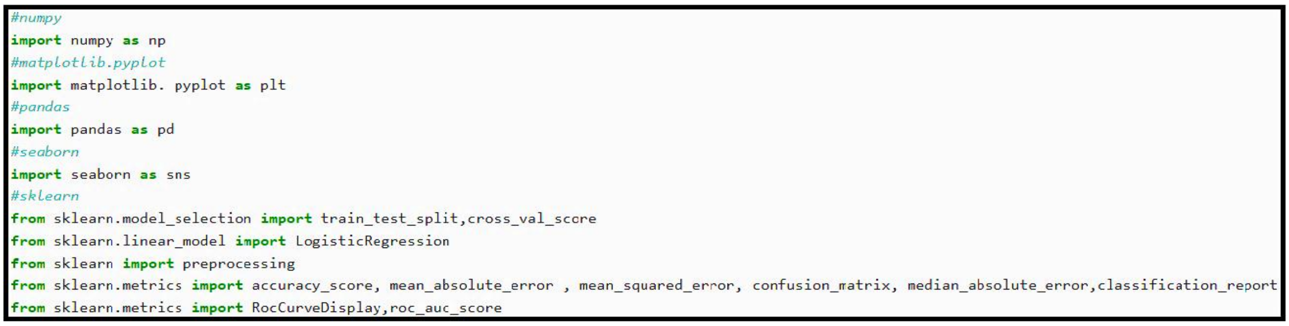
$$Precision = \frac{TP}{TP+FP} \tag{11}$$

The F-score (F1-score) is a measure of accuracy and sensitivity that is effective in unbalanced datasets, considering both false positive and false negative situations.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision+Recall} \tag{12}$$

4. Result Analysis

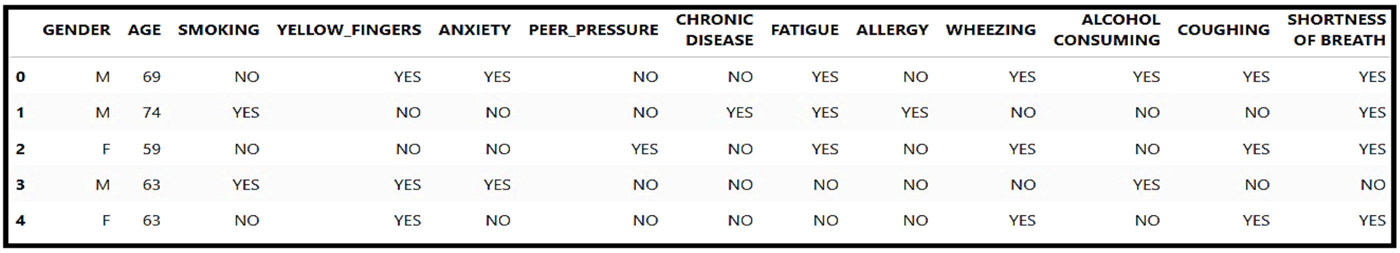
The authors used Python to implement the simulation work on the Jupyter Notebook platform. Using the Python programs shown in Figure. 3, a few libraries are first added, including pandas, numpy, matplotlib, and seaborn.



```
#numpy
import numpy as np
#matplotlib.pyplot
import matplotlib. pyplot as plt
#pandas
import pandas as pd
#seaborn
import seaborn as sns
#sklearn
from sklearn.model_selection import train_test_split,cross_val_score
from sklearn.linear_model import LogisticRegression
from sklearn import preprocessing
from sklearn.metrics import accuracy_score, mean_absolute_error , mean_squared_error, confusion_matrix, median_absolute_error,classification_report
from sklearn.metrics import RocCurveDisplay,roc_auc_score
```

Figure 3: Snapshot of the imported libraries

Numpy and pandas are python libraries used for numerical operations and data analysis. They are also used for checking outliers and randomness present in the data. The authors use the head method to observe the first few samples of the dataset, as shown in Figure. 4.

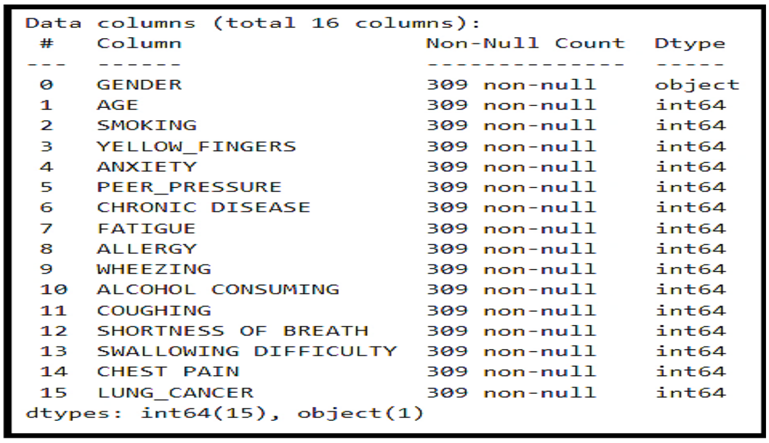


	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH
0	M	69	NO	YES	YES	NO	NO	YES	NO	YES	YES	YES	YES
1	M	74	YES	NO	NO	NO	YES	YES	YES	NO	NO	NO	YES
2	F	59	NO	NO	NO	YES	NO	YES	NO	YES	NO	YES	YES
3	M	63	YES	YES	YES	NO	NO	NO	NO	NO	YES	NO	NO
4	F	63	NO	YES	NO	NO	NO	NO	NO	YES	NO	YES	YES

Figure 4: Head view of the dataset

Null values are checked using the “lung_data.isnull().sum()” and no null value is found in the dataset. Data type information is obtained using the “info()” as illustrated in Figure. 5. There are four columns;

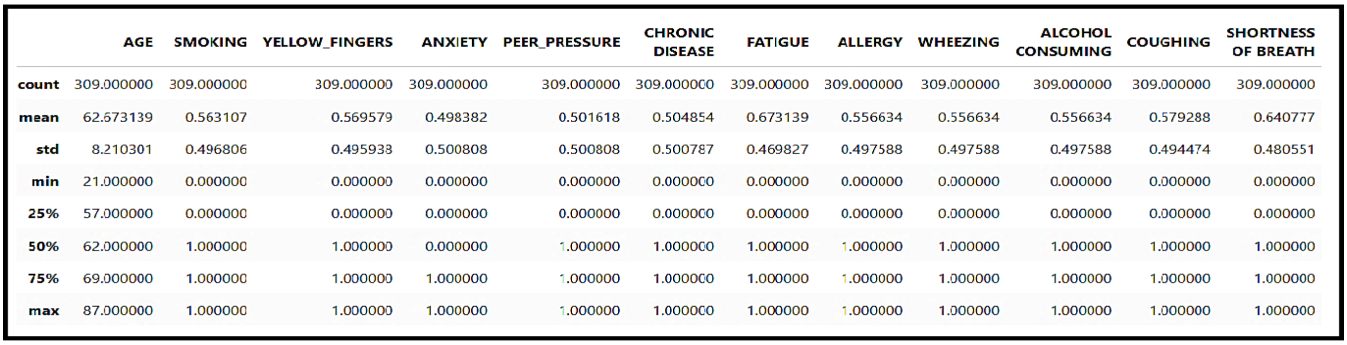
the first one indicates the serial number, the second one is for the name of the features, the non-null count is found in the third column and the last one identifies the datatype of different features considered here.



```
Data columns (total 16 columns):
 #   Column                 Non-Null Count  Dtype 
---  ------                 --------------  ----- 
 0   GENDER                 309 non-null    object
 1   AGE                    309 non-null    int64 
 2   SMOKING                309 non-null    int64 
 3   YELLOW_FINGERS         309 non-null    int64 
 4   ANXIETY                309 non-null    int64 
 5   PEER_PRESSURE          309 non-null    int64 
 6   CHRONIC DISEASE        309 non-null    int64 
 7   FATIGUE                309 non-null    int64 
 8   ALLERGY                309 non-null    int64 
 9   WHEEZING               309 non-null    int64 
 10  ALCOHOL CONSUMING      309 non-null    int64 
 11  COUGHING               309 non-null    int64 
 12  SHORTNESS OF BREATH    309 non-null    int64 
 13  SWALLOWING DIFFICULTY  309 non-null    int64 
 14  CHEST PAIN             309 non-null    int64 
 15  LUNG_CANCER            309 non-null    int64 
dtypes: int64(15), object(1)
```

Figure 5: Datatype information of the feature variables

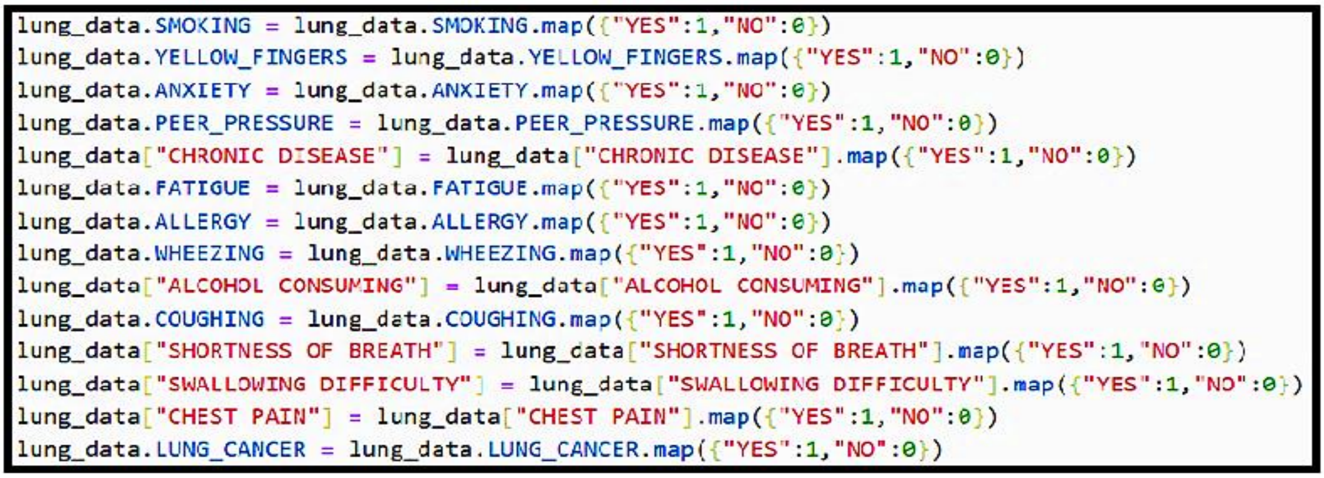
The method "describe()" results the statistical outcomes of the features such as mean, standard deviation, minimum, maximum values along with 25-percentile, 50-percentile, 75-percentile values. The complete statistical results are illustrated in Figure.6.



	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH
count	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000
mean	62.673139	0.563107	0.569579	0.498382	0.501618	0.504854	0.673139	0.556634	0.556634	0.556634	0.579288	0.640777
std	8.210301	0.496806	0.495933	0.500808	0.500808	0.500787	0.469827	0.497588	0.497588	0.497588	0.494474	0.480551
min	21.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	57.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	62.000000	1.000000	1.000000	0.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
75%	69.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
max	87.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

Figure 6: Statistical observations of the feature variables

In order to make category data appropriate for mathematics or machine learning models, binary mapping is the act of transforming information into binary values, usually 0 and 1. Here, we transformed all of the data into a symmetric binary variable in Figure.7, denoted by the numbers 0 and 1.



```
lung_data.SMOKING = lung_data.SMOKING.map({"YES":1,"NO":0})
lung_data.YELLOW_FINGERS = lung_data.YELLOW_FINGERS.map({"YES":1,"NO":0})
lung_data.ANXIETY = lung_data.ANXIETY.map({"YES":1,"NO":0})
lung_data.PEER_PRESSURE = lung_data.PEER_PRESSURE.map({"YES":1,"NO":0})
lung_data["CHRONIC DISEASE"] = lung_data["CHRONIC DISEASE"].map({"YES":1,"NO":0})
lung_data.FATIGUE = lung_data.FATIGUE.map({"YES":1,"NO":0})
lung_data.ALLERGY = lung_data.ALLERGY.map({"YES":1,"NO":0})
lung_data.WHEEZING = lung_data.WHEEZING.map({"YES":1,"NO":0})
lung_data["ALCOHOL CONSUMING"] = lung_data["ALCOHOL CONSUMING"].map({"YES":1,"NO":0})
lung_data.COUGHING = lung_data.COUGHING.map({"YES":1,"NO":0})
lung_data["SHORTNESS OF BREATH"] = lung_data["SHORTNESS OF BREATH"].map({"YES":1,"NO":0})
lung_data["SWALLOWING DIFFICULTY"] = lung_data["SWALLOWING DIFFICULTY"].map({"YES":1,"NO":0})
lung_data["CHEST PAIN"] = lung_data["CHEST PAIN"].map({"YES":1,"NO":0})
lung_data.LUNG_CANCER = lung_data.LUNG_CANCER.map({"YES":1,"NO":0})
```

Figure 7: Snapshot of Binary Mapping

A count plot is a kind of bar chart that displays how many times each category appears in a categorical variable. Here, in our dataset we denote 0 as no lung cancer is found and also 1 is denoted for whos' have lung cancer. In Figure. 8.1(a) illustrates the correlation between the diagnosis of lung cancer (LUNG_CANCER: 0 = No, 1 = Yes) and smoking status (SMOKING: 0 = Non-smoker, 1 = Smoker). Lung cancer is the most common diagnosis for both smokers and non-smokers. There are significantly more lung cancer instances among smokers than among non-smokers, and smokers (1 on the x-axis) have a higher total count. In Figure. 8.1(b) the authors have identified that individuals with yellow fingers (1) have a much higher count of lung cancer cases compared to those without yellow fingers. In below Figure. 8.1(c), count plot displays the relationship between ANXIETY (0 = No, 1 = Yes) and LUNG_CANCER (0 = No, 1 = Yes). However, we have found that people with anxiety (1) seem to have a marginally higher incidence of lung cancer than people without anxiety. Lung cancer affects most of the people in Figure. 8.1(d) who were subjected to peer pressure (PEER_PRESSURE = 1). Lung cancer is highly prevalent among people with chronic diseases (CHRONIC DISEASE = 1), and there are very few cases of lung cancer-free people, as shown in Figure. 8.1(e). However, Figure. 8.1(f) shows that people with fatigue have a slightly higher incidence of lung cancer. Most of the people in Figure. 8.2(g) who were subjected to allergy (ALLERGY = 1), have lung cancer. Wheezing-free people can also develop lung cancer, but to a much lesser degree, as shown in Figure. 8.2(h). Only a small fraction of non-cancer patients have wheezing, which could suggest that wheezing might be more specific to lung conditions, including cancer. Although it happens to a much lesser degree but Figure. 8.2(i) illustrates that lung cancer also develops in people who do not consume alcohol. The incidence of lung cancer is significantly higher in patients who cough (COUGHING =1), as shown in Figure 8.2(j). We have greatly observed in Figure. 8.2(k) that those individuals who have shortness of breath, suffer from lung cancer. In Figure. 8.2(l), chest pain portrays extremely high lung cancer incidence.

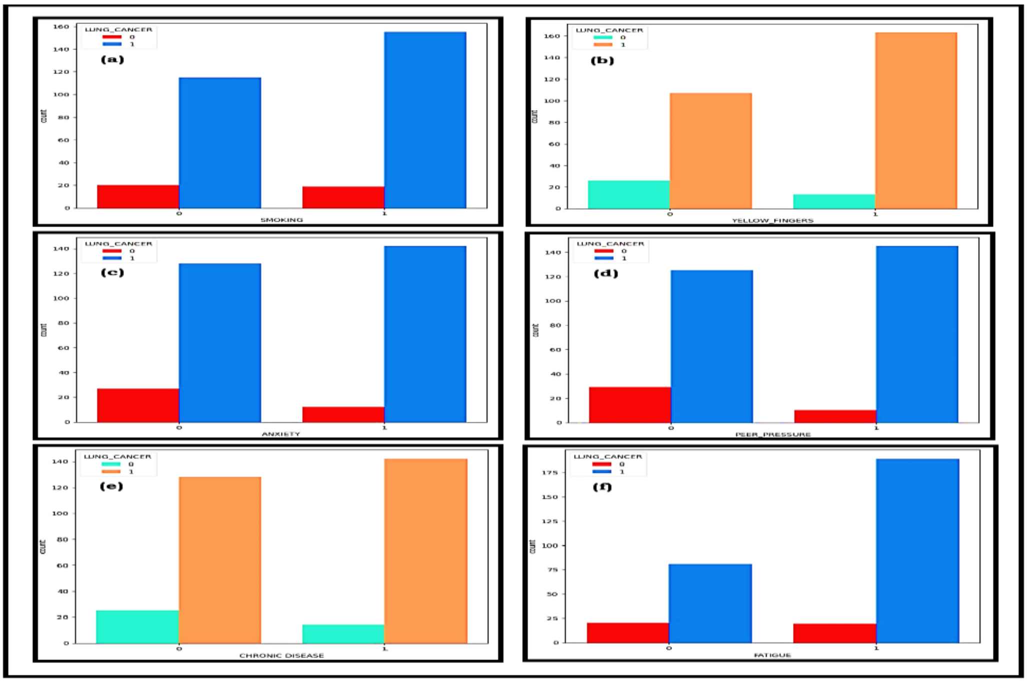


Figure 8.1: Snapshot of Countplot

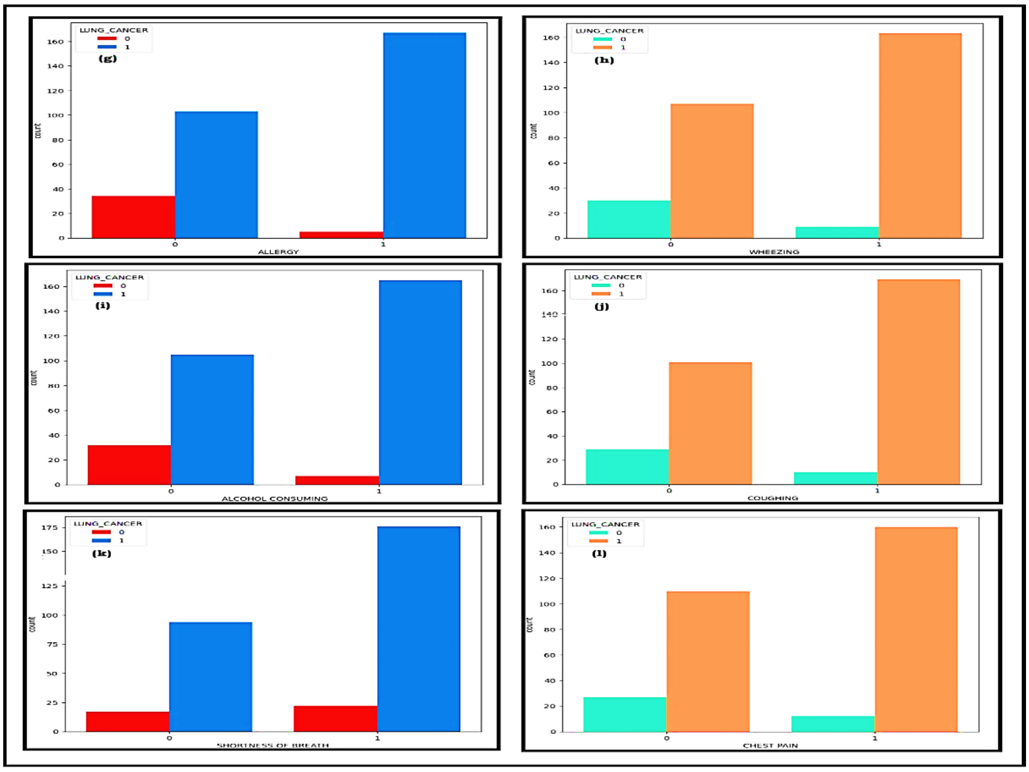


Figure 8.2: Snapshot of Countplot

A pie plot (or pie chart) is a circular statistical graphic used to represent data in a proportional way. It is divided into slices, each representing a particular category's contribution to the whole dataset. In Figure. 9.1(a), the gender distribution shows that males(52.8%) slightly outnumber females(47.2%). Figure. 9.1(b) illustrates smoking habits, where 56.4% of individuals are smokers and 43.6% are non-smokers. In Figure. 9.1(c), the presence of yellow-stained fingers is observed in 57.0% of the people, indicating a link with smoking. Figure. 9.1(d) displays anxiety levels, which are almost evenly split, with 50.2% showing signs of anxiety. Figure. 9.1(e) highlights peer pressure, affecting 50.5% of individuals. Figure. 9.1(f) deals with chronic disease, where 50.2% are affected, showing a nearly balanced distribution. Moving on, Figure. 9.2(g) shows that fatigue is present in 67.1% of the individuals, making it a major symptom. In Figure. 9.2(h), allergies affect 55.4% of people, making it a significant concern. Figure. 9.2(i) highlights wheezing, experienced by 55.7% of individuals. Figure. 9.2(j) shows that 56.0% of the population consumes alcohol. Figure. 9.2(k) reveals that coughing is a common symptom, affecting 58.0% of the individuals. Figure. 9.2(l) shows shortness of breath in 63.8% of the population, indicating a widespread issue. In Figure. 9.3(m), swallowing difficulty is reported by 47.2% of individuals, while Figure. 9.3(n) shows that chest pain is experienced by 55.7% of the participants, making it a common health problem.

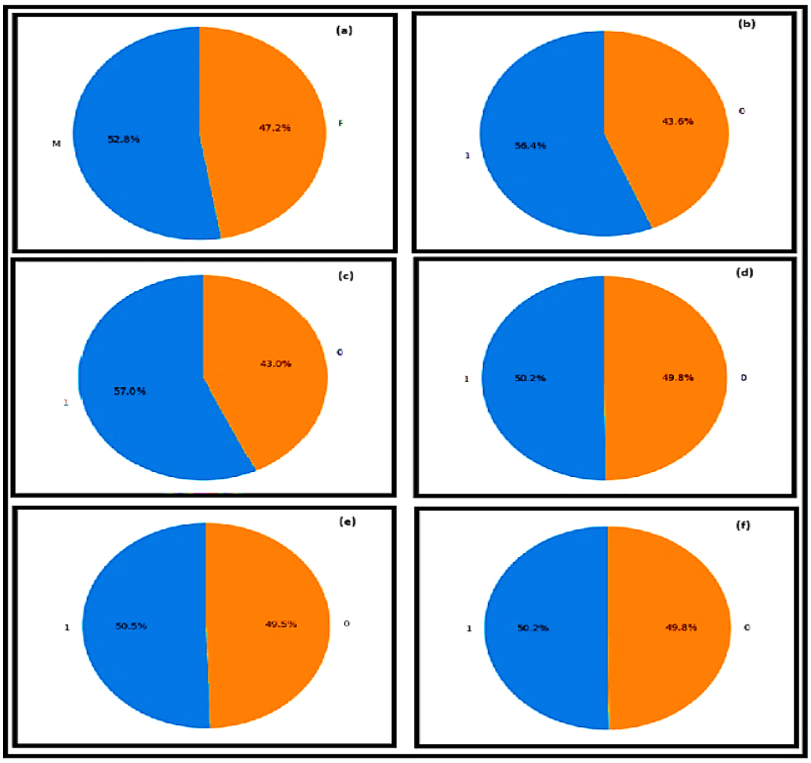


Figure 9.1: Snapshot of Pie plot

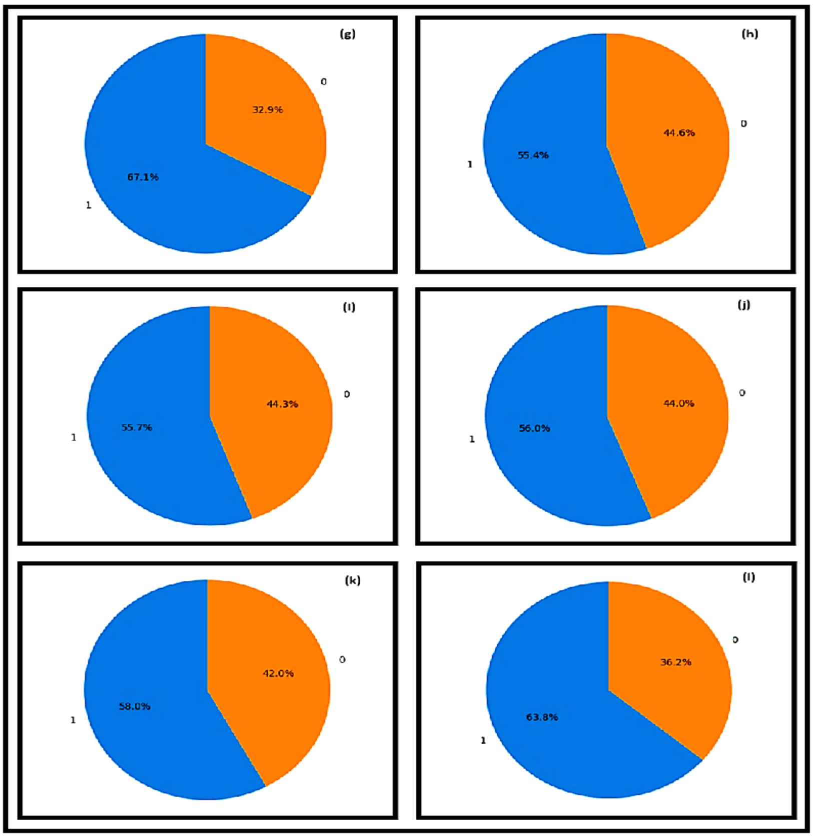


Figure 9.2: Snapshot of Pieplot

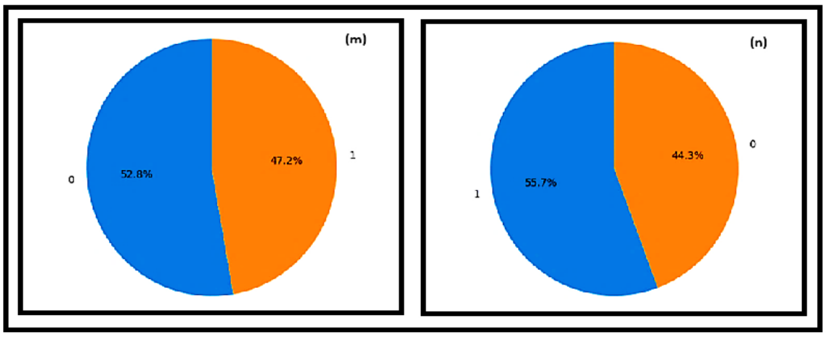


Figure 9.3: Snapshot of Pie plot

A boxplot is a graphical tool that makes it easier to see the variability, central tendency, and spread. Figure. 10 illustrates two box plots comparing age distributions in relation to lung cancer prediction. The age distribution of lung cancer patients is shown in Figure. 10(a), with a median age of approximately 63 and a wider range of ages with a few lower outliers around 20 to 30 years. There is a slight bias toward younger people. Figure. 10(b), provides a more symmetrical distribution with no noticeable outliers to illustrate the age distribution of individuals without lung cancer. These individuals have a median age of about 60 and a slightly smaller interquartile range than plot A.

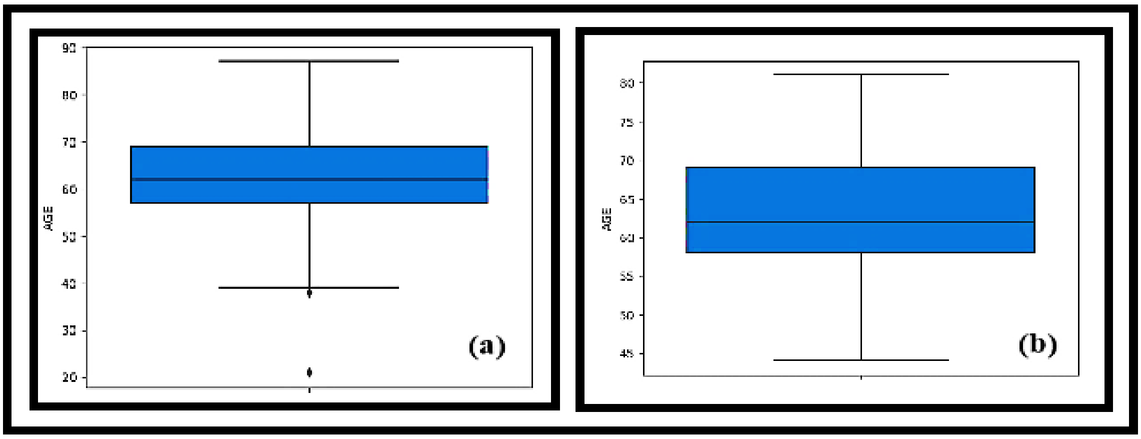


Figure 10: Snapshot of Boxplot

A histogram is a graphical representation that organizes a group of data points into user-specified ranges, showing the frequency distribution of a dataset. The histogram for lung cancer (Figure. 11) shows that most patients fall between the ages of 55 and 70, with the highest concentration around 60 years old, indicating lung cancer prevalence increases in older adults.

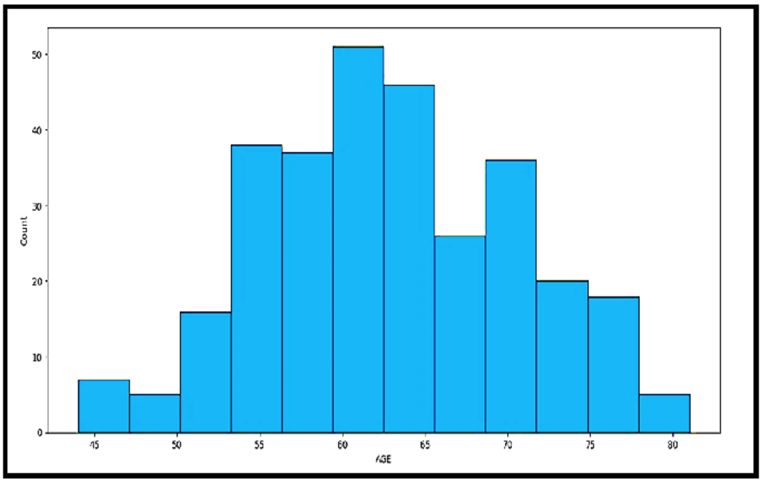


Figure 11: Snapshot of Histogram

A correlation heatmap is a visual aid that uses colour gradients to show the direction and intensity of correlations between several variables. The heatmap (Figure. 12) illustrates the relationship between several variables and the occurrence of lung cancer in the context of lung cancer prediction. The heatmap indicates a positive relationship between lung cancer and factors like alcohol consumption (0.28), chronic illness (0.28), and shortness of breath (0.25). Other variables that show weaker but still noticeable correlations are smoking (0.058) and yellow fingers (0.17). Interestingly, peer pressure (0.24) also seems to have a moderate correlation, indicating influences from lifestyle and behaviour. Overall, the analysis shows that although there isn't a very strong association between any one element alone, a mix of symptoms and behaviours greatly adds to Lung Cancer prediction.

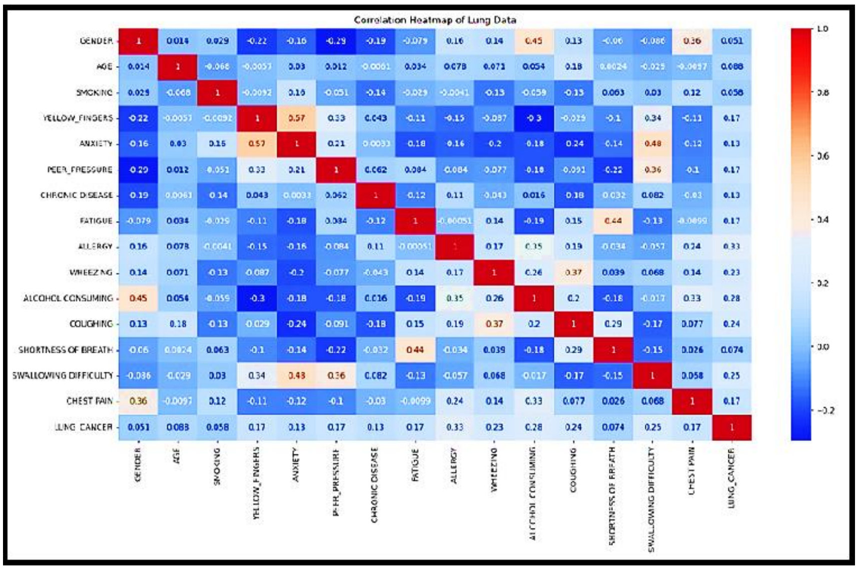
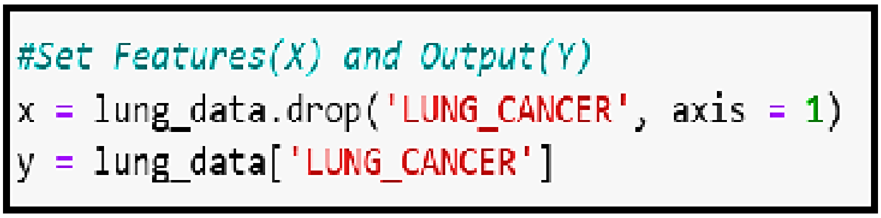


Figure 12: Snapshot of Correlation Heatmap

To create a logistic regression (LR) model, the authors must define the input variables as x, which should include all features except "LUNG_CANCER", and the output variable as y, which should consist of the "LUNG_CANCER" data series only, as shown in Figure. 13.



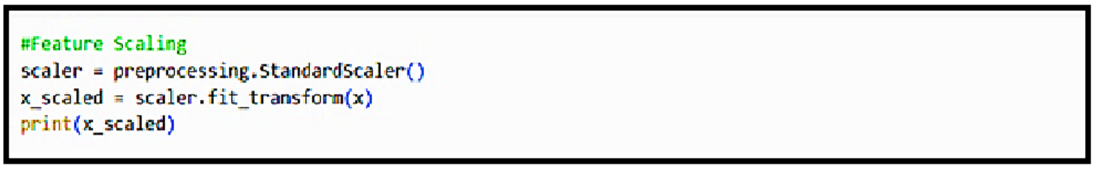
```
#Set Features(X) and Output(Y)
x = lung_data.drop('LUNG_CANCER', axis = 1)
y = lung_data['LUNG_CANCER']
```

Figure 13: Snapshot of Code Snippet

Before training the logistic regression model, it is essential to ensure that all input features contribute equally to the logistic regression model, so feature scaling (standardization of feature variables) is applied using StandardScaler from the sklearn library, as shown in Figure. 14. The following equation is used to find the standardized values of all feature variables.

$$x_{\text{stand}} = \frac{x-\mu}{\sigma} \tag{13}$$

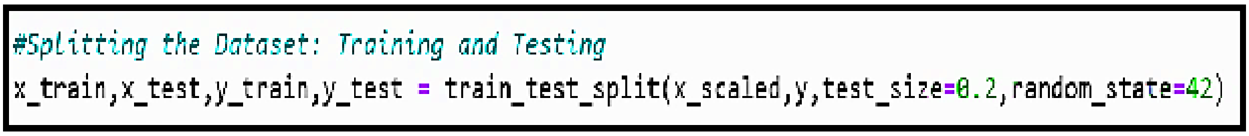
In this Figure. 14, the input feature set x is standardized using the “fit_transform” method, and the standardized values are stored in the x_scaled variable. This preprocessing step is crucial, especially for algorithms like logistic regression that are sensitive to the scale of input features.



```
#Feature Scaling
scaler = preprocessing.StandardScaler()
x_scaled = scaler.fit_transform(x)
print(x_scaled)
```

Figure 14: Snapshot of Code

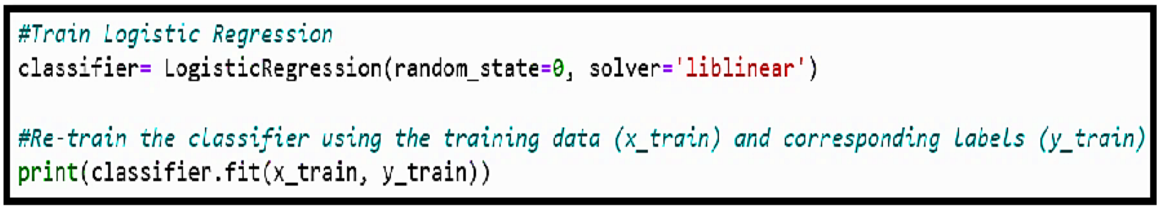
To evaluate model performance, the dataset was split into training and testing sets using the “train_test_split()” from the sklearn.model_selection module. In Figure. 15, test_size=0.2 illustrates that the feature variable (x_scaled) and target variable (y) are divided such that 80% of the data was used for training and 20% for testing purposes. Here, the random_state=42 parameter ensures the reproducibility of the data split. As a result, x_train and y_train are used to train the model, while x_test and y_test are used for testing.



```
#Splitting the Dataset: Training and Testing
x_train,x_test,y_train,y_test = train_test_split(x_scaled,y,test_size=0.2,random_state=42)
```

Figure 15: Snapshot of Code

To train the logistic regression model, the LogisticRegression class from the sklearn.linear_model module was used with the liblinear solver, which is efficient for small datasets and binary classification problems. The random_state parameter was set to ensure reproducibility. The model was trained using the method “fit()”, which takes the standardized training features (x_train) and their corresponding labels (y_train) as input. This process allows the model to learn the relationship between features and target labels by optimizing the weight parameters. The code snippet for this training process is shown in Figure. 16.



```
#Train Logistic Regression
classifier= LogisticRegression(random_state=0, solver='liblinear')

#Re-train the classifier using the training data (x_train) and corresponding labels (y_train)
print(classifier.fit(x_train, y_train))
```

Figure 16: Snapshot of Code

The model is evaluated using a confusion matrix, which is a table that evaluates the performance of a classification model by comparing its predictions to the actual values. In this manuscript, the obtained confusion matrix for both the stages such as training and the testing is depicted in Figure. 17.

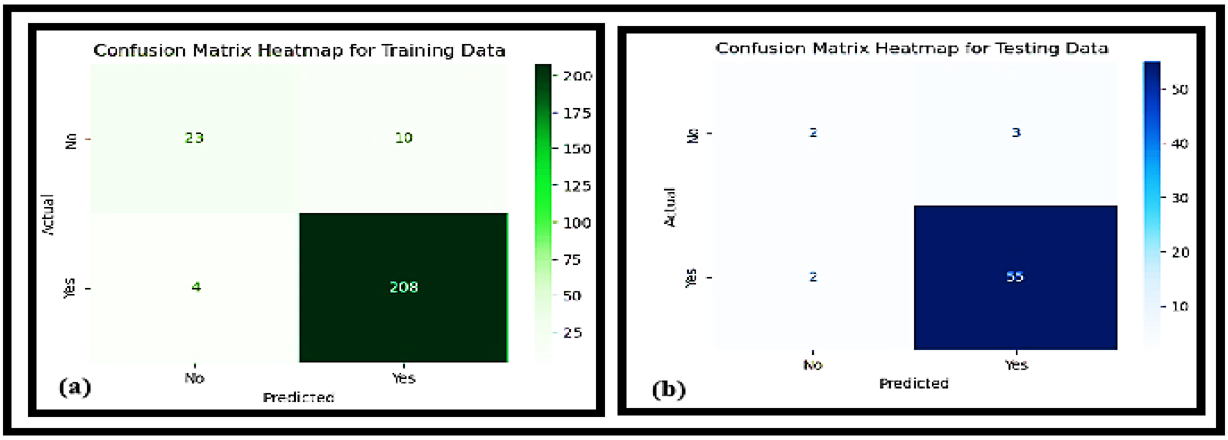


Figure 17: Snapshot of Confusion matrix

Above, Figure. 17.(a) and Figure. 17.(b) identify the confusion matrix for training data and testing data where the evolutionary parameters like Precision, Recall and F1 score, Accuracy are illustrated in Table 1.

Table 1:

	Precision	Recall	F1 score	Accuracy
Training	0.95	0.98	0.97	0.94
Testing	0.95	0.96	0.96	0.92

Evolutionary Parameters

According to Table 1, it is evaluated that the lung cancer prediction model demonstrates high performance and strong generalization capability. On the training dataset, it achieved a precision of 0.95, recall of 0.98, F1-score of 0.97, and accuracy of 0.94, indicating excellent learning from the training data. On the testing dataset, the model maintained similarly high results with a precision of 0.95, recall of 0.96, F1-score of 0.96, and accuracy of 0.92. This close alignment between training and testing metrics suggests that the model is neither overfitting nor underfitting and is well-suited for practical deployment.

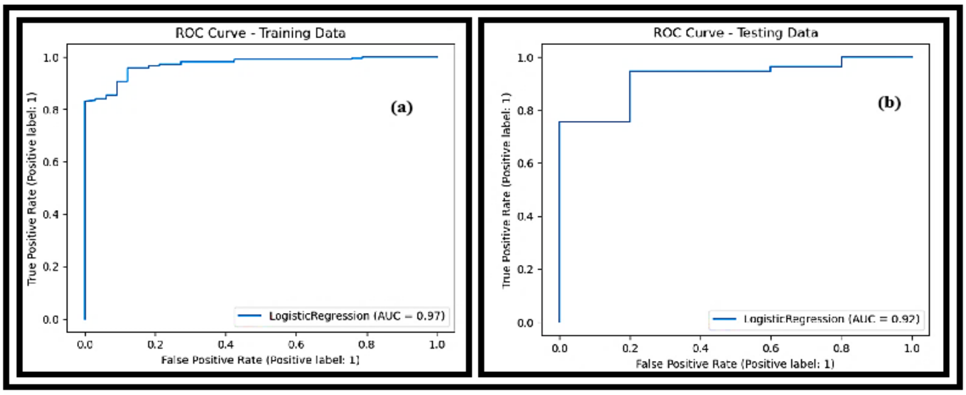


Figure 18: ROC Curves

This model is also estimated by using a Receiver Operating Characteristic (ROC) curve, which is a graphical plot used to show the diagnostic ability of a binary classification model. Figure. 18.(a) and Figure. 18.(b) illustrate that the Area Under the Curve (AUC) is **0.97** for the training data and **0.92** for the testing data, indicating that the logistic regression model demonstrates strong discriminatory power. The close proximity of the testing AUC to the training AUC suggests that the model generalizes well and is not overfitting. The curves rise steeply toward the top-left corner, highlighting a high true positive rate with a low false positive rate, which is critical for early and accurate lung cancer prediction.

5. Conclusion

The study shows how supervised machine learning, especially logistic regression, can be used effectively for early detection and risk prediction of lung cancer. The logistic regression model performed well by using a dataset that included important clinical and behavioral factors like smoking habits, anxiety, peer pressure, chronic diseases, fatigue, allergies, and other related symptoms. The model was created in Python with Jupyter Notebook and included steps like preprocessing, exploratory data analysis, feature engineering, and model training. The evaluation of the model confirmed its strength, achieving an F1-score of 0.97 for training data and 0.96 for testing data, along with high AUC values of 0.97 and 0.92, respectively. This research shows how logistic regression can provide valuable clinical insights and suggests further exploration with ensemble models, deep learning methods, or hybrid approaches to improve diagnostic accuracy.

6. Authors' Biography

Gourab Mukhopadhaya
Currently pursuing Bachelor of Technology in Computer Science and Engineering (CS&DS) at Brainware University, Barasat, West Bengal, India.

Suman Sett
Currently pursuing Bachelor of Technology in Computer Science and Engineering (CS&DS) at Brainware University, Barasat, West Bengal, India.

Sumit Basu
Currently pursuing Bachelor of Technology in Computer Science and Engineering (CS&DS) at Brainware University, Barasat, West Bengal, India.

Susmit Chakraborty
Currently working as an Assistant Professor in the Department of Computer Science and Engineering (CS&DS), Brainware University, Barasat, West Bengal, India.

Swarnendu Shil
Currently pursuing Bachelor of Technology in Computer Science and Engineering (CS&DS) at Brainware University, Barasat, West Bengal, India.

Arijit Sen

Currently pursuing Bachelor of Technology in Computer Science and Engineering (CS&DS) at Brainware University, Barasat, West Bengal, India.

Priti Munyan
Currently pursuing Bachelor of Technology in Computer Science and Engineering (CS&DS) at Brainware University, Barasat, West Bengal, India.

References

1. Boța, M., Vlaia, L., Jîjie, A. R., Marcovici, I., Crişan, F., Oancea, C., ... & Moacă, E. A. (2024). Exploring synergistic interactions between natural compounds and conventional chemotherapeutic drugs in preclinical models of lung cancer. Pharmaceuticals, 17(5), 598.
2. Atmakuru, A., Chakraborty, S., Faust, O., Salvi, M., Barua, P. D., Molinari, F., ... & Homaira, N. (2024). Deep learning in radiology for lung cancer diagnostics: A systematic review of classification, segmentation, and predictive modeling techniques. Expert Systems with Applications, 124665.
3. Mohandass, G. H. K. D. S. C. S. G., Krishnan, G. H., Selvaraj, D., & Sridhathan, C. (2024). Lung cancer classification using an optimized attention-based convolutional neural network with a DenseNet-201 transfer learning model on CT images. Biomedical Signal Processing and Control, 95, 106330.
4. Gayap, H. T., & Akhloufi, M. A. (2024). Deep machine learning for medical diagnosis, application to lung cancer detection: a review. BioMedInformatics, 4(1), 236-284.
5. Bhuiyan, M. S., Chowdhury, I. K., Haider, M., Jisan, A. H., Jewel, R. M., Shahid, R., ... & Siddiqua, C. U. (2024). Advancements in early detection of lung cancer in public health: a comprehensive study utilizing machine learning algorithms and predictive models. Journal of Computer Science and Technology Studies, 6(1), 113-121.
6. Maurya, S. P., Sisodia, P. S., Mishra, R., & Singh, D. P. (2024). Performance of machine learning algorithms for lung cancer prediction: a comparative approach. Scientific Reports, 14(1), 18562.
7. Neravetla, A. R., Samreen, S., Mohammed, A. S., Jiwani, N., & Logeshwaran, J. (2024, February). An Improved Lung Cancer Prediction Algorithm Using Generative Adversarial Network in Modern Healthcare. In 2024 International Conference on Integrated Circuits and Communication Systems (ICICACS) (pp. 1-6). IEEE.
8. Zhang, S., Yang, L., Xu, W., Wang, Y., Han, L., Zhao, G., & Cai, T. (2024). Predicting the risk of lung cancer using machine learning: A large study based on UK Biobank. Medicine, 103(16), e37879.
9. Sweet, M. M. R., Ahmed, M. P., Mozumder, M. A. S., Arif, M., Chowdhury, M. S., Bhuiyan, R. J., ... & Mamun, M. A. I. (2024). COMPARATIVE ANALYSIS OF MACHINE LEARNING TECHNIQUES FOR ACCURATE LUNG CANCER PREDICTION. The American Journal of Engineering and Technology, 6(09), 92-103.
10. Pathan, R. K., Shorna, I. J., Hossain, M. S., Khandaker, M. U., Almohammed, H. I., & Hamd, Z. Y. (2024). The efficacy of machine learning models in lung cancer risk prediction with explainability. PLOS ONE, 19(6), e0305035.