Lung Cancer Prediction Analysis using Machine Learning Techniques

Kanimozhi Subramanian (T00708068)

Dr. Erfanul Hoque

DASC 5420 – Theoretical Machine Learning

Department of Science

Thompson Rivers University

April 15, 2023

Table of Contents

Abstract

Lung cancer prediction aims to identify individuals who are at high risk of developing lung cancer based on various factors such as age, smoking history and other medical factors. The goal of lung cancer prediction is to develop accurate and reliable models that can be used to identify individuals who are at high risk of developing lung cancer. We have used different machine learning techniques such as full logistic regression, SVM, Penalized regression (Lasso and ridge) and neural network for predicting the lung cancer and analysed which model is best for predicting the lung cancer with high accuracy. In this research, lasso logistic regression outperforms all other models in terms of lung cancer prediction accuracy.

Introduction

Background and Context

Lung cancer is the leading cause of cancer-related deaths worldwide, with the majority of cases being caused by smoking. NSCLC is the most common type of lung cancer, accounting for about 85% of all cases. Early detection and diagnosis of lung cancer are crucial for successful treatment and improved survival rates. Machine learning techniques can be used to analyze large amounts of data, to predict the likelihood of lung cancer and assist with early detection. These techniques can also help in identifying the most effective treatments for individual patients based on their unique characteristics and medical history.

Hence, we will use the “survey lung cancer.csv” dataset from Kaggle to conduct Prediction analysis for early detection and treatment. We will try to predict the future risk of a heart attack in those 309 samples of the dataset.

Objective

The primary goal of this study is to use machine learning techniques such as logistic regression, SVM, neural network to estimate the chance of a lung cancer in a patient based on health factors like gender, age, smoking, yellow fingers, anxiety, peer pressure, chronic disease and other medical factors.

Exploratory Data Analysis

Correlation among the predictors: There is no correlation between the predictors when the correlation between the predictors is tested. Figure 2 correlation plot illustrates this.

Lung cancer is more prevalent in people over the age of 40, as seen by the distribution of age in lung cancer in Figure 1.

Author: Kanimozhi Subramanian

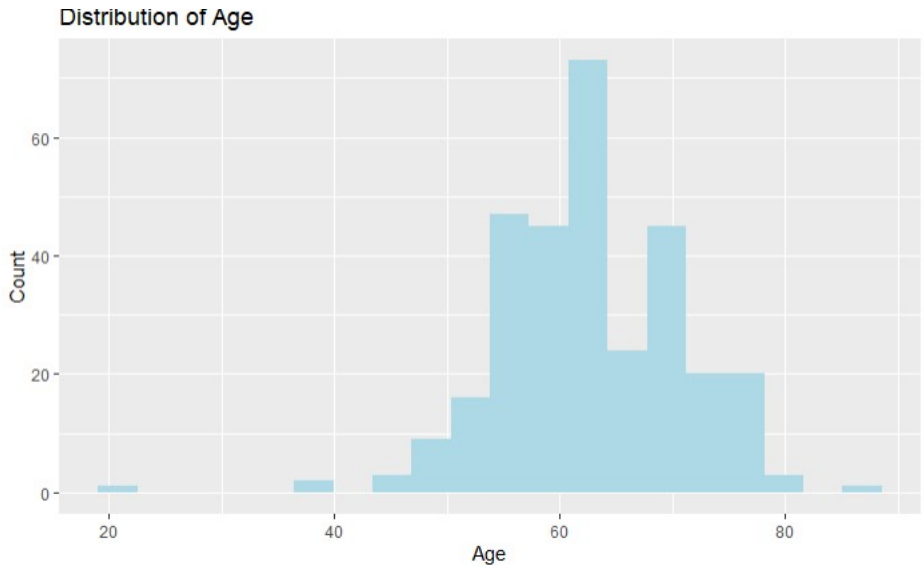

Figure 1: Distribution of Age in Lung cancer dataset

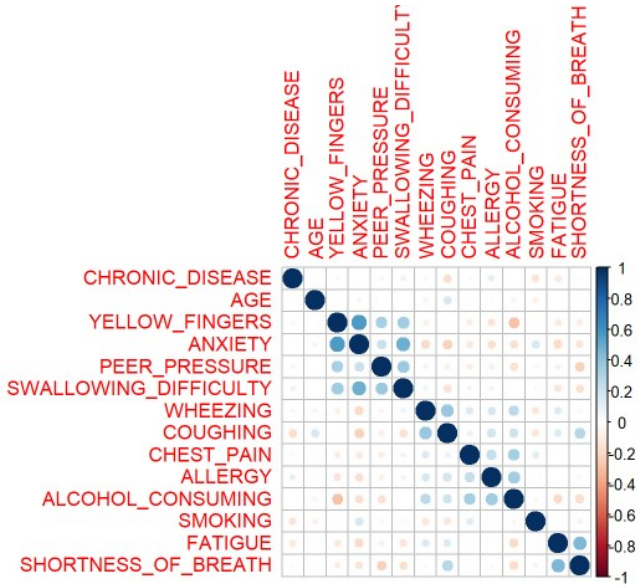

Figure 2: Correlation plot of predictors

Data Pre-processing

Removing missing values: We checked for missing values in the lung cancer disease dataset and we didn't find any missing values.

Splitting data into input features and output target: The lung cancer dataset is split into input features (X) and output target (y). The input features are all the columns except the last one,

which contains the output target. The output target is the presence or absence of lung cancer, which is stored in the LUNG_CANCER column.

Splitting data into training and test data:
The dataset is split into training and test data.75% of data is training data and 25% data is test data.

Scaling the input features: The input features are scaled. Scaling is an important step in data pre-processing because it ensures that all input features are on the same scale.

Logistic Regression

Logistic regression is a statistical method, can be used to predict the likelihood of a patient having lung cancer based on various clinical and demographic features.

The logistic regression model calculates the probability of a patient having lung cancer. The output of the model is a predicted probability between 0 and 1, with values closer to 1 indicating a higher likelihood of lung cancer.

The predictor variables used in logistic regression for lung cancer prediction can include a range of clinical and demographic factors, such as gender, age, smoking, yellow fingers, anxiety, chronic disease, fatigue, allergy, wheezing, alcohol consuming, and other medical factors.

Analysis

- Full logistic regression is carried out using all the predictors such as gender, age, smoking, yellow fingers, anxiety, peer pressure, chronic disease and other medical factors.
- The dataset is divided into 25% of test data and 75% of training data.
- For the purpose of developing the model, the outcome variable LUNG_CANCER is modified from its default values of "YES" or "NO" to 1 or 0, respectively.
- The model is fitted using train dataset using glm() function with family set to binomial.
- Next, the model is used to predict the probabilities of the test data using predict() function with type set to "response".
- Predicted classes are then determined based on whether the probability is greater than or less than 0.5.
- The performance of the model is determined using confusion matrix.

- The coefficients represent the log-odds of the probability of having lung cancer (the outcome variable) associated with each predictor variable.

Result

The model is predicting the probability of a binary outcome (e.g., the presence of a disease) based on several predictor variables.

- The model correctly classified 71 out of 76 instances (i.e., 93.42% accuracy) using confusion matrix.
- The model's sensitivity (true positive rate) was 55.56% and its specificity (true negative rate) was 98.51%.
- The positive predictive value (PPV) was 83.33%, meaning that when the model predicted the positive class (lung cancer), it was correct 83.33% of the time.
- The negative predictive value (NPV) was 94.29%, meaning that when the model predicted the negative class (no lung cancer), it was correct 94.29% of the time.
- The balanced accuracy takes into account both sensitivity and specificity and provides a measure of overall model performance. In this case, the balanced accuracy was 0.77032.
- The results suggest that the logistic regression model performs reasonably well at predicting lung cancer based on the given features, although there is room for improvement in terms of sensitivity.

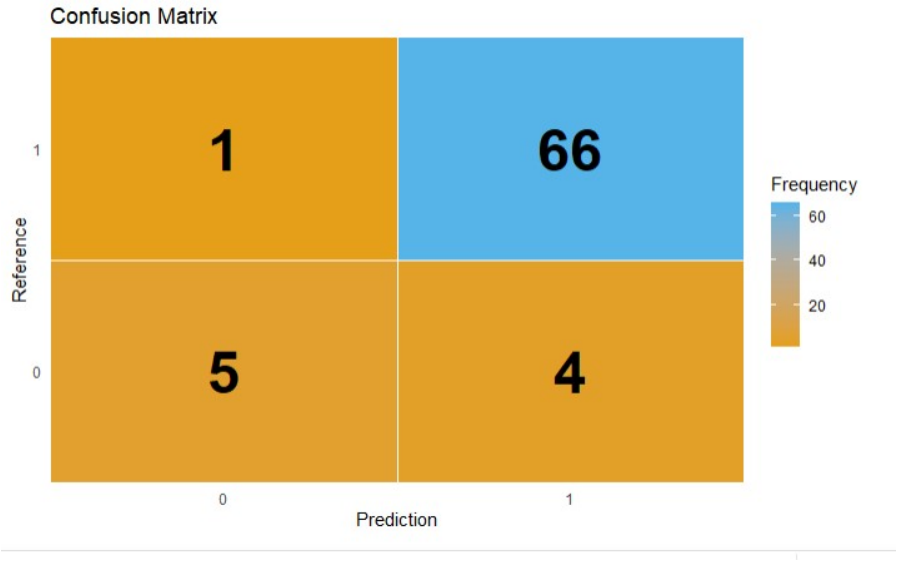

Figure 3: Confusion Matrix of logistic Regression

- Confusion matrix for a binary classification problem is shown in Figure 3 above, where 0 denotes no risk of lung cancer and 1 denotes risk of lung cancer.

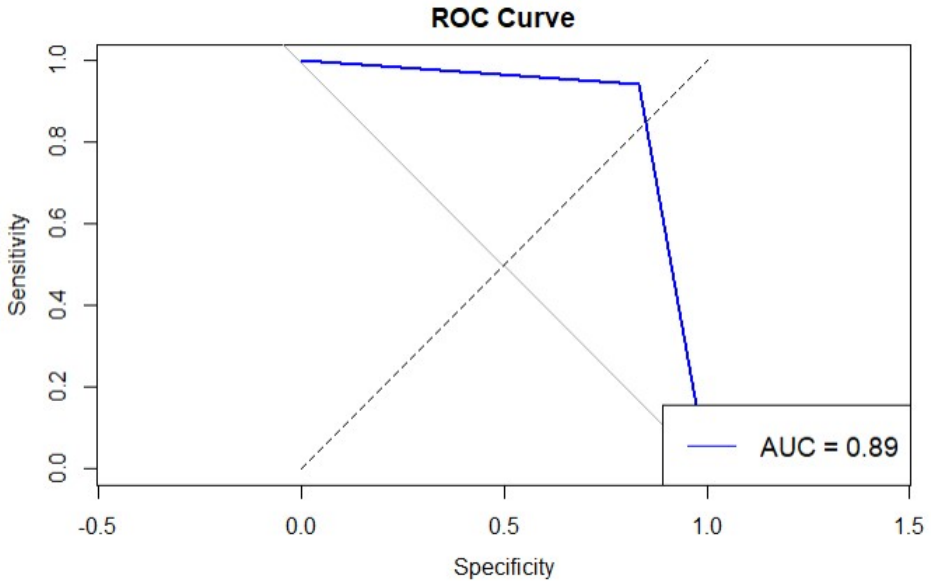

Figure 4: ROC Curve for Logistic Regression

- The AUC value is 0.89 in Figure 4, the model's accuracy is high as indicated by the ROC curve. More accuracy is achieved with a higher AUC value. Our model's accuracy is higher.

- An AUC of 0.89 indicates that the model has good discriminatory power and is able to distinguish between the positive and negative classes with a high degree of accuracy in predicting lung cancer.

Penalized Logistic regression

We will analyse the lung cancer dataset using penalized regression such as ridge and lasso logistic regression to improve the model's performance of logistic regression.

Ridge Logistic regression

Ridge logistic regression is a type of regularized logistic regression that is commonly used in predictive modelling tasks such as lung cancer prediction. The goal of ridge regression is to improve the model's performance and prevent over fitting by adding a penalty term to the logistic regression objective function.

In the context of lung cancer prediction, ridge logistic regression can be used to identify the most important predictors that are associated with the disease and to estimate the probability of an

individual developing lung cancer based on their demographic, lifestyle, and clinical characteristics. By using ridge regression, it is possible to build a more robust and accurate model that can help to improve early detection and treatment of lung cancer.

Analysis

- Once the data pre-processing is done on the dataset, the next steps are carried out to fit the ridge regression model to improve the performance.
- The model.matrix() function is used to create the design matrix x and the outcome variable y for the training data.
- The cv.glmnet() function is used to perform cross-validation to select the optimal value of the regularization parameter (lambda) for ridge regression. The alpha parameter is set to 0 to specify ridge regression, and the family parameter is set to "binomial" for binary classification
- The optimal value of lambda is obtained using cv.ridge$lambda.min.
- The glmnet() function is used to fit the final ridge regression model on the training data using the optimal value of lambda.
- The model.matrix() function is used to create the design matrix x.test for the test data, and predict() function is used to make predictions on the test data using the fitted model.
- The predicted classes are obtained by applying a threshold of 0.5 to the predicted probabilities. The observed classes are taken from the LUNG_CANCER column of the test. data frame.
- The accuracy of the model is calculated by comparing the predicted classes to the observed classes using the mean () function.

Result

The accuracy of a classification model using the mean of a logical expression that compares the predicted classes to the observed classes. The value of 0.9473684 suggests that the model is performing well, with an accuracy of 94.74%.

The value of cv.ridge$lambda.min is 0.01966874, which suggests that the optimal value of lambda for the ridge regression model is around 0.02. This means that the ridge regression model with this lambda value would be the best model for predicting the outcome variable, given the available data and the specific model specification. The minimum log lambda value is shown in the Figure 5 with the vertical line. The coefficients of ridge regression are never zero but it is reduced towards to zero as shown in Figure 6.

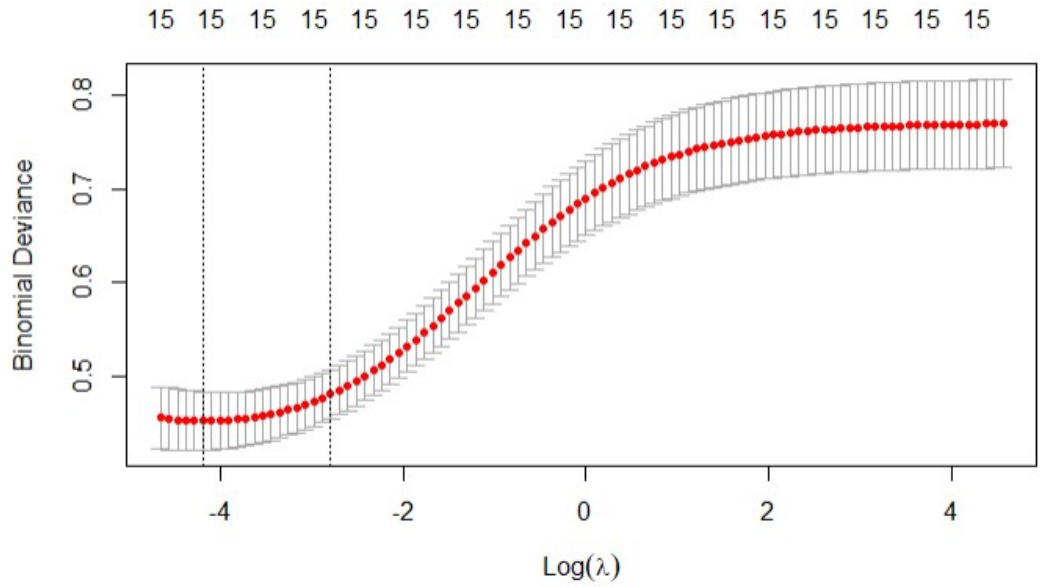

Figure 5: Cross validation plot of Ridge Regression

Predictors	Coefficients
Intercept	-15.59625394
GENDERM	0.09196379
AGE	0.01875057
SMOKING	0.61071255
YELLOW_FINGERS	0.8673353
ANXIETY	0.59852478
PEER_PRESSURE	0.90966838
CHRONIC_DISEASE	1.15659307
FATIGUE	1.12557344
ALLERGY	1.27630616
WHEEZING	0.96421586
ALCOHOL_CONSUMING	1.35733719
COUGHING	1.02142649
SHORTNESS_OF_BREATH	0.01113638
SWALLOWING_DIFFICULTY	1.26387715
CHEST_PAIN	0.12606282

Figure 6: Ridge Regression Coefficients

Lasso Logistic Regression

Lasso regression is a type of linear regression that uses regularization to prevent over fitting. It shrinks the coefficients of the regression variables towards zero, effectively performing variable selection by setting some coefficients to exactly zero. This is done by minimizing the sum of squared errors subject to a constraint on the sum of the absolute values of the coefficients.

In the context of lung cancer prediction, Lasso regression can be used to identify the most important risk factors for lung cancer and to build a predictive model using these risk factors. The Lasso regression can also help to avoid over fitting, which can lead to more accurate predictions on new data.

Analysis

- Once the data pre-processing is done on the dataset, the next steps are carried out to fit the lasso regression model to improve the performance.

- The cv.glmnet() function is used to perform cross-validation to select the optimal value of the regularization parameter (lambda) for lasso regression. The alpha parameter is set to 1 to specify lasso regression, and the family parameter is set to "binomial" for binary classification.

- The optimal value of lambda is obtained using cv.lasso$lambda.min.

- The glmnet() function is used to fit the final lasso regression model on the training data using the optimal value of lambda.

- The model.matrix() function is used to create the design matrix x.test for the test data, and predict() function is used to make predictions on the test data using the fitted model.

- The predicted classes are obtained by applying a threshold of 0.5 to the predicted probabilities. The observed classes are taken from the LUNG_CANCER column of the test.data dataframe.

- The accuracy of the model is calculated by comparing the predicted classes to the observed classes using the mean () function.

Result

The coefficients of predictors such as gender and Shortness of breath are reduced to zero and is neglected from the model as it doesn’t contribute any value to the prediction of the outcome variable LUNG_CANCER and is shown in Figure 8.

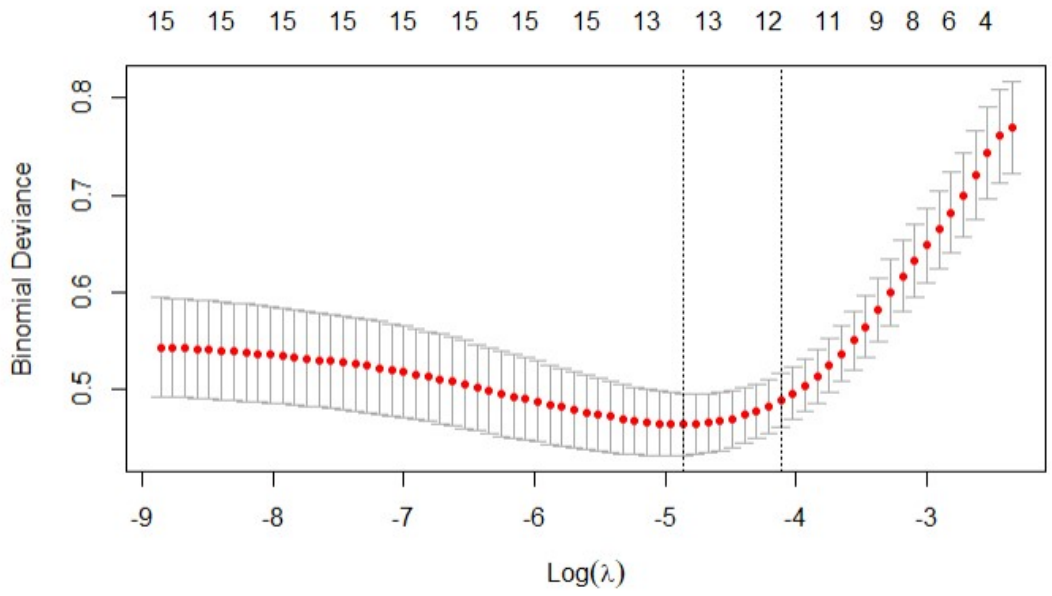

Figure 7: Lasso Regression plot for Cross Validation

The lambda.min value returned by the function represents the value of lambda that yields the minimum mean cross-validated error, indicating the best trade-off between bias and variance in the model. In this case, the optimal value of lambda is 0.007745129. The minimum log lambda value is shown in the Figure 7 with the vertical line.

A positive coefficient indicates that an increase in the variable is associated with an increase in the log-odds of having lung cancer, while a negative coefficient indicates the opposite as shown in the Figure 8.

Predictors	Coefficients
Intercept	-18.35
GENDERM	.
AGE	0.02
SMOKING	0.96
YELLOW_FINGERS	1.07
ANXIETY	0.26
PEER_PRESSURE	1.10
CHRONIC.DISEASE	1.64
FATIGUE	1.52
ALLERGY	1.21
WHEEZING	0.81
ALCOHOL.CONSUMING	0.79
COUGHING	1.71

SHORTNESS.OF.BREATH	.
SWALLOWING.DIFFICULTY	1.84
CHEST.PAIN	0.21

Figure 8: Lasso Regression Coefficients

The model obtained an accuracy of 97.4% on the test data, which indicates that it performs well in predicting the presence or absence of lung cancer based on the selected variables.

SVM

Support Vector Machines (SVM) is a supervised machine learning algorithm that can be used for classification tasks, such as predicting the likelihood of lung cancer based on certain features. SVM works by finding the hyper plane in a high-dimensional space that best separates the data into different classes. The hyper plane is determined by maximizing the margin between the two classes, where the margin is defined as the distance between the hyper plane and the nearest points of each class.

Analysis

- Once the data pre-processing is done on the dataset, the next steps are carried out to fit the SVM regression model to improve the performance.
- An SVM model is created by specifying a train control object with 10-fold cross-validation. Train control specifies the type of re-sampling method used to evaluate the model's performance.
- The output target variable is converted into a factor variable with levels 0 and 1. This is important because SVM requires the output target variable to be a factor with levels indicating the presence or absence of lung cancer disease.
- The SVM model is trained on the training data using the train() function and considered the output target variable "y" to be predicted using all the input features in "X". The "svmLinear" algorithm was chosen to predict the SVM model.
- Using the SVM model, we predicted the output target variable that indicates the presence or absence of lung cancer disease based on the scaled input features of the test data.
- The confusion matrixes of the predicted and actual output target variables are calculated.

Result

- Out of the 76 test samples, the model correctly predicted 2 patients as negative for lung cancer (true negative) and 61 patients as positive for lung cancer (true positive).

However, it incorrectly classified 6 patients as negative (false negative) and 7 patients as positive (false positive).

- The overall accuracy of the model is 0.8553, indicating that the model correctly classified 85.53% of the test samples.
- The sensitivity of the model is 22.22%, which means that the model correctly identified 22.22% of the cases with lung cancer.
- The specificity is 94.03%, which means that the model correctly identified 94.03% of the cases without lung cancer.
- The positive predictive value is 33.33%, which means that the probability of having lung cancer given a positive prediction from the model is 33.33%.
- The negative predictive value is 90.00%, which means that the probability of not having lung cancer given a negative prediction from the model is 90.00%.
- The balanced accuracy is 58.13%, which is the average of sensitivity and specificity. It indicates the overall effectiveness of the model in identifying both positive and negative cases.

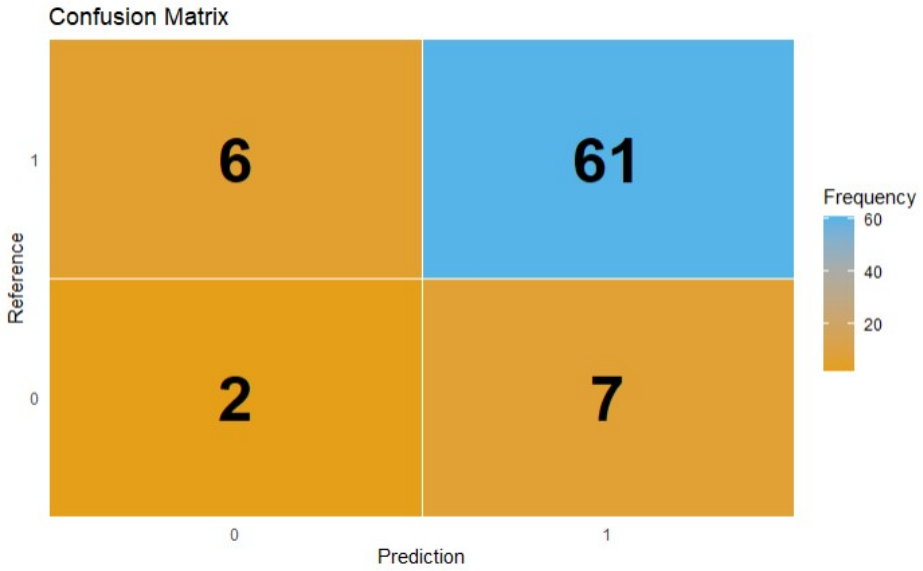

Figure 9: Confusion Matrix of Support Vector Machine Regression.

Confusion matrix for a binary classification problem is shown in Figure 9 above, where 0 denotes no risk of lung cancer disease and 1 denotes risk of lung cancer disease. The confusion matrix shows the number of true positives (2), false positives (7), false negatives (4), and true negatives (63)

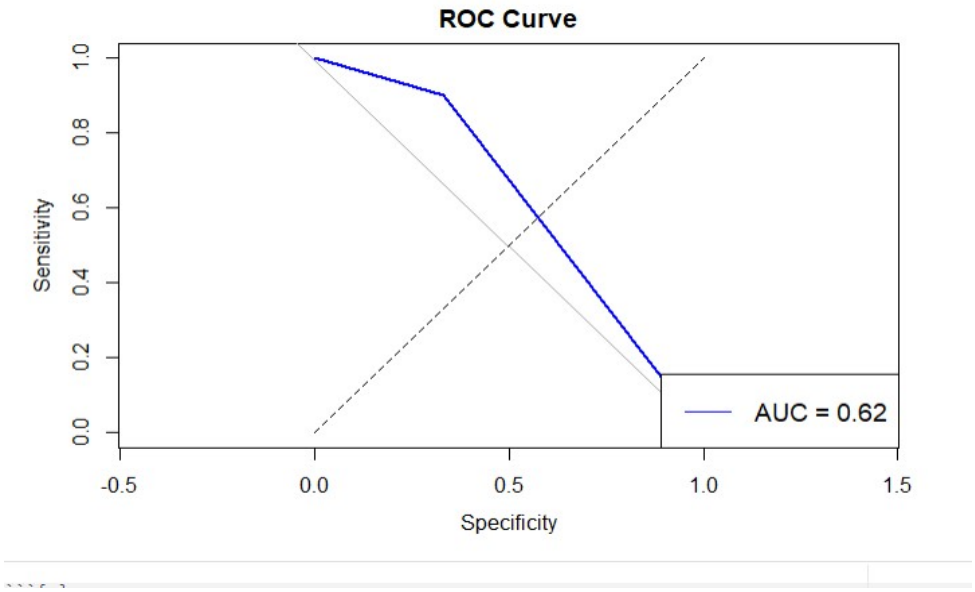

Figure 10: ROC Curve for Support Vector Machine Regression

As per above Figure 10, more accuracy is achieved with a higher AUC value. Our model's accuracy is slightly higher, as our model's AUC value is 0.62.

Neural network

A neural network is a type of machine learning model inspired by the structure and function of the human brain. In the context of lung cancer prediction, a neural network can be trained to analyze patient data and predict the likelihood of developing lung cancer.

One advantage of using a neural network for lung cancer prediction is that it can incorporate a wide range of factors and identify complex patterns in the data that may not be obvious to human experts. This can improve the accuracy of the predictions and potentially lead to earlier detection and better outcomes for patients.

Analysis

- Once the data pre-processing is done on the dataset, the next steps are carried out to fit the neural network model to improve the performance.

- A neural network model is created using the neuralnet() function from the neuralnet library. The formula for the model is created using the formula() function and specifying the response variable "LUNG_CANCER" and the predictor variables from the dataset. The neural network model has two hidden layers with 10 neurons in each layer, an activation function of logistic and linear output as shown in Figure 10.

- The neural network model is used to predict lung cancer outcomes on the testing set using the compute() function.
- The predicted outcomes are descaled and compared to the actual outcomes using confusion matrix.

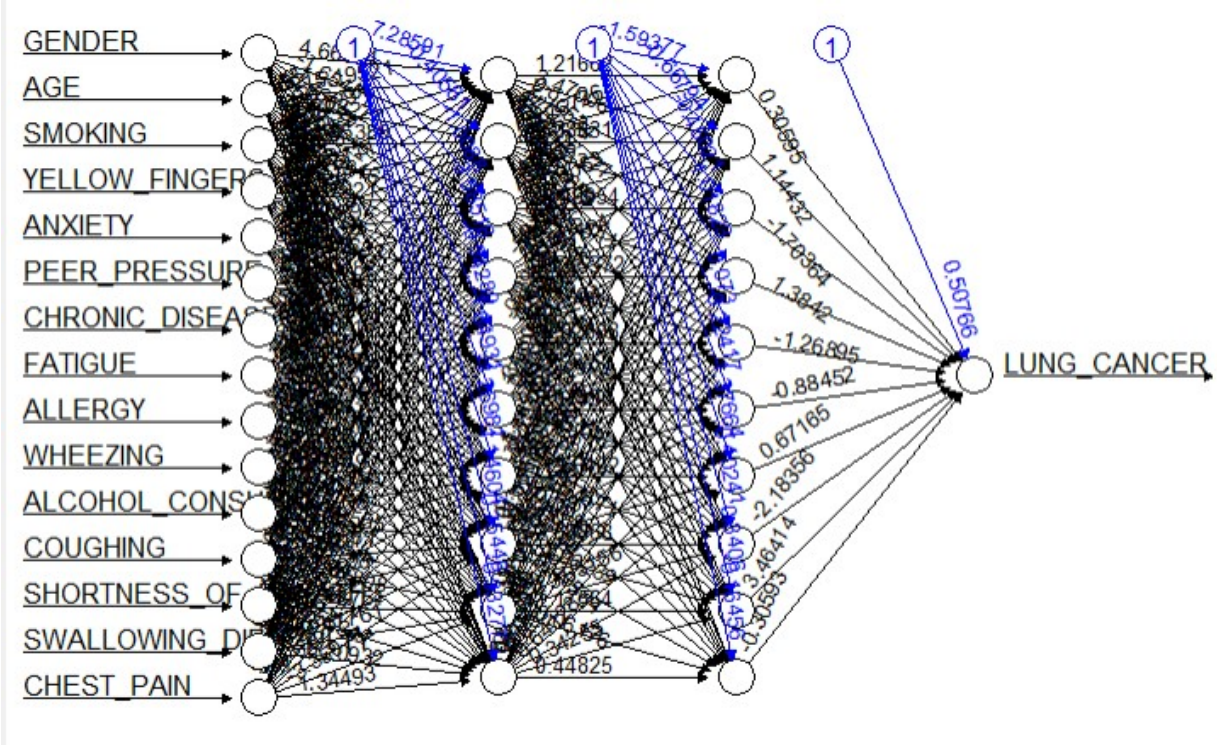

Figure 10: Neural Network Diagram with Weights

Result

- The model correctly predicted 6 instances of the negative class (0) and 64 instances of the positive class (1), while incorrectly predicting 2 instances of the negative class and 5 instances of the positive class and is shown in the Figure 11.
- The accuracy of the model is 0.9091, which means that it correctly predicted the class of 90.91% of the instances.
- The 95% confidence interval (CI) ranges from 0.8216 to 0.9627.
- The sensitivity of the model is 0.54545, which means that it correctly identified 54.54% of the instances of the positive class.
- The specificity of the model is 0.96970, which means that it correctly identified 96.97% of the instances of the negative class.

- The positive predictive value (PPV) of the model is 0.75000, which means that when the model predicted a positive class, it was correct 75% of the time.
- The negative predictive value (NPV) of the model is 0.92754, which means that when the model predicted a negative class, it was correct 92.75% of the time.

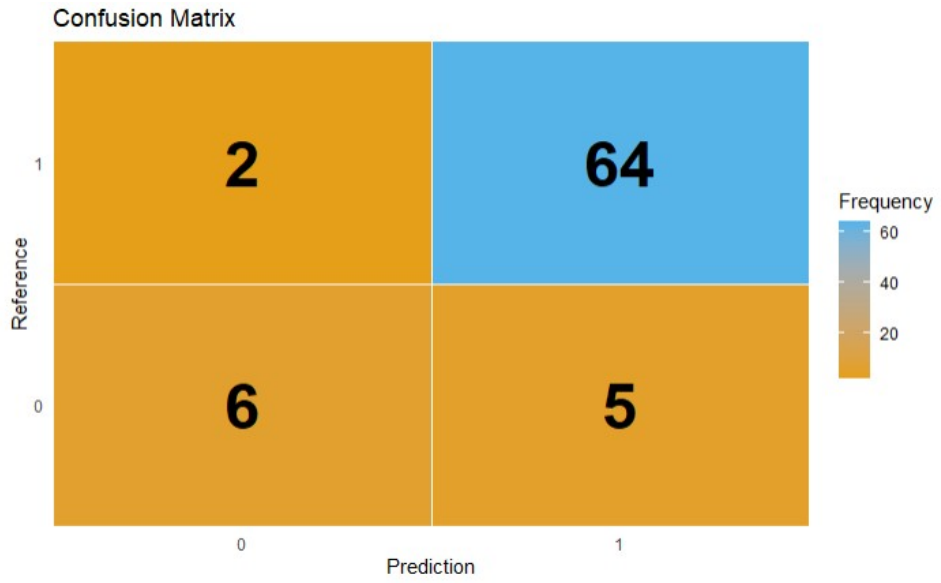

Figure 11: Confusion matrix of Neural Network

Comparison of Models

To determine the most accurate model, we performed lung cancer prediction using a variety of machine learning techniques. The accuracy of all the models is displayed in Figure 12.

Based on accuracy, Lasso Logistic Regression is more accurate than the other models. However, all of the models, including neural networks, ridge regression, complete logistic regression, and SVM, have excellent prediction accuracy of more than 85%, making them useful in the medical area for lung cancer prediction.

Full logistic regression	Ridge regression	Lasso regression	SVM	Neural network
93.42%	94.74%	97.40%	85.53%	90.91%

Figure 12: Accuracy of Models

Data Collection Method

We took the dataset from Kaggle with the file name "survey lung cancer.csv." It includes columns for age, gender, smoking, yellow fingers and so on. The dataset contains 15 columns and 309 observations.

The data available in the lung cancer dataset and its characteristics are given below.

- Age: age of the patient
- Sex: gender of the patient
- Smoking: Categorical variable (2-YES,NO-1)
- Yellow fingers: Categorical variable (2-YES,NO-1)
- Anxiety: Categorical variable (2-YES,NO-1)
- Peer_ pressure: Categorical variable (2-YES,NO-1)
- Chronic Disease: Categorical variable (2-YES,NO-1)
- Fatigue: Categorical variable (2-YES,NO-1)
- Allergy: Categorical variable (2-YES,NO-1)
- Wheezing: Categorical variable (2-YES,NO-1)
- Alcohol: Categorical variable (2-YES,NO-1)
- Coughing: Categorical variable (2-YES,NO-1)
- Shortness of Breath: Categorical variable (2-YES,NO-1)
- Swallowing Difficulty: Categorical variable (2-YES,NO-1)
- Chest Pain: chest pain (1 – No chest pain, 2 – Chest Pain)

Conclusion

Lung cancer prediction analysis can help identify individuals who are at high risk for developing lung cancer prediction before any symptoms appear and can provide personalized care that is tailored to the individual patient's needs. This can allow for early interventions to prevent or slow the progression of lung cancer prediction.

Overall, lung cancer prediction analysis can help improve patient outcomes, reduce healthcare costs, and promote public health. It is an important tool for healthcare providers and policymakers in the fight against lung cancer disease, which remains one of the leading causes of death worldwide.

References

- Sandra Grace Nelson (2021) *Lung cancer prediction dataset*, *Kaggle*. Available at: https://www.kaggle.com/code/sandragracenelson/lung-cancer-prediction

- Erfanul Hoque (2023) *Neural Network prediction, Available at :* https://moodle.tru.ca/mod/folder/view.php?id=2103753

- Erfanul Hoque (2023) *Regularization, Available at :* https://moodle.tru.ca/pluginfile.php/2793661/mod_resource/content/4/DASC-5420-Lab---Regularization_Solution.pdf

- Erfanul Hoque (2023) *SVM, Available at :* https://moodle.tru.ca/pluginfile.php/2785669/mod_resource/content/2/Unit6_Support-Vector-Machines.pdf

Git hub link

https://github.com/Kani042/MachineLearning-code