2023

LUNG CANCER PREDICTION

DEEP NEURAL NETWORKS AND LEARNING SYSTEMS
ASHIKA SRINIVASAN
32066221

Contents

INTRODUCTION

Lung carcinoma constitutes one of the most often types of cancer and causes a large number of deathsglobally. Early identification of lung cancer is critical for effective treatment and better patient outcomes. The symptoms of lung cancer emerge in the end stage. As a consequence, when compared to all other types of cancer, the death rate for pulmonary cancer is exceptionally high. Tiny cell lung cancers and non-little cell lung cancers are two kinds of lung cancer that develop and spread in unusual ways. Lung cancer ismostly predicted and diagnosed using computed tomography (CT) images. Disease should be treated in the patient as soon as feasible, particularly in tumors. Predicting the possibility of acquiring lung cancer can aid in the identification of high-risk patients who may benefit from screening or preventative interventions**.**

LITERATURE REVIEW

In order to help with early diagnosis and prevention, the multi-layer perceptron (MLP) technique has demonstrated encouraging results in predicting the likelihood of lung cancer.Multiple hidden layers is a concept proposed by Xiao et al. This work presents a multi-hidden layer method for obtaining parameter characteristics from the first layer. For analysis, a three-hidden layer structure is used. Hidden neurons with activation functions are present in the three-layer hidden-layer network structure. Many regression and classification studies are carried out as a resultof this. The findings reveal that the suggested method achieves good results. Panchal et al. demonstrate a method for selecting nodes in a hidden layer. The first is backpropagation. To teach the network, the procedure begins with a single hidden layer and a pair of neurons, the overall amount of neurons is rapidly increased. According to some findings, the backpropagation approach is stable. Furthermore, the investigation discovered that using an appropriate number of hidden nodes yields a better outcome with fewer training hours. However, raising the count of layers that are hidden enhances efficiency while complicating the network. Finally, the number of secret nodesis believed to be determined by the similarity of the input data. The study found that MLP may bea useful technique for identifying those who are at high risk of lung cancer. Overall, these studies show that MLP has the potential to predict lung cancer.

DEFINITION AND ANALYSIS OF A PROBLEM

The term *lung cancer prediction* refers to the use of a variety of techniques, including genetic testing, statistical analysis, and medical imaging, to determine a person's chance of acquiring lung cancer. In order to implement proper screening and early detection procedures, lung cancer prediction aims to identify those who are most likely to get the disease. The dataset contains a range of factors related to cancer patients and air pollution. The prediction models take into consideration a number of risk variables, including age, family history, smoking history, and exposure to environmental contaminants, among others, to find cancer incidence rates. Technology has enabled the gathering of large amounts of data, which can be used to uncover previously undiscovered patterns or correlations.

The study's goal is to develop a prediction model that can correctly identify people whoare at a higher risk of acquiring lung cancer. One area where neural networks may be extremely useful is in medical diagnostics, including lung cancer treatment. A neural network might be trainedto seek out possible signs of lung cancer, such as the existence of nodules or tumors, in order to examine the data. The neural network would learn to distinguish behavior that is indicative of malignancy in order to evaluate if a certain patient is at risk for lung cancer. To build such a neuralnetwork, a massive dataset of both positive and negative lung cancer cases would be required. To ensure that the network learns to recognize lung cancer accurately while limiting false positives and false negatives.

After training, the neural network might be utilized to analyze new data and give new insights to aid in the identification and management of lung cancer. It may enhance patient outcomes by detecting potential lung cancer cases earlier and adopting preventative measures.

The goal is to provide a dependable and accurate model for predicting lung cancer using data by developing an accurate and strong prediction model with careful preprocessing, suitable model selection, and assessment. This can aid in the early identification and treatment of lung cancer, thereby increasing patient survival rates. The Multi-Layer Perceptron (MLP) model seeks to predict the existence of lung cancer in patients to achieve high accuracy in determining the risks. The scope of the model can also help in the creation of tailored treatment regimens for lung cancer patients based on their unique medical characteristics. The MLP model might, in general, increase the precision and effectiveness of lung cancer detection and therapy.

A multilayer perceptron (MLP) was utilized in a 2018 study that was published Second International Conference to predict lung cancer. The study discovered that the MLP model outperformed other machine-learning methods with an AUC-ROC of 98.31%. These findings show the ability of MLP models to effectively predict lung cancer diagnosis and offer a strong justification for utilizing MLP models for lung cancer prediction. By allowing for earlier identification and more precise diagnosis of lung cancer, the application of MLP models for lung cancer prediction has the potential to enhance patient outcomes. One study that employed an MLP neural network to detect lung cancer based on chest CT images was published in the journal Computers in Biology and Medicine in 2017. According to the study, the MLP model had a high area under the receiver operating characteristic curve (AUC-ROC) for lung cancer detection, which was 0.933. These results show the potential of the models to offer a strong justification for the use of MLP neural networks in predicting lung cancer.

By using a neural network to predict lung cancer, the disease may be detected at a more manageable and earlier stage. A neural network can provide patients undergoing lung cancer screening with faster and more accurate results, reducing anxiety and improving the patient experience. One advantage of adopting a multilayer perceptron (MLP) for lung cancer prediction is its capacity to be trained on massive datasets of lung cancer patient data to produce predictions that are more accurate. An MLP can assist with prevention measures by identifying individuals who are more likely to develop lung cancer, potentially decreasing the need for more expensive treatments. In general, the application of an MLP for lung cancer prediction has the potential to enhance patient outcomes, lower medical expenses, and boost the effectiveness of lung cancer screening and treatment initiatives. Overall, the advantages of a neural network for lung cancer prediction are robust, and it may have an enormous effect on lung cancer screening and therapy.

SPECIFICATION

An MLP is made up of several layers of nodes or *neurons* organized in a feedforward design. It has an input layer, a hidden layer, and an output layer. Each neuron gathers information from the neurons in the layer below, conducts a weighted sum of the inputs, and then applies a nonlinear activation function to the output. The neurons in the next level output are then exposed to a nonlinear function of activation. The network's predictions are created by the output layer oncethe input layer has received the input information. Calculating the difference between the projectedand actual output and using backpropagation to reduce the error. The network's performance on a specific task must then be adjusted by altering the number of hidden layers and the number of neurons in each layer, which are both hyperparameters. MLPs are flexible and frequently utilized in a wide range of applications, such as classification, regression, and unsupervised learning.

In general, more neurons in the hidden layers can help the network better represent complicated connections between the input and output data. The input layer will contain as many neurons as there are characteristics in the dataset. Here are 22 features. The identity function, which simply outputs the input values without alteration, is frequently employed as the activation function in this layer.

Identifying the underlying patterns in the data is the job of the hidden layers. Through experimentation, the count of hidden layers and neurons that can be adjusted in each layer should be the same as in the trained network. Here, one hidden layer is used to capture the relationships in the data. The rectified linear unit (ReLU), whose definition is f(x) = max(0,x), is frequently the activation function employed in these layers.

ReLU's activation function's summation formula,

$$\mathbf{Y = f(w1x1 + w2x2 + \ldots + wn*xn + b)}$$

The bias term *b* is added to the weighted sum of the inputs (w1x1 + w2x2 +... + wn*xn) in this formula after the weighted sum has already been computed. The ReLU function returns zero, simply *turning off* the neuron if the weighted total of the inputs plus the bias term is less than zero. The ReLU function generates the input value, thereby *activating* the neuron if the weightedsum of the inputs plus the bias term is higher than or equal to zero.

The output layer is made up of just one neuron, which makes the final determination. Softmax is a popular activation function for the output layer that produces a probability distribution for each class, ensuring that the sum of probabilities is equal to one. Also, it generates a continuous, smooth, and stable training process and lessens the risk of overfitting.

The MLP will be trained on a dataset that has been labeled as 0, 1, and 2. The trained network's weights and biases should be stored, and the MLP's prediction function should be able to take in new data and make predictions. Overall, the architecture provided by the specification is used to construct an MLP neural network for lung cancer detection.

IMPLEMENTATION

Here's a more detailed description of the procedure to implement an MLP.

Importing the common libraries and assigning the generated data frame to a variable named *cancer_data*. Using the drop(), the axis is set to 1, indicating that the procedure should eliminate the given columns, and in place is set to True, indicating that the cancer_data should be modified directly rather than returning a new data frame. The measure of missing data in every column of the cancer_data data frame will be displayed in the output. If there are no missing values, 0 will be displayed for each column in the output.

It is possible to understand lung cancer risk factors by examining the correlation between these characteristics and the incidence of the disease, which can help with the creation of more precise prediction models. Splitting the training and testing data to ensure accuracy, test_size is used for testing and is commonly set between 0.2 and 0.3. Random_state is used to shuffle the data at random before dividing. The StandardScaler() class scales a dataset by dividing it by its standard deviation after subtracting the mean value of each feature from the dataset.

The hidden_layer_sizes hyperparameter specifies the neurons count in the MLP neural network's single hidden layer. There is just one hidden layer, which has 100 neurons. The activation function used here is the rectified linear unit. The optimizer used is Adam, and to update the neural network weights during training, the solver hyperparameter is specified. The normalization intensity used to avoid overfitting is specified by the alpha hyperparameter. The 'constant' value of the learning_rate hyperparameter designates the training's learning rate schedule.

The output layer activation function of the trained MLP classifier is then set to "softmax" or "logistic." The softmax function enables the model to produce a probability distribution over the many classes rather than simply a single class prediction. Here, the output layer could produce a multi-class classification indicating the cancer risks as 0, 1, and 2. (Low, Medium, High)

$$\mathbf{softmax(x) = exp(x)/sum(exp(x))}$$

The following steps would be involved in training a multi-layer perceptron (MLP) to predict lung cancer:

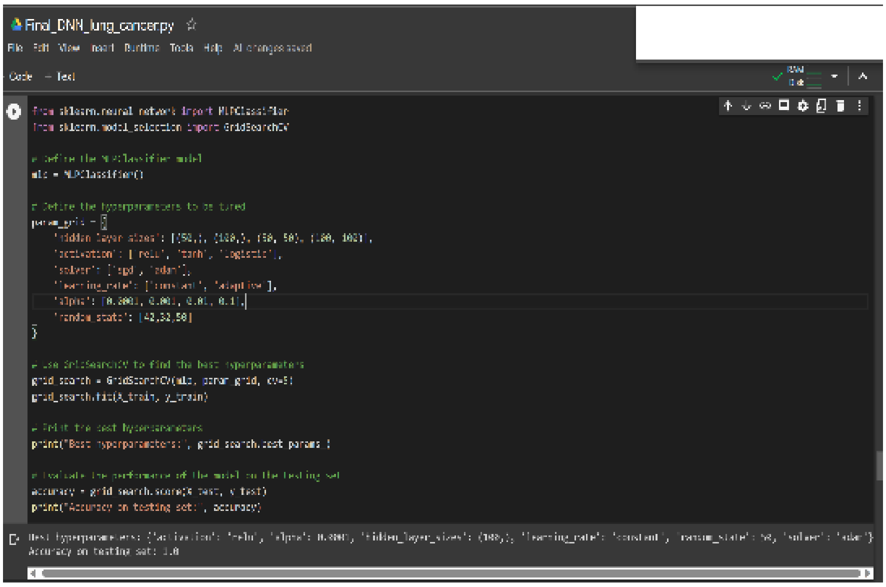
1. Preprocessing the data: The dataset of patients with lung cancer is processed by deleting any redundant features, handling missing data, and encoding categorical variables. To make sure that each feature contributes equally to the model, it is crucial to scale the features before training the MLP. The StandardScaler class is used to scale the features.

2. Splitting the data: The dataset would be used to generate the training, validation, and testing sets. The set for validation would be utilized to fine-tune and avoid overfitting, the testing set to assess the MLP's eventual performance, and the training set to train it.

3. Designing the model: MLP for predicting the presence of lung cancer consists of an input layer, one hidden layer with ReLU activation functions, and an output layer with a softmax function.

4. Model training: The fit approach is used to train the MLP using the training data. The forward and backward propagation techniques would be run iteratively over a number of epochs or until a convergence condition is satisfied while utilizing the training set to train the MLP.

5. Hyperparameter tuning: Need to adjust the MLP's hyperparameters, such as the learning rate, the number of hidden layers, and neurons in each hidden layer. Then do a grid search across a number of hyperparameters using the GridSearchCV class. Techniques like hyperparameter tweaking and feature selection are used to increase the model's precision on the testing set.



```
from sklearn.neural_network import MLPClassifier
from sklearn.model_selection import GridSearchCV

# Define the MLPClassifier model
mlp = MLPClassifier()

# Define the hyperparameters to be tuned
param_grid = {
    'hidden_layer_sizes': [(50,), (100,), (50, 50), (100, 100)],
    'activation': ['relu', 'tanh', 'logistic'],
    'solver': ['sgd', 'adam'],
    'learning_rate': ['constant', 'adaptive'],
    'alpha': [0.0001, 0.001, 0.01, 0.1],
    'random_state': [42,32,58]
}

# Use GridSearchCV to find the best hyperparameters
grid_search = GridSearchCV(mlp, param_grid, cv=5)
grid_search.fit(X_train, y_train)

# Print the best hyperparameters
print("Best hyperparameters:", grid_search.best_params_)

# Evaluate the performance of the model on the testing set
accuracy = grid_search.score(X_test, y_test)
print("Accuracy on testing set:", accuracy)
```

```
Best hyperparameters: {'activation': 'relu', 'alpha': 0.0001, 'hidden_layer_sizes': (100,), 'learning_rate': 'constant', 'random_state': 58, 'solver': 'adam'}
Accuracy on testing set: 1.0
```

6. Model evaluation: Metrics including accuracy, precision, recall, F1 score, and area under the

ROC curve were used to assess the MLP's final performance on the testing set. The score method, which provides the mean accuracy on the supplied test data and labels, will be used to assess the performance of the MLP on the validation set.

7. Model deployment: The MLP may be used to forecast the diagnosis of lung cancer in new patients once it has been trained.

Overall, by accurately predicting lung cancer based on patient data, the deliverable's successful implementation has the potential to improve patient outcomes. By computing these parameter gradients and making the necessary adjustments, the backpropagation method updates these parameters.

EVALUATION OF NEURAL NETWORKS

A neural network has been evaluated by preprocessing the data by removing errors and finding the missing values. As a feature distribution, the resultant figure offers an immediate visual summary of the distribution of each characteristic in the dataset, which is helpful for spotting possible problems such as skewed distributions or missing data. By analyzing the correlation matrix, a feature selection tool can recognize which features are significantly associated with each other and utilize these to aid in feature selection. Also used to identify which features have the strongest correlation. Plotting the graphs as distribution plots for individual features by class and pie charts. In the case of lung cancer prediction features such as age, gender, smoking status, medical history, and genetic markers may be relevant. The neural network can be trained and evaluated using different feature sets to determine which features are most relevant for predicting lung cancer.

The KFold class is used in this code to build a 4-fold cross-validation process. In the cross-validation process, the dataset is divided into four equal parts, each of which is used once as the validation set, while the other three parts are used to train the model.

Once fitting the model to the training set of information, its performance is assessed on the validation set using the *score()* method. The mean accuracy of the model on the validation set is 1.0, the test set is 1.0, and the training set is 1.0. The accuracy_score() method is then used to determine the model's predicted values on the test set, which are 0.58. A multi-class classification problem's ROC curve contrasts the real beneficial rate against the rate of false positives at various classification levels. The area under the ROC curve (AUC) for classification models with a higher AUC score for each class indicates better performance, such as 0.9,1.0. Here, the model has achieved perfect scores for all of these metrics, indicating that it is accurately predicting the classes in the dataset.

Additionally, a confusion matrix is printed, which shows the number of positives and negatives for each class. Since the model has perfect performance, the confusion matrix shows only the true positives for each class and zeros elsewhere. Finally, the classification report shows precision, recall, and F1 scores for each class, as well as the weighted average across all classes. In this case, all classes have perfect precision and recall, and F1 scores are 1.0.

In conclusion, evaluating the performance of a neural network for lung cancer prediction involves data collection and preprocessing, feature selection, train-test split, cross-validation, ROC curve analysis, and confusion matrix analysis.

CONCLUSION

To summarize, predicting lung cancer is one of the most difficult medical problems to solve. If the treatment of lung cancer is delayed, the probability of death rates rises considerably. Cancer can be curable if identified and treated early enough. MLP is used to predict the development of lung cancer. The suggested method's performance evaluation provided favorable findings, suggesting that MLP may be utilized successfully by oncologists to help in the treatment of cancer. Using a multilayer perceptron (MLP) neural network to predict lung cancer can be an effective method. The use of MLPs for lung cancer prediction is a promising field of study that has the potential to enhance patient diagnosis and treatment results.

REFERENCES

Panchal, F. S., & Panchal, M. (2014). Review on methods of selecting number of hidden nodes in artificial neural network. *International Journal of Computer Science and Mobile Computing*, *3*(11), 455-464.

Potghan, S., Rajamenakshi, R., & Bhise, A. (2018, March). Multi-layer perceptron based lung tumor classification. In *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)* (pp. 499-502). IEEE.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). *Learning internal representations by error propagation*. California Univ San Diego La Jolla Inst for Cognitive Science.

Xiao, D., Li, B., & Mao, Y. (2017). A multiple hidden layers extreme learning machine method and its application. *Mathematical Problems in Engineering*, *2017*.