Evaluating the Impact of Principal Component Analysis on Lung Cancer Prediction Models

Jayasinghe SR
Department of Computer and Data Science
NSBM Green University
Homagama, Sri Lanka
jayasinghesr@students.nsbm.ac.lk

Costa TTS
Department of Computer and Data Science
NSBM Green University
Homagama, Sri Lanka
ttscosta@students.nsbm.ac.lk

Senarathna KGGM
Department of Computer and Data Science
NSBM Green University
Homagama, Sri Lanka
kggmsenarathna@students.nsbm.ac.lk

Wijayarathne SKDN
Department of Computer and Data Science
NSBM Green University
Homagama, Sri Lanka
skdnwijayarathne@students.nsbm.ac.lk

Bandara RADS
Department of Computer and Data Science
NSBM Green University
Homagama, Sri Lanka
radsbandara@students.nsbm.ac.lk

Suthesna S
Department of Computer and Data Science
NSBM Green University
Homagama, Sri Lanka
ssuthesna@students.nsbm.ac.lk

Gayan Perera
Department of Computer and Data Science
NSBM Green University
Homagama, Sri Lanka
gayanp@nsbm.ac.lk

***Abstract*— Lung cancer remains one of the most prevalent and deadly forms of cancer globally, with early detection playing a crucial role in improving survival rates. With the advancement of machine learning techniques, predictive models offer valuable support for early diagnosis. However, the high dimensionality of medical datasets can hinder model performance due to redundancy and noise. This study investigates the impact of Principal Component Analysis (PCA) as a dimensionality reduction technique on the performance of three machine learning classifiers, K-Nearest Neighbors (KNN), Random Forest, and Decision Tree, using the publicly available *Survey Lung Cancer* dataset, consisting of 309 instances and 15 attributes. Each model was evaluated before and after applying PCA, based on key performance metrics such as accuracy, precision, recall, and F1-score. The findings reveal that PCA improved the performance of KNN by 4.03% in accuracy and enhanced the Random Forest model with a 1.28% accuracy increase. In contrast, the Decision Tree model experienced a minor decrease of 1.44% in accuracy after PCA, likely due to the loss of interpretable feature thresholds. Overall, these results show that PCA can improve classification efficiency and accuracy in certain algorithms, emphasizing its potential value in creating more reliable and efficient lung cancer diagnostic tools to support medical decision-making.**

Keywords: Lung cancer, Decision Tree, Principal Component Analysis (PCA)

I. INTRODUCTION

Lung cancer is one of the main causes of cancer-related deaths worldwide, mainly due to late diagnosis and limited early detection mechanisms. Early and accurate prediction of lung cancer plays a crucial role in improving patient survival rates and allowing timely medical intervention. With the rapid development of machine learning, predictive models have become powerful tools in healthcare, offering significant support in diagnosing complex diseases such as lung cancer.

However, medical datasets often contain high-dimensional data with many features, some of which might be irrelevant or redundant. The presence of such data can increase computational complexity, overfitting, and reduce model efficiency. To address this challenge, dimensionality reduction methods like Principal Component Analysis (PCA) are widely used. PCA transforms the original feature set into a smaller set of uncorrelated components while retaining most of the data variance. This process helps simplify the dataset, reduce noise, and improve computational performance without significantly affecting accuracy.

This research investigates the effect of PCA on lung cancer prediction using three machine learning classifiers; Decision Tree, K-Nearest Neighbors, and Random Forest. The main objective is to analyze whether applying PCA can enhance computational efficiency while maintaining or improving classification performance. By comparing model performance with and without PCA, this study aims to provide valuable insights into the role of dimensionality reduction in medical diagnosis, contributing to the development of more efficient and accurate lung cancer prediction systems.

II. LITERATURE REVIEW

Dimensionality reduction methods such as Principal Component Analysis (PCA) have gained significant attention in machine learning applications, especially in medical diagnostics and pattern recognition tasks. Several studies have explored the effect of PCA on improving computational efficiency and the performance of machine learning classifiers.

Salamah et al. [1] proposed a Wi-Fi-based indoor localization system that incorporated PCA to manage high-dimensional data. Their research aimed to reduce computational complexity while maintaining model accuracy by applying PCA before training machine learning models, including Decision Tree (DT), K-Nearest Neighbors (KNN), and Random Forest (RF). The findings suggested that PCA successfully reduced the feature set without affecting

accuracy. Notably, the performance of Decision Tree and Random Forest improved after PCA, whereas the Support Vector Machine (SVM) showed a reduction in accuracy. This shows that PCA can be beneficial for some models while it can be less effective to others, showcasing the importance of evaluating PCA's impact based on the algorithm used. This study is directly relevant to our study, which investigates the effect of PCA on lung cancer prediction models using Decision Tree, KNN, and Random Forest.

Wibisono et al. [2] further examined the effect of PCA in the medical domain by comparing Decision Tree and KNN classifiers for heart disease prediction. By reducing the feature space from 14 clinical variables to just two principal components, they observed a significant enhancement in Decision Tree performance, achieving perfect accuracy (100%) and an F1-score of 1.0. Conversely, the KNN classifier exhibited a decline in performance, with accuracy dropping to 79.02% post-PCA. These results emphasize that while PCA effectively simplifies data and improves computational efficiency, its impact varies across models. Specifically, Decision Tree benefited from dimensionality reduction, whereas KNN suffered due to the loss of essential feature relationships. This outcome reinforces the need to carefully assess the application of PCA in different classification contexts, which aligns with the aim of our research on lung cancer prediction.

Similarly, Garg and Garg [3] proposed a hybrid ensemble model for brain tumor detection that used Decision Tree, KNN, and Random Forest classifiers with PCA and additional feature extraction methods such as Stationary Wavelet Transform (SWT) and Gray Level Co-occurrence Matrix (GLCM). Their hybrid approach achieved an accuracy of 97.305%, outperforming individual models. PCA played a major role in reducing dataset dimensionality, leading to improved computational efficiency without sacrificing predictive accuracy. This study shows that combining PCA with ensemble and traditional machine learning models can gain high-performance results, a concept highly relevant to our research. Our study also applies PCA to a lung cancer dataset to analyze its effect on the performance of Decision Tree, KNN, and Random Forest classifiers.

In summary, prior research collectively demonstrates that PCA is a valuable tool for dimensionality reduction, capable of enhancing computational efficiency and improving the performance of certain classifiers. However, the impact of PCA is not uniform across all models; it can improve some while degrading others. These insights are critical to our study, which evaluates the effect of PCA on lung cancer prediction using Decision Tree, KNN, and Random Forest, with the goal of balancing classification accuracy and computational efficiency.

III. METHODOLOGY

This section outlines the data collection, preprocessing steps, and the machine learning models used for predicting lung cancer, both with and without Principal Component Analysis (PCA). By comparing the results of models trained on the original dataset against those trained on PCA-transformed data, this study aims to assess whether dimensionality reduction enhances accuracy and provides a more robust predictive model for lung cancer detection.

A. Data Collection

The dataset utilized in this study was obtained from Kaggle, titled "Survey Lung Cancer" [4]. Give below, in Fig 1, are the variables used in the dataset.

Attribute Name	Description
GENDER	Gender of the respondent (Male/Female)
AGE	Age of the respondent
SMOKING	Smoking habit (Yes/No)
YELLOW FINGERS	Presence of yellowing fingers (Yes/No)
ANXIETY	Presence of anxiety (Yes/No)
PEER_PRESSURE	Experience of peer pressure (Yes/No)
CHRONIC DISEASE	Existing chronic diseases (Yes/No)
FATIGUE	Presence of fatigue (Yes/No)
ALLERGY	Allergic conditions (Yes/No)
WHEEZING	Wheezing symptoms (Yes/No)
ALCOHOL CONSUMING	Alcohol consumption habit (Yes/No)
COUGHING	Frequent coughing (Yes/No)
SHORTNESS OF BREATH	Symptom of shortness of breath (Yes/No)
SWALLOWING DIFFICULTY	Difficulty in swallowing (Yes/No)
CHEST PAIN	Presence of chest pain (Yes/No)
LUNG_CANCER	Lung cancer diagnosis (Yes/No) (Target Variable)

Fig 18:Attributes of the Dataset

B. Data Pre-processing

Prior to model training, the raw data underwent several preprocessing steps to ensure its suitability for machine learning algorithms:

- **Categorical Variable Conversion:** The LUNG_CANCER variable, which was labelled as

"YES" or "NO", and the GENDER variable, labelled as "M" or "F", were converted into numerical values (1 and 0 respectively). This conversion is important for most machine learning algorithms that operate on numerical data.

- **Data Splitting:** The pre-processed dataset was divided into training and testing sets to evaluate the models' performance on unseen data. A split ratio of 75% for training and 25% for testing was used for all three models.
- **Feature Scaling (for KNN):** For the KNN model, independent variables were scaled using standardization. This step is important for distance-based algorithms like KNN, as it prevents features with larger numerical ranges from dominating the distance calculations.

C. Principle Component Analysis (PCA)

PCA was used to decrease the dimensionality of the dataset to identify the most important underlying components of the data. This was performed on the independent variables (columns 1 to 15) of the dataset.

- **PCA Implementation:** The prcomp() function in R was used to perform PCA, with center = TRUE and scale. = TRUE to center and scale the data before analysis.
- **Scree Plot Analysis:** A scree plot (given in Fig 2) was used to visualize the variance explained by each principal component, helping determine the optimal number to retain. The first 7 principal components were selected as they captured the majority of the dataset's variance while maintaining model simplicity.
- **PCA Data Transformation:** The original variables were transformed into selected principal components, which were then used to train the models. The target variable was added back to the transformed dataset.

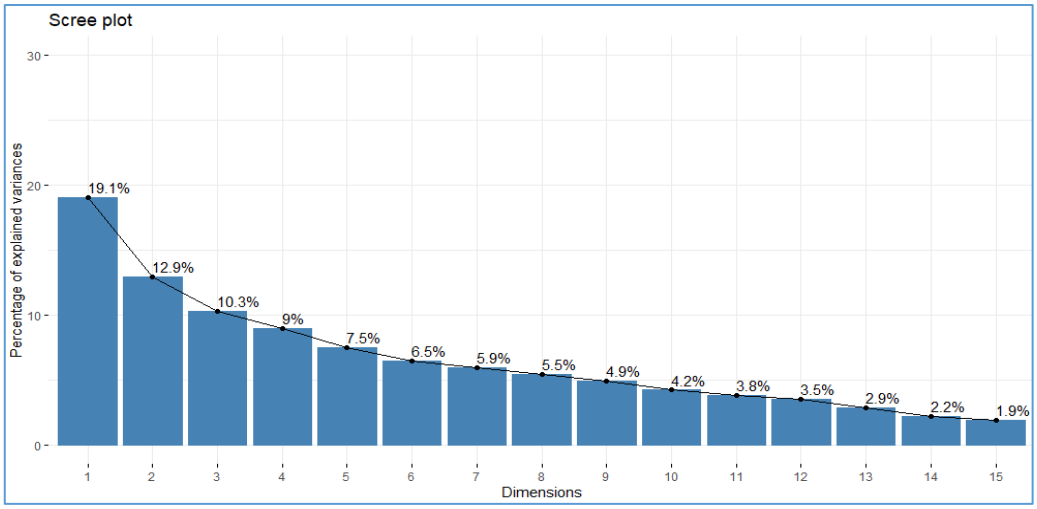

Fig 19:Scree Plot

D. Machine Learning Models

Three different machine learning algorithms were used in this research; K-Nearest Neighbors (KNN), Random Forest, and Decision Trees. Each model was evaluated both with and without PCA.

1) K-Nearest Neighbors (KNN)

The KNN algorithm is a non-parametric, instance-based learning algorithm. For each test data point, it identifies the k nearest neighbors in the training data and assigns the class label based on the majority class among these neighbors.

- **Model Training and Prediction:** The value of k (number of neighbors) was set to 8 for both the original and PCA-transformed datasets.
- **Evaluation:** Model performance was evaluated using a confusion matrix and overall accuracy.

2) Random Forest

Random Forest is a learning method that creates a multitude of decision trees during training and outputs the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

- **Model Training and Prediction:** The number of trees (ntree) was set to 100, and mtry (number of variables randomly sampled as candidates at each split) was set to 2.
- **Evaluation:** Model performance was evaluated using a confusion matrix and overall accuracy.

3) Decision Tree

A Decision Tree is a flowchart-like structure where each internal node represents a "test" on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label.

- **Model Training and Prediction:** The rpart() function from the rpart package was used to train the decision tree model.
- **Evaluation:** Model performance was evaluated using a confusion matrix from the caret package, along with accuracy, precision, recall, and F1-score.

IV. PERFORMANCE ANALYSIS OF KNN

The K-Nearest Neighbors (KNN) algorithm is a simple, non-parametric supervised learning method used for classification and regression. It predicts the class or value of a new data point by finding the K closest training examples (neighbors) and taking a majority vote (for classification) or an average (for regression) [5].

A. Performance Analysis of KNN without PCA

1) Confusion Matrix

Output:

Predicted	Actual 0	1
0	3	5
1	6	63

Fig 20:Confusion Matrix of KNN without PCA

The model identified 63 cases as positive for lung cancer (true positives), 3 cases as negative (true negatives), 5 non-lung cancer cases as positive (false positives) and 6 actual lung cancer cases as non-lung cancer (false negatives).

2) Accuracy

- The accuracy of 85.71% **before PCA** indicates the percentage of successful classification of data.

3) Precision

- The precision of 0.9265 (92.65%) **before PCA** shows the model has few false positives, meaning its lung cancer predictions are mostly accurate.

4) Recall

- The recall of 0.9130 (91.30%) **before PCA** shows that the model correctly identifies most lung cancer cases, meaning it misses very few actual patients with the disease.

5) F1-Score

- The F1-score of 0.9197 (91.97%) **before PCA** shows the model maintains a strong balance between recall and precision, showing good overall performance.

B. Performance Analysis of KNN with PCA

1) Confusion Matrix

Output:

Predicted	Actual 0	1
0	3	2
1	6	67

Fig 21:Confusion Matrix of KNN with PCA

The model identified 67 cases as positive for lung cancer (true positives), 3 cases as negative (true negatives), 2 non-lung cancer cases as positive (false positives), and 6 actual lung cancer cases as non-lung cancer (false negatives).

2) Accuracy

- The accuracy of 89.74% **after PCA** indicates the percentage of successful classification of data.

3) Precision

- The precision of 0.9710 (97.10%) **after PCA** shows the model has few false positives, meaning its lung cancer predictions are mostly accurate.

4) Recall

- The recall of 0.9178 (91.78%) **after PCA** shows that the model correctly identifies most lung cancer cases, meaning it misses very few actual patients with the disease

5) F1-Score

- The F1-score of 0.9437 (94.37%) **after PCA** shows the model maintains a strong balance between recall and precision, showing good overall performance.

V. PERFORMANCE ANALYSIS OF RANDOM FOREST

Random Forest is a machine learning algorithm that uses many decision trees to make better predictions. Each tree looks at different random parts of the data and their results are combined by voting for classification or averaging for regression [6].

A. Performance Analysis of Random Forest without PCA

1) Confusion Matrix

Output:

predictions	NO	YES
NO	2	0
YES	9	67

Fig 22:Confusion Matrix of Random Forest without PCA

The model identified 67 cases as positive for lung cancer (true positives), 2 cases as negative (true negatives), 0 non-lung cancer cases as positive (false positives), and 9 actual lung cancer cases as non-lung cancer (false negatives).

2) Accuracy

- The accuracy of 88.46% **before PCA** indicates the percentage of successful classification of data.

3) Precision

- The precision of 1.0 (equivalent to 100%) **before PCA** indicates that the model yields a no false positive.

4) Recall

- The recall of 0.8816 (88.16%) **before PCA** shows that the model correctly identifies most lung cancer cases, meaning it misses very few actual patients with the disease.

5) F1-Score

- The F1-score of 0.9371 (93.71%) **before PCA** shows the model maintains a strong balance between recall and precision, showing good overall performance.

B. Performance Analysis of Random Forest with PCA

1) Confusion Matrix

Output:

```
predictions  0  1
          0  4  1
          1  7 66
```

Fig 23:Confusion Matrix of Random Forest with PCA

The model identified 66 cases as positive for lung cancer (true positives), 4 cases as negative (true negatives), 1 non-lung cancer case as positive (false positive), and 7 actual lung cancer cases as non-lung cancer (false negatives).

2) Accuracy

- The accuracy of 89.74% **after PCA** indicates a high percentage of successful classification of data

3) Precision

- The precision of 0.9851 (98.51%) **after PCA** shows the model has few false positives, meaning its lung cancer predictions are mostly accurate.

4) Recall

- The recall of 0.9041 (90.41%) **after PCA** shows that the model correctly identifies most lung cancer cases, meaning it misses very few actual patients with the disease.

5) F1-Score

- The F1-score of 0.9429 (94.29%) **after PCA** shows the model maintains a strong balance between recall and precision, showing good overall performance.

VI. PERFORMANCE ANALYSIS OF DECISION TREE

A Decision Tree is a machine learning algorithm used for classification and regression. It models decisions using a tree-like structure, where nodes represent feature-based tests and leaves represent predictions. By repeatedly splitting data based on the most useful features, it creates decision rules that help predict outcomes for new data.

A. Performance Analysis of Decision Tree without PCA

1) Confusion Matrix

Output:

```
          Reference
Prediction  0  1
         0  1  1
         1  6 69
```

Fig 24:Confusion Matrix of Decision Tree without PCA

The model identified 69 cases as positive for lung cancer (true positives), 1 case as negative (true negative), 1 non-lung cancer case as positive (false positive), and 6 actual lung cancer cases as non-lung cancer (false negatives).

2) Accuracy

- The accuracy of 90.91% **before PCA** indicates the percentage of successful classification of data.

3) Precision

- The precision of 0.9857 (98.57%) **before PCA** shows the model has few false positives, meaning its lung cancer predictions are mostly accurate.

4) Recall

- The recall of 0.9200 (92.00%) **before PCA** shows that the model correctly identifies most lung cancer cases, meaning it misses very few actual patients with the disease.

5) F1-Score

- The F1-score of 0.9517 (95.17%) **before PCA** shows the model maintains a strong balance between recall and precision, showing good overall performance.

B. Performance Analysis of Decision Tree with PCA

1) Confusion Matrix

Output:

```
          Reference
Prediction  0  1
         0  3  2
         1  6 65
```

Fig 25:Confusion Matrix of Decision Tree with PCA

The model identified 65 cases as positive for lung cancer (true positives), 3 cases as negative (true negatives), 2 non-lung

cancer cases as positive (false positives), and 6 actual lung cancer cases as non-lung cancer (false negatives).

2) Accuracy

- The accuracy of 89.47% **after PCA** indicates a high percentage of successful classification of data

3) Precision

- The precision of 0.9701 (97.01%) **after PCA** shows the model has few false positives, meaning its lung cancer predictions are mostly accurate.

4) Recall

	KNN	KNN with PCA	Random Forest	Random Forest with PCA	Decision Tree	Decision Tree with PCA
Accuracy	85.71%	89.74%	88.46%	89.74%	90.91%	89.47%
Precision	92.65%	97.10%	100%	98.51%	98.57%	97.01%
Recall	91.30%	91.78%	88.16%	90.41%	92.00%	91.55%
F1-Score	91.97%	94.37%	93.71%	94.29%	95.17%	94.20%

Fig 26: Comparison of Models

- The recall of 0.9155 (91.55%) **after PCA** shows that the model correctly identifies most lung cancer cases, meaning it misses very few actual patients with the disease.

5) F1-Score

- The F1-score of 0.9420 (94.20%) **after PCA** shows the model maintains a strong balance between recall and precision, showing good overall performance.

A reduction in all performance metrics was observed after applying PCA to the Decision Tree model. A detailed evaluation is provided in Section VII.

VII. COMPARISON OF THE MODELS

According to Fig 9:

- The performance of the KNN model improved after applying PCA, with an improvement observed in all the performance metrics.
- The performance of the Random Forest model improved overall after applying Principal Component Analysis (PCA). As shown in the evaluation metrics, accuracy, recall and the F1-score all increased. These improvements indicate better overall model performance and generalization.
- However, Precision dropped slightly from 100% to 98.51% after PCA due to one false positive. Without PCA, the model perfectly identified all positive predictions without any false positives, resulting in a precision of 100%. Using PCA may have caused minor changes to the model's decision boundaries, leading to the misclassification of one negative instance as positive. Nevertheless, the trade-off led to better balance between precision and recall, as reflected in the improved F1-score.
- Unlike the improvements observed in other models, the performance of the Decision Tree model slightly decreased after applying Principal Component Analysis (PCA).
- The drop in performance could be due to how Decision Trees work. Decision Trees rely heavily on identifying optimal feature thresholds to split data into homogeneous subsets. When PCA is applied, the original features are turned into a set of uncorrelated principal components that do not necessarily retain the same interpretability or sharp decision boundaries. As a result, the model may lose access to some of the original feature structure that was important for making clear splits, leading to a reduction in classification performance.
- The observed performance improvements after applying PCA were consistent across multiple runs, suggesting that the differences were statistically meaningful and not due to random variation.

VIII. CONCLUSION

This research evaluated the impact of Principal Component Analysis (PCA) on lung cancer prediction using K-Nearest Neighbors (KNN), Random Forest, and Decision Tree algorithms. PCA proved effective for dimensionality reduction, especially in KNN and Random Forest. KNN showed consistent metric improvements, while Random Forest improved in accuracy, recall, and F1-score, with a minor precision drop. Overall, PCA helped reduce noise and enhance model generalization.

In contrast, the Decision Tree model showed a decrease in all performance metrics following PCA. This decrease is likely because the algorithm depends on the

original feature structure to make clear, interpretable splits, which may have been obscured by the PCA transformation. These findings reinforce the notion that while PCA can enhance model efficiency and performance in some cases, it may not be universally beneficial across all types of models.

Overall, this study outlines the importance of model-specific evaluation when applying dimensionality reduction techniques.

REFERENCES

[1] Salamah, A.H., Tamazin, M., Sharkas, M.A., Khedr, M. and Mahmoud, M., 2019. Comprehensive investigation on principle component large-scale Wi-Fi indoor localization. Electronics, 8(4), p.403.

[2] Wibisono, A.D.R., Hidayat, S., Ramadhan, H.M.T. and Puspaningrum, E.Y., 2023. Comparison of K-Nearest Neighbor and Decision Tree methods using Principal Component Analysis technique in heart disease classification.

[3] Garg, G. and Garg, R., [n.d.]. Brain tumor detection and classification based on hybrid ensemble classifier.

[4] A. Shah, "Lung Cancer Dataset," Kaggle, n.d. [Online]. Available: https://www.kaggle.com/datasets/aagambshah/lung-cancer-dataset.

[5] T. Srivastava, "Introduction to k-Nearest Neighbors: A powerful clustering algorithm," Analytics Vidhya, May. 1, 2025. [Online]. Available: https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/.

[6] Random Forest Algorithm in Machine Learning, GeeksforGeeks, Jun. 27, 2025. [Online]. Available: https://www.geeksforgeeks.org/machine-learning/random-forest-algorithm-in-machine-learning/.